

Towards 3D Human Pose Estimation in the Wild: a Weakly-supervised Approach

Xingyi Zhou^{1,2}, Qixing Huang², Xiao Sun³, Xiangyang Xue¹, Yichen Wei³

¹Shanghai Key Laboratory of Intelligent Information Processing

School of Computer Science, Fudan University

² The University of Texas at Austin

³ Microsoft Research

{zhouxy13, xyxue}@fudan.edu.cn, huangqx@cs.utexas.edu, {xias, yichenw}@microsoft.com

Abstract

In this paper, we study the task of 3D human pose estimation in the wild. This task is challenging due to lack of training data, as existing datasets are either in the wild images with 2D pose or in the lab images with 3D pose.

We propose a weakly-supervised transfer learning method that uses mixed 2D and 3D labels in a unified deep neural network that presents two-stage cascaded structure. Our network augments a state-of-the-art 2D pose estimation sub-network with a 3D depth regression sub-network. Unlike previous two stage approaches that train the two sub-networks sequentially and separately, our training is end-to-end and fully exploits the correlation between the 2D pose and depth estimation sub-tasks. The deep features are better learnt through shared representations. In doing so, the 3D pose labels in controlled lab environments are transferred to in the wild images. In addition, we introduce a 3D geometric constraint to regularize the 3D pose prediction, which is effective in the absence of ground truth depth labels. Our method achieves competitive results on both 2D and 3D benchmarks.

1. Introduction

Human pose estimation problem has been heavily studied in computer vision. It has numerous important applications in human-computer interaction, virtual reality, and action recognition. Existing research works falls into two categories: 2D pose estimation and 3D pose estimation. Thanks to the availability of large-scale 2D annotated human poses and the emergence of deep neural networks, the 2D human pose estimation problem has gained tremendous success recently [17, 29, 11, 4, 7]. State-of-the-art techniques are able to achieve accurate predictions across a wide range of settings (e.g., on images in the wild [2]).

In contrast, advance in 3D human pose estimation re-

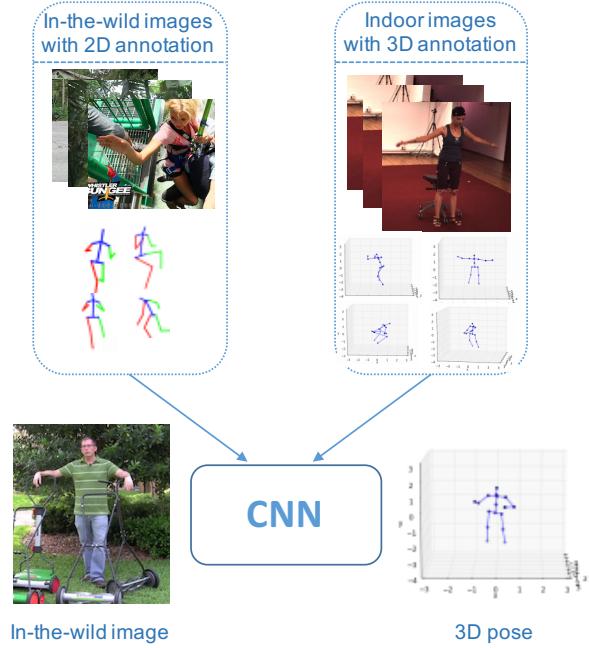


Figure 1. A schematic illustration of our method: transferring 3D annotation from indoor images to in-the-wild images. Top (Training): Both indoor images with 3D annotation (Right) and in-the-wild images with 2D annotation (Left) are used to train the deep neural network. Bottom (Testing): The learned network can predict the 3D pose of the human in in-the-wild images.

mains limited. This is partially due to the ambiguity of recovering 3D information from single images, and partially due to the lack of large scale 3D pose annotation dataset. Specifically, there is not yet a comprehensive 3D human pose dataset for images in the wild. The commonly used 3D datasets [12, 24] were captured by mocap systems in controlled lab environments. Deep neural networks [13, 33] trained on these datasets do not generalize well to other environments, such as in the wild.

There has been quite a few works on 3D human pose estimation in the wild. They usually proceed in two sequential steps [34, 26, 5, 3, 30, 31]. The first step estimates 2D joint locations [17, 29, 11]. The second step recovers a 3D pose from these 2D joints [21, 32, 1]. Training in the two steps are performed separately. Namely, 2D pose predictions are trained from 2D annotations in the wild, and 3D pose recovery from 2D joints is trained from existing 3D MoCap data. Such a sequential pipeline is clearly sub-optimal because the original in-the-wild 2D image information, which contains rich cues for 3D pose recovery, is discarded in the second step.

Recently, Mehta et al. [15] have shown that 2D-to-3D knowledge transfer, i.e., using pre-trained 2D pose networks to initialize the 3D pose regression networks can significantly improve 3D pose estimation performance. This indicates that the 2D and 3D pose estimation tasks are inherently entangled and could share common representations.

Inspired by this work, we argue that the inverse knowledge transfer, i.e., from 3D annotations of indoor images to in-the-wild images, offers an effective solution for 3D pose prediction in the wild. In this work, we introduce a unified framework that can exploit 2D annotations of in-the-wild images as weak labels for the 3D pose estimation task. In other words, we consider a weakly-supervised transfer learning problem, where the source domain consists of fully annotated images in restricted indoor environment and the target domain consists of weakly-labeled images in the wild.

Similar to previous works [34, 26, 5, 3, 30, 31], our network also consists of a 2D module and a 3D module. However, instead of merely feeding the output of the 2D module as input to the 3D module, our approach connect the 3D module with the intermediate layers of the 2D module. This allows us to share the common representations between the 2D and the 3D tasks. The network is trained end-to-end with both 2D and 3D data simultaneously. This distinguishes our work from all existing works.

To better regularize the learning of weakly-supervised 3D pose estimation, we introduce a geometric constraint for training the 3D module. The geometric constraint is based on the fact that relative bone length in a human skeleton remains approximately fixed. The effectiveness of this constraint is experimentally verified when adapting the 3D pose information from labeled images in indoor environments to unlabeled images in the wild.

This work makes the following contributions:

- For the first time, we propose an end-to-end 3D human pose estimation framework for in-the-wild images. It achieves state-of-the-art performance on several benchmarks.
- We propose a 3D geometric constraint for 3D pose es-

timation from images with only 2D joint annotations. It has low cost in memory and computation. It improves the geometric validity of estimated poses.

Code is publicly available at <https://github.com/xingyizhou/pose-hg-3d>.

2. Related Work

Human pose estimation has been studied considerably in the past [16, 23], and it is beyond the scope of this paper to provide a complete overview of the literature. In this section, we focus on previous works on 3D human pose estimation, which are most relevant to the context of this paper. We will also discuss related works on imposing weakly-/un-supervised constraints for training neural networks.

3D Human Pose Estimation. Given well labeled data (e.g., 3D joint locations of a human skeleton [12, 24]), 3D human pose estimation can be formulated as a standard supervised learning problem. A popular approach is to train a neural network to directly regress joint locations [13]. Recently, people have generalized this approach in different directions. Zhou et al. [33] propose to explicitly enforce the bone-length constraints in the prediction, using a generative forward-kinematic layer; Tekin et al. [25] embed a pre-trained auto-encoder at the top of the network. In contrast these works, Pavlakos et al. introduce a 3D approach, which regresses a volumetric representation of 3D skeleton [19]. Despite the performance gain on standard 3D pose estimation benchmark datasets, the resulting networks do not generalize to images in the wild due to the domain difference between natural images and the specific capture environments utilized by these benchmark datasets.

A standard approach to address the domain difference between 3D human pose estimation datasets and images in the wild is to split the task into two separate sub-tasks [34, 26, 5, 3, 30]. The first sub-task estimates 2D joint locations. This sub-task can utilize any existing 2D human pose estimation method (e.g., [17, 29, 11, 4]) and can be trained from datasets of in-the-wild images. The second sub-task regresses the 3D locations of these 2D joints. Since the input at this step is just a set of 2D locations, the 3D pose estimation network can be trained on any benchmark datasets and then adapted in other settings. Regarding 3D pose estimation from 2D joint locations, [34] use an EM algorithm to compute a 3D skeleton by combining a sparse dictionary induced from the 2D heat-maps; [30, 19] use 3D pose data and its 2D projection to train a heatmap-to-3D pose network without the original image; Bogo et al. [3] optimize both the pose and shape terms of a linear 3D human model [14] to best fit its 2D projection; Chen et al. [5] use nearest-neighbor search to match the estimated 2D pose to a 3D pose as well as a camera-view which may produce a similar 2D projection from a large 3D pose library; finally,

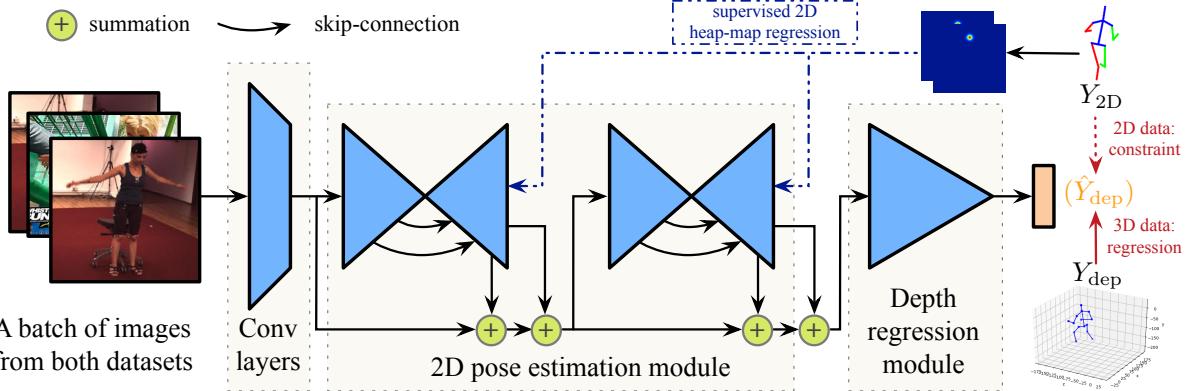


Figure 2. Illustration of our framework: In testing, images go through the stacked hourglass network and turn into 2D heat-maps. The 2D heat-maps and with lower-layer images features are summed as the input of the following depth regression module. In training, images from both 2D and 3D datasets are mixed in a single batch. For the 3D data, the standard regression with Euclidean Loss is applied. For the 2D data, we propose a weakly-supervised loss based on its 2D annotation and prior knowledge of human skeleton.

Tome et al. [26] propose a pre-trained probabilistic 3D pose model layer that first generates plausible 3D human model from 2D heat-maps, and then refines these heat maps by combining 3D pose projection and image features. All these methods, however, share a common limitation: the 3D pose is only estimated from the 2D joints, which is known to produce ambiguous results. In contrast, our approach leverages both 2D joint locations as well as intermediate feature representations from the original image.

An alternative approach for 3D human pose estimation is to train from synthetic datasets which are generated from deforming a human template model with known 3D ground truth [6, 22]. This is indeed a viable solution, but the fundamental challenge is how to model the 3D environment so that the distribution of the synthesized images matches that of the natural images. It turns out state-of-the-art methods along this line are less competitive on natural images.

There are also other works utilizing mixed 2D and 3D data for 3D human pose estimation. Mehta et al. [15] fine-tune a pre-trained 2D pose estimation network with 3D data. Popa et al. [20] consider 3D human pose estimation as a multi-task learning of 2D and depth regression with different data. Ours is different from those work that we use a weakly-supervised loss that seamlessly integrates both 2D and 3D data in a unified framework.

Weakly-/un-supervised constraints. In the presence of insufficient training data, incorporating generic or weakly supervised constraints among the prediction serves as a powerful tool for performance boosting. This idea was usually utilized in image classification or segmentation. Pathak et al. [18] propose a constrained optimization framework that utilizes a linear constraint over sum of label probabilities for weakly supervised semantic segmentation. Tzeng et al. [28] propose a domain confusion loss to maximize

the confusion between two datasets so as to encourage a domain-invariant feature. Recently, Hoffman et al. [10] introduce an adversarial learning based global domain alignment method and utilize a weak label constraint to apply fully connected networks in the wild. In this paper, we show this general concept can be used for pose estimation as well. To best of our knowledge, our approach is the first to leverage geometry-guided constraint to regularize the pose estimation network for images in the wild.

3. Approach

3.1. Overview

Given an RGB image I containing a human subject, we aim to estimate the 3D human pose $Y \in \mathcal{Y}_{3D}$, represented by a set of 3D joint coordinates of the human skeleton, i.e. $\mathcal{Y}_{3D} \in \mathcal{R}^{J \times 3}$, where J is the number of joints. We follow the convention of representing each 3D coordinate in the local camera coordinate system associated with I , namely, the first two coordinates are given by image pixel coordinates (which define the corresponding 2D joint location), and the third coordinate is the joint depth in metric coordinates, e.g., millimeters in this work.

Our proposed network architecture is illustrated in Fig. 2. It consists of a 2D pose estimation module (Section 3.2) and a depth regression module (Section 3.3). They predict the 2D joint locations $Y_{2D} \in \mathcal{Y}_{2D}$, where $\mathcal{Y}_{2D} \subset \mathcal{R}^{J \times 2}$, and the depth values $Y_{dep} \in \mathcal{Y}_{dep}$, where $\mathcal{Y}_{dep} \subset \mathcal{R}^{J \times 1}$, respectively. The final output is the concatenation of Y_{2D} and Y_{dep} .

The network is trained from both images in the lab with 3D ground truth (for both \mathcal{Y}_{2D} and \mathcal{Y}_{dep}) and images in the wild with only 2D ground truth (for \mathcal{Y}_{2D}). In the reminder of this paper, the 3D and 2D training image sets are denoted

as \mathcal{I}_{3D} and \mathcal{I}_{2D} , respectively.

3.2. 2D Pose Estimation Module

We adopt the state-of-the-art hourglass network architecture in [17] as our 2D pose estimation module. The network output is a set of J low-resolution heat-maps. Each map $\hat{Y}_{HM} \in \mathcal{R}^{H \times W}$ represents a 2D probability distribution of one joint. The predicted joints in the 2D pose $\hat{Y}_{2D} \in \mathcal{Y}_{2D}$ are the peak locations on these heat-maps. This heat-map representation is convenient as it can be easily combined (concatenate or sum) with the other deep layer feature maps, e.g., as shown in Fig 2.

To train this module, the loss function is

$$L_{2D}(\hat{Y}_{HM}, Y_{2D}) = \sum_h^H \sum_w^W (\hat{Y}_{HM}^{(h,w)} - G(Y_{2D})^{(h,w)})^2. \quad (1)$$

The loss measures the L^2 distance between the predicted heat-maps \hat{Y}_{HM} and the heat-maps $G(Y_{2D})$ rendered from the ground truth Y_{2D} through a Gaussian kernel [17].

3.3. Depth Regression Module

Compared with previous methods that recover 3D joint locations from only 2D joint predictions [21, 32, 1], our approach innovates in terms of (i) the integration of 2D and 3D modules for end-to-end network training, and (ii) the usage of a 3D geometric constraint induced loss. They are elaborated below.

Integration of 2D and 3D modules. A key issue for depth estimation is how to effectively exploit image features. A widely used strategy in previous [34, 26, 5] is to take the 2D joint locations as the only input for depth prediction as in this way the Mocap-only data can be utilized. However, this strategy is inherently ambiguous, as there typically exist multiple 3D interpretations of a single 2D skeleton. We propose to combine the 2D joint heatmaps and the intermediate feature representations in the 2D module as input to the depth regression module. These features, which extract semantic information at multiple levels for 2D pose estimation, provide additional cues for 3D pose recovery. This shared common feature learning is crucial in our approach.

3D geometric constraint induced loss. One challenge for depth learning is to how to integrate both fully-labeled and weakly-labeled images. For fully-annotated 3D dataset $\mathcal{S}_{3D} = \{\mathcal{I}_{3D}, \mathcal{Y}_{2D}, \mathcal{Y}_{dep}\}$, the training loss can be simply the standard Euclidean Loss using ground-truth depth label. For weakly-labeled dataset $\mathcal{S}_{2D} = \{\mathcal{I}_{2D}, \mathcal{Y}_{2D}\}$, we propose a novel loss induced from a geometric constraint. In the absence of ground truth depth label, this geometric constraint serves as effective regularization for depth prediction.

Overall, let \hat{Y}_{dep} denote the predicted depth. The loss of the depth regression module is

$$L_{dep}(\hat{Y}_{dep}|I, Y_{2D}) = \begin{cases} \lambda_{reg} ||Y_{dep} - \hat{Y}_{dep}||^2, & \text{if } I \in \mathcal{I}_{3D} \\ \lambda_{geo} L_{geo}(\hat{Y}_{dep}|Y_{2D}), & \text{if } I \in \mathcal{I}_{2D} \end{cases} \quad (2)$$

where λ_{reg} and λ_{geo} are the corresponding loss weights.

$L_{geo}(\hat{Y}_{dep}|Y_{2D})$ is the proposed geometric loss. It is based on the fact that ratios between bone lengths remain relative fixed in a human skeleton (e.g., upper/lower arms have a fixed length ratio, left/right shoulder bones share the same length).

Specifically, let R_i be a set of involved bones in a skeleton group i , e.g. $R_{arm} = \{\text{left upper arm, left lower arm, right upper arm, right lower arm}\}$, let l_e be the length of bone e , and let \bar{l}_e denote the length of bone e in a canonical skeleton (in our experiments, it is set as the average of all training subjects of Human 3.6M dataset). The ratio $\frac{l_e}{\bar{l}_e}$ for each bone e in each group R_i should remain fixed. The proposed loss measures the sum of variance among $\{\frac{l_e}{\bar{l}_e}\}_{e \in R_i}$ of each R_i :

$$L_{geo}(\hat{Y}_{dep}|Y_{2D}) = \sum_i \frac{1}{|R_i|} \sum_{e \in R_i} \left(\frac{l_e}{\bar{l}_e} - \bar{r}_i \right)^2, \quad (3)$$

where

$$\bar{r}_i = \frac{1}{|R_i|} \sum_{e \in R_i} \frac{l_e}{\bar{l}_e}.$$

Note that the bone length is a function of joint locations, which are in turn functions of the predicted depths. Thus, L_{geo} is continuous and differentiable with respect to \hat{Y}_{dep} . The math details of forward and backward equations are provided in the supplemental material. Also note that L_{geo} is defined on the ground truth 2D position Y_{2D} instead of the predicted 2D position \hat{Y}_{2D} . This makes the training easier as there is no back-propagation into the 2D module.

In our experiments, we consider 4 groups of bones: $R_{arm} = \{\text{left/right lower/upper arms}\}$, $R_{leg} = \{\text{left/right lower/upper legs}\}$, $R_{shoulder} = \{\text{left/right shoulder bones}\}$, $R_{hip} = \{\text{left/right hip bones}\}$. We do not include bones on the torso as we found them exhibit relatively high variance in bone lengths across different human shapes, which makes our constraint less valid. Note that bones in different sets do not affect each other.

3.4. Training

Combining the losses in Eq. (1), (2), and (3), the overall loss for each training image $I \subset \mathcal{I}_{2D} \cup \mathcal{I}_{3D}$ is

$$L(\hat{Y}_{HM}, \hat{Y}_{dep}|I) = L_{2D}(\hat{Y}_{HM}, Y_{2D}) + L_{dep}(\hat{Y}_{dep}|I, Y_{2D}). \quad (4)$$

Stochastic gradient descent optimization is used for training. Similar to [28] and [10], each mini-batch contains both the 2D and 3D training examples (half-half), which are randomly sampled.

In experiments, we found the direct end-to-end training of the whole network from scratch does not work well, likely because of the dependency between the two modules and highly non-linear property of the new geometric constraint induced loss. Thus, we propose a three-stage training scheme that we found is more stable and effective in practice. Note that the final stage is end-to-end.

~~Stage 1~~ initializes the 2D pose module using 2D annotated images, as described in [17]. ~~Stage 2~~ initializes the 3D pose estimation module and fine-tunes the 2D pose estimation module. Both 2D and 3D annotated data are used. The geometric constraint is not activated, by setting $\lambda_{geo} = 0$ in Equation 2. ~~Stage 3~~ fine-tunes the whole network with all data. The geometric constraint is activated.

4. Experimental Evaluation

To validate our approach, a single model is trained using Human3.6M data [12] and MPII data [2]. Evaluation is performed on three different testing datasets.

The evaluations are from two aspects: supervised 3D human pose estimation (Section 4.2) and transferred 3D human pose estimation in the wild (Section 4.3).

Qualitative results are summarized in Table 5. More qualitative results on MPII validation set can be found in the supplementary material.

4.1. Experimental Setup

4.1.1 Implementation Detail

Our method was implemented with torch7 [8]. The hourglass component was based on the public code in [17]. For fast training, we used a shallow version of stacked hourglass, i.e. 2 stacks with 2 residual modules [9] for each hourglass. The depth regression module contains 4 sequential residual & pooling modules, which can be regarded as a half hourglass. The same network architecture and training iterations are used in all of our experiments.

The first training stage in Section 3.4 took 240k with a batchsize of 6. This gave us a 2D pose estimation module with similar performance as in [17]. Stage 2 and stage 3 took 200k and 40k iterations, respectively. The whole training procedure took about two days in one Titan X GPU with CUDA 8.0 and cudnn 5. A forward pass at testing is about 30ms. We set $\lambda_{reg} = 0.1$ and $\lambda_{geo} = 0.01$. We followed [17] to set all the other hyper-parameters.

4.1.2 Datasets & Metrics

MPII-training. MPII dataset [2] is used for training. It is a large scale in-the-wild human pose dataset. The images are collected from on-line videos and annotated by human for $J = 16$ 2D joints. It contains 25k training images and 2957 validation images [27]. The human subjects are annotated with bounding boxes. We use the training set of MPII to train the 2D pose estimation module. It also provides weak supervision for the depth regression module.

Human3.6M. Human 3.6M dataset [12] is used both in training and testing. It is a widely used dataset for 3D human pose estimation. This dataset contains 3.6 millions of RGB images captured by a MoCap System in an indoor environment. We down-sampled the video from 50fps to 10fps for both the training and testing sets to reduce redundancy. Following the standard protocol in [13, 34, 33], we use 5 subjects (S1, S5, S6, S7, S8) for training and the rest 2 subjects (S9, S11) for testing. The evaluation metric is mean per joint position error (MPJPE) in mm after aligning the depths of the root joints. We use its projected 2D locations for training the 2D module and its depth annotation for depth regression module.

We use the ground truth 2D joint locations provided in the dataset in training (thus implicitly use the camera calibration information), for aligning the 3D and 2D poses. During testing, such calibration is not needed, by requiring that the sum of all 3D bones lengths is equal to that of a pre-defined canonical skeleton, as is done in [19, 35]. The converting formulation is as follows:

$$\hat{Y} = (Y_{out} - Y_{out}^{(root)}) * \frac{AvgSumLen}{SumLen_{out}} + Y_{GT}^{(root)}$$

Where Y_{out} is the combined 2D and depth 3D joint, which is the output of the network; $SumLen_{out}$ is the calculated sum-of-skeleton-length of the output joints; and $AvgSumLen$ is a constant, which is calculated as the average sum-of-skeleton-length of all the training subjects in Human 3.6M dataset.

MPI-INF-3DHP. MPI-INF-3DHP [15] is a newly proposed 3D human pose dataset. The images were captured by a MoCap system both in indoor and outdoor scenes. We only use its test set split for evaluation. The test set contains 2929 valid frames from 6 subjects, performing 7 actions. Following [15], we employ average PCK (with a threshold 150mm) and AUC as the evaluation metrics, i.e., after aligning the root joint (pelvis). Note that we assume the global scale is known for experimental evaluation. We observe that the definition of pelvis position in MPI-INF-3DHP is different from the one used in our training sets (i.e., Human 3.6M and MPII), so we moved the pelvis and hips towards neck in a fixed ratio (0.2) as post processing in our evaluation.

	Directions	Discussion	Eating	Greeting	Phoning	Photo	Posing	Purchases
Chen & Ramanan [5]	89.87	97.57	89.98	107.87	107.31	139.17	93.56	136.09
Tome et al. [26]	64.98	73.47	76.82	86.43	86.28	110.67	68.93	74.79
Zhou et al. [35]	87.36	109.31	87.05	103.16	116.18	143.32	106.88	99.78
Metha et al. [15]	59.69	69.74	60.55	68.77	76.36	85.42	59.05	75.04
Pavlakos et al. [19]	58.55	64.56	63.66	62.43	66.93	70.74	57.72	62.51
3D/wo geo	73.25	79.17	72.35	83.90	80.25	81.86	69.77	72.74
3D/w geo	72.29	77.15	72.60	81.08	80.81	77.38	68.30	72.85
3D+2D/wo geo	55.17	61.16	58.12	71.75	62.54	67.29	54.81	56.38
3D+2D/w geo	54.82	60.70	58.22	71.41	62.03	65.53	53.83	55.58
	Sitting	SittingDown	Smoking	Waiting	WalkDog	Walking	WalkPair	Average
Chen & Ramanan [5]	133.14	240.12	106.65	106.21	87.03	114.05	90.55	114.18
Tome et al. [26]	110.19	172.91	84.95	85.78	86.26	71.36	73.14	88.39
Zhou et al. [35]	124.52	199.23	107.42	118.09	114.23	79.39	97.70	79.9
Metha et al. [15]	96.19	122.92	70.82	68.45	54.41	82.03	59.79	74.14
Pavlakos et al. [19]	76.84	103.48	65.73	61.56	67.55	56.38	59.47	66.92
3D/wo geo	98.41	141.60	80.01	86.31	61.89	76.32	71.47	82.44
3D/w geo	93.52	131.75	79.61	85.10	67.49	76.95	71.99	80.98
3D+2D/wo geo	74.79	113.99	64.34	68.78	52.22	63.97	57.31	65.69
3D+2D/w geo	75.20	111.59	64.15	66.05	51.43	63.22	55.33	64.90

Table 1. Results of Human3.6M Dataset. The numbers are mean Euclidean distance(mm) between the ground-truth 3D joints and the estimations of different methods.

3D/wo geo	3D/w geo	3D+2D/wo geo	3D+2D/w geo
90.01%	90.57%	90.93%	91.62%

Table 2. 2D pose accuracy (PCKh@0.5) on Human 3.6M dataset.

MPII-Validation. Although MPII dataset does not provide 3D pose annotation, we use its validation subset [27] in our evaluation for two purposes. It contains 2958 in-the-wild images out of the training set.

First, we provide qualitative 3D pose estimation results. Many of them looks plausible and convincing. See more in supplementary material.

Second, we can still evaluate the geometric validity of the estimated 3D pose, which is improved by our proposed constraint. We use the symmetric bone lengths' difference (e.g., left and right upper arms) as the evaluation metric. To compute the metric, we normalize the 2D joints in 256×256 pixels (so that the predicted joints can be directly plotted in the input image). The depth is normalized by the same scale. We then compute the L1 distance between the left and right symmetric bones, e.g. for upper arms it is $\|Y^{(left\ shoulder)} - Y^{(left\ elbow)}\| - \|Y^{(right\ shoulder)} - Y^{(right\ elbow)}\|$. This metric is applied for both MPI-INF-3DHP dataset and MPII-Validation set to evaluate the effectiveness of our proposed weakly-supervised geometric loss.

4.1.3 Baselines for Ablation Study

We implemented three baseline methods and trained the baseline models in the same way as for proposed method.

3D/wo geo It only uses 3D labeled data to train the network in *Stage2* and *Stage3* of Sec. 3.4. The in-the-wild images are not used. Note that the 2D hourglass module is pre-trained on the 2D dataset in *Stage1*.

3D/w geo It adds the geometric constraint induced loss into the first baseline.

3D+2D/wo geo Its only difference from the proposed method is that the geometric constraint is not utilized for 2D labeled data when training the 3D module.

The proposed method is denoted as **3D+2D/w geo**.

4.2. Supervised 3D Human Pose Estimation

We first report and analyze the performance of our method on Human 3.6M dataset [12].

Baseline comparison. Table 1 compares the proposed approach with the three baselines. The average MPJPE of baseline **3D/wo geo** is 82.44mm. This is already comparable to most state-of-the-art methods [33, 26, 35]. Note that this baseline is similar with Metha et al. [15], which fine-tuned 2D pose network [11] with 3D data for information transfer. The difference is that we did not use $1000 \times$ learning rate decay for the transferred layers, which in our case yielded worse performance.

	Studio GS	Studio no GS	Outdoor	ALL	PCK	AUC
Metha et al.(H36M+MPII) [15]	70.8	62.3	58.8	64.7	31.7	
3D/wo geo	34.4	40.8	13.6	31.5	18.0	
3D/w geo	45.6	45.1	14.4	37.7	20.9	
3D+2D/wo geo	68.8	61.2	67.5	65.8	32.1	
3D+2D/w geo	71.1	64.7	72.7	69.2	32.5	
Metha et al.(MPI-INF-3DHP) [15]	84.1	68.9	59.6	72.5	36.9	

Table 3. Results of MPI-INF-3DHP Dataset by scene. GS indicates green screen background. The results are shown in PCK and AUC.

	3D+2D/wo geo	3D+2D/w geo
Upper arm	42.4mm	37.8mm
Lower arm	60.4mm	50.7mm
Upper leg	43.5mm	43.4mm
Lower leg	59.4mm	47.8mm
Upper arm	6.27px	4.80px
Lower arm	10.11px	6.64px
Upper leg	6.89px	4.93px
Lower leg	8.03px	6.22px

Table 4. Evaluation of left-right Symmetry of with and without constraint on MPI-INF-3DHP(Up) and MPII-Validation set (Bottom). Results shown in average L1 distance between left and right bone in mm/3D pixels, respectively

Adding the geometric constraint, *i.e.*, **3D/w geo**, provides a decent performance gain.

Training with both 2D and 3D data (**3D+2D/wo geo**), provides significant performance gain — average MPJPE dropped to $64.90mm$, which is superior to all previous work [15, 19]. This verifies the effectiveness of combining data sources in our unified training.

Finally, the proposed approach **3D+2D/w geo** achieves the best results. Note that the constraints are applied on the disjoint 2D dataset, showing that the provided prior knowledge is universal. We have also tested adding constraints on fully-supervised 3D data. The results are similar.

Comparisons to other *in-the-wild* methods. Our method is superior to other methods that are applicable to *in-the-wild* images. Comparing to two two-step methods, MPJPE of Chen & Ramanan [5] is $114.18mm$ and MPJPE of Zhou et al. [35] is $79.9mm$. Pavlakos et al. [19] provided an alternative decoupled version which can also be applied in the wild, but its MPJPE increased to $78.1mm$. MPJPE of our method is $64.90mm$ and significantly better.

Why combining 2D and 3D data is better? A reasonable question is that it is still unclear whether the benefit of combined training comes from better depth estimation, or just from more accurate 2D pose estimation.

To answer this question, we only evaluate the accu-

racy of the 2D pose estimation, using the standard metric PCKh@0.5 (see [2]). The results in Tab. 2 show that the 2D pose is very accurate in all the three baselines and the proposed method. This convincingly indicates that adding 2D data into training *does not* improve the 2D accuracy but mostly benefits the the depth regression module via shared deep feature representation.

4.3. Transferred Human Pose In the Wild

We evaluate the generalization of our method on two datasets captured in different *in-the-wild* environments.

4.3.1 MPI-INF-3DHP Dataset

It exhibits considerable domain shift from both MPII and Human 3.6M datasets. Table 3 compares the performance of various methods on MPI-INF-3DHP. In this case, the first two baseline methods, *i.e.*, **3D/wo geo** and **3D/w geo**, have low performance. This is not surprising, as the 3D training set contains only indoor images. We note that even in this case, the geometric constraint is still effective (**3D/wo geo** is worse than **3D/w geo**).

3D+2D/wo geo achieved 65.8 and 32.1 in PCK and AUC, respectively. These numbers are better than their counterparts (64.7 PCK and 31.7 AUC) in [15] with Human 3.6M training data, again showing the advantage of our training scheme.

The proposed approach yields 69.2 in PCK and 32.5 in AUC. These numbers are close to the one that is derived from the original training data of MPI-INF-3DHP [15], which has 72.5 in PCK and 36.5 in AUC. Our result is strong even though we didn't use their training data. This confirms the ability of our method on *in-the-wild* images.

We also tested the left-right symmetry as described in Sec. 4.1.2. The results in Table. 4 (Bottom) shows that using the geometric constraint considerably improves the geometric validity.

4.3.2 MPII Validation Dataset

Finally, we evaluate our method on the most challenging *in-the-wild* MPII validation set. The qualitative 3D pose

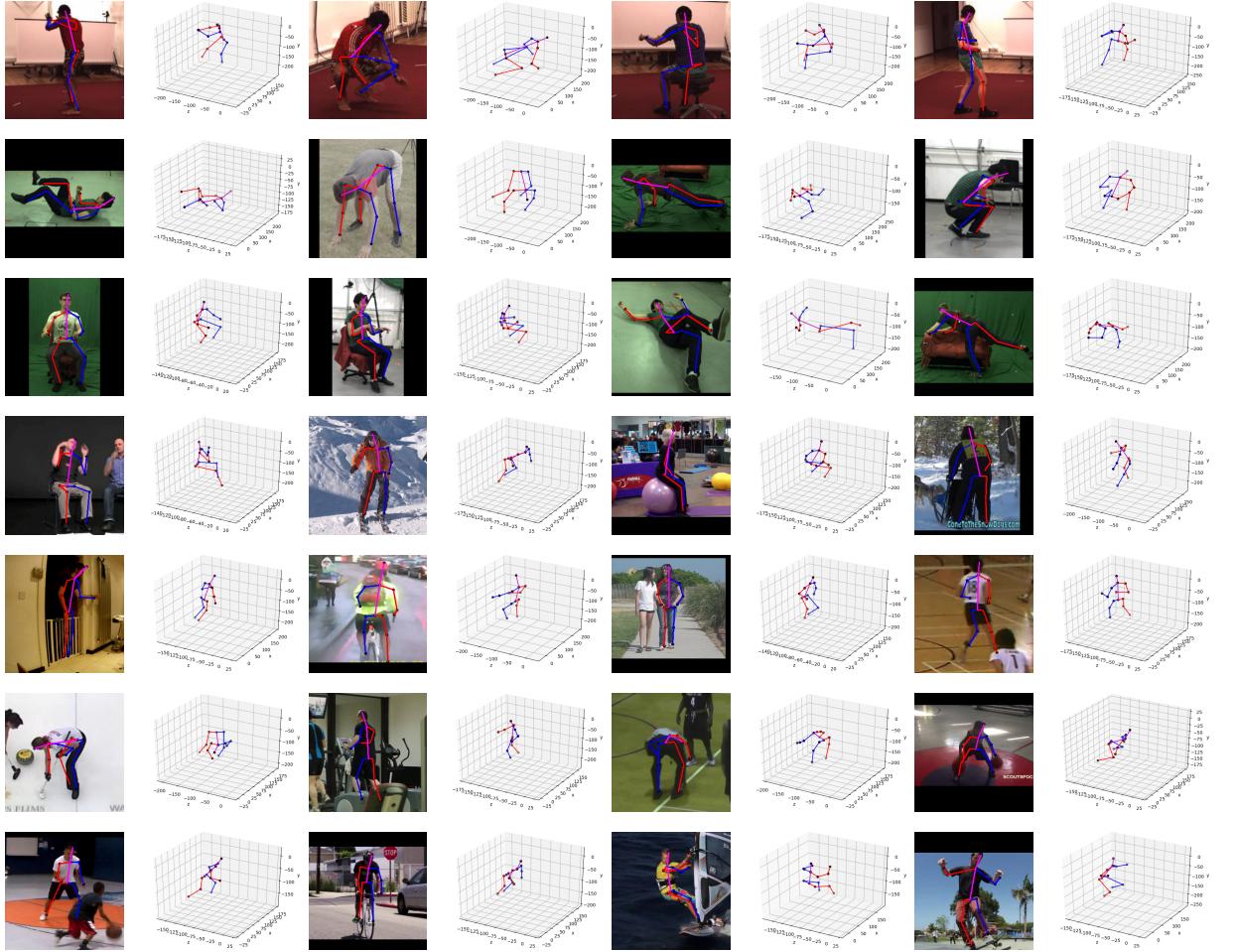


Table 5. Qualitative results from different datasets. We show the 2D pose on the original image and 3D pose from a novel view. First line: Human 3.6M dataset; Second and third lines: MPI-INF-3DHP dataset; Fourth to seventh lines: MPII dataset.

results in Table 5 are quite plausible.

Geometric validity. As explained in sec. 4.1.2, we evaluate the left-right symmetry metric. The results in Table 4 (Top) show that our approach is considerably better.

2D accuracy versus 3D accuracy. We note that our method has a slightly lower 2D joint accuracy than the original Hourglass model. This can be expected as our model learns the additional depth regression task. However, utilizing the geometric constraint improves the 2D joint accuracy as well. This indicates that our network is able to propagate this geometric constraint from the 3D module to the 2D module, which justifies the design goal of our network.

5. Future Work and Conclusions

In this paper, we introduced an end-to-end system that combines 2D pose labels in the wild and 3D pose labels in restricted environments for the challenge problem of 3D human pose estimation in the wild. In the future, we plan to

explore more un-/weakly-supervised constraints for a better transfer, e.g., a domain alignment network as in [10, 28]. We hope this work can inspire more works on un-/weakly-supervised transfer learning and on 3D human pose estimation in the wild.

Acknowledgements

We thank Dushyant Mehta and Helge Rhodin for helping about evaluating on MPI-INF-3DHP dataset and thank Danlu Chen for help with Fig. 2. Also, we thank Wei Zhang for helpful discussion. This work is supported in part by the National Natural Science Foundation of China (#U1611461, #61572138), Shanghai Municipal Science and Technology Commission (#16JC1420401).

References

- [1] 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
- [2] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 1, 5, 7
- [3] 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] A. Bulat and G. Tzimiropoulos. Human pose estimation via convolutional part heatmap regression. In *European Conference on Computer Vision*, pages 717–732. Springer, 2016. 1, 2
- [5] C.-H. Chen and D. Ramanan. 3d human pose estimation= 2d pose estimation+ matching. *arXiv preprint arXiv:1612.06524*, 2016. 2, 4, 6, 7
- [6] W. Chen, H. Wang, Y. Li, H. Su, Z. Wang, C. Tu, D. Lischinski, D. Cohen-Or, and B. Chen. Synthesizing training images for boosting human 3d pose estimation. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 479–488. IEEE, 2016. 3
- [7] X. Chu, W. Yang, W. Ouyang, C. Ma, A. L. Yuille, and X. Wang. Multi-context attention for human pose estimation. *arXiv preprint arXiv:1702.07432*, 2017. 1
- [8] R. Collobert, K. Kavukcuoglu, and C. Farabet. Torch7: A matlab-like environment for machine learning. In *BigLearn, NIPS Workshop*, 2011. 5
- [9] 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. 5
- [10] J. Hoffman, D. Wang, F. Yu, and T. Darrell. Fcns in the wild: Pixel-level adversarial and constraint-based adaptation. *arXiv preprint arXiv:1612.02649*, 2016. 3, 4, 8
- [11] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepcut: A deeper, stronger, and faster multi-person pose estimation model. In *European Conference on Computer Vision*, pages 34–50. Springer, 2016. 1, 2, 6
- [12] 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, jul 2014. 1, 2, 5, 6
- [13] 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, 2, 5
- [14] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. Smpl: A skinned multi-person linear model. *ACM Transactions on Graphics (TOG)*, 34(6):248, 2015. 2
- [15] D. Mehta, H. Rhodin, D. Casas, O. Sotnychenko, W. Xu, and C. Theobalt. Monocular 3d human pose estimation using transfer learning and improved cnn supervision. *arXiv preprint arXiv:1611.09813*, 2016. 2, 3, 5, 6, 7
- [16] T. B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer vision and image understanding*, 81(3):231–268, 2001. 2
- [17] 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, 2, 4, 5
- [18] D. Pathak, P. Krahenbuhl, and T. Darrell. Constrained convolutional neural networks for weakly supervised segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1796–1804, 2015. 3
- [19] G. Pavlakos, X. Zhou, K. G. Derpanis, and K. Daniilidis. Coarse-to-fine volumetric prediction for single-image 3d human pose. *arXiv preprint arXiv:1611.07828*, 2016. 2, 5, 6, 7
- [20] A.-I. Popa, M. Zanfir, and C. Sminchisescu. Deep multitask architecture for integrated 2d and 3d human sensing. *arXiv preprint arXiv:1701.08985*, 2017. 3
- [21] 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
- [22] G. Rogez and C. Schmid. Mocap-guided data augmentation for 3d pose estimation in the wild. In *Advances in Neural Information Processing Systems*, pages 3108–3116, 2016. 3
- [23] N. Sarafianos, B. Boteanu, B. Ionescu, and I. A. Kakadiaris. 3d human pose estimation: A review of the literature and analysis of covariates. *Computer Vision and Image Understanding*, 152:1–20, 2016. 2
- [24] L. Sigal, A. O. Balan, and M. J. Black. Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion. *International journal of computer vision*, 87(1-2):4, 2010. 1, 2
- [25] 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. 2
- [26] D. Tome, C. Russell, and L. Agapito. Lifting from the deep: Convolutional 3d pose estimation from a single image. *arXiv preprint arXiv:1701.00295*, 2017. 2, 4, 6
- [27] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Breuel. Efficient object localization using convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–656, 2015. 5
- [28] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015. 3, 4, 8
- [29] 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. 1, 2
- [30] J. Wu, T. Xue, J. J. Lim, Y. Tian, J. B. Tenenbaum, A. Torralba, and W. T. Freeman. Single image 3d interpreter network. In *European Conference on Computer Vision*, pages 365–382. Springer, 2016. 2
- [31] H. Yasin, U. Iqbal, B. Kruger, A. Weber, and J. Gall. A dual-source approach for 3d pose estimation from a single image.

- In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4948–4956, 2016. 2
- [32] X. Zhou, S. Leonardos, X. Hu, and K. Daniilidis. 3d shape estimation from 2d landmarks: A convex relaxation approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4447–4455, 2015. 2, 4
- [33] X. Zhou, X. Sun, W. Zhang, S. Liang, and Y. Wei. Deep kinematic pose regression. In *Computer Vision–ECCV 2016 Workshops*, pages 186–201. Springer, 2016. 1, 2, 5, 6
- [34] 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. 2, 4, 5
- [35] X. Zhou, M. Zhu, G. Pavlakos, S. Leonardos, K. G. Derpanis, and K. Daniilidis. Monocap: Monocular human motion capture using a cnn coupled with a geometric prior. *arXiv preprint arXiv:1701.02354*, 2017. 5, 6, 7