Not All Parts Are Created Equal: 3D Pose Estimation by Modeling Bi-directional Dependencies of Body Parts – *Supplementary Material* –

Jue Wang[1,2] Shaoli Huang*[1] Xinchao Wang[3] Dacheng Tao[1]

[1]UBTECH Sydney AI Centre, School of Computer Science, FEIT, University of Sydney, Darlington, NSW 2008, Australia

[2]University of Technology Sydney [3]Stevens Institute of Technology

jue.wang.0911@gmail.com, {shaoli.huang, dacheng.tao}@sydney.edu.au, xinchao.wang@stevens.edu

In this document, we provide materials that could not be included in the main manuscript due to the page limit. In Section 1, we provide additional ablation study results. In Section 2, we show more qualitative results on Human3.6M, MPI-INF-3DHP and MPII, to further demonstrate the domain transfer capability of our method. More 3D results are presented in the attached video. In Section 3, we provide the 3D pose estimation results from a given 2D pose but conditioned on different pose attributes, to investigate how the pose attributes affect 3D pose estimation.

1. Additional ablation studies

In what follows, we show the 3D pose estimation results by first varying the number of attribute categories and then removing the domain adaptation component.

1.1. Influence of the number of attribute categories

In the main manuscript, we defined the three-category pose attributes, *front*, *on-plane*, and *back*. In fact, nothing prevents us from dividing the attributes into more categories, for example, one category every 100mm between the joint and the torso. However, it might be not a good idea to define a large number of categories, since the difficulty of predicting attributes will also tend to increase as the number of categories increases.

To see the influences of the category number on the performance, we first review our definition of attribute categorization. Let d_i denote the Euclidean distance of joint i to the torso reference plane. The three-category attribute is defined as follows,

$$\text{attribute}_i = \begin{cases} 0 & \text{if } d_i < -\tau, \\ 1 & \text{if } |d_i| \leq \tau, \\ 2 & \text{if } d_i > \tau, \end{cases} \quad (1)$$

where τ is the offset used to distinguish *front*, *on-plane*, and *back*. The five-category, seven-category and nine-category attributes are defined in a similar way.

#Class	Error (Attr. GT)	Acc. of Attr.	Error (Attr. Pred.)
3	51.3	84.0%	52.6
5	49.7	75.9%	53.4
7	48.7	72.4%	53.7
9	48.3	66.3%	54.6

Table 1. The performances of our method at different numbers of attribute classes under Protocol #1.

We then test our method under these different settings and show the results in Tab. 1. From the table we can see that, when using GT attributes, the larger the number of attribute categories is, the smaller the 3D pose prediction error is. But when using predicted attributes, the accuracy of attribute prediction decreases with the increase of the number of attribute categories. The same trend is observed on the 3D estimation performance.

1.2. Effect of domain adaptation

We also conduct experiments by turning off the domain adaptation component to see its effect. The comparative results are shown in Tab. 2 below.

Method	H36M	3DPCK	AUC
Ours/wo DA	54.4	67.60	34.6
Ours	52.6	71.9	35.8

Table 2. The performance of our method with and without the domain adaptation technique. H36M denote the result on Human3.6M under Protocol #1, while 3DPCK and AUC denote the ones on MPI-INF-3DHP.

2. Additional qualitative results

We show here more qualitative results on the Human3.6M dataset and on the in-the-wild images.

2.1. More qualitative results on Human3.6M

In the attached video, we show the 3D pose estimation results of several actions from different camera view at

*Corresponding author

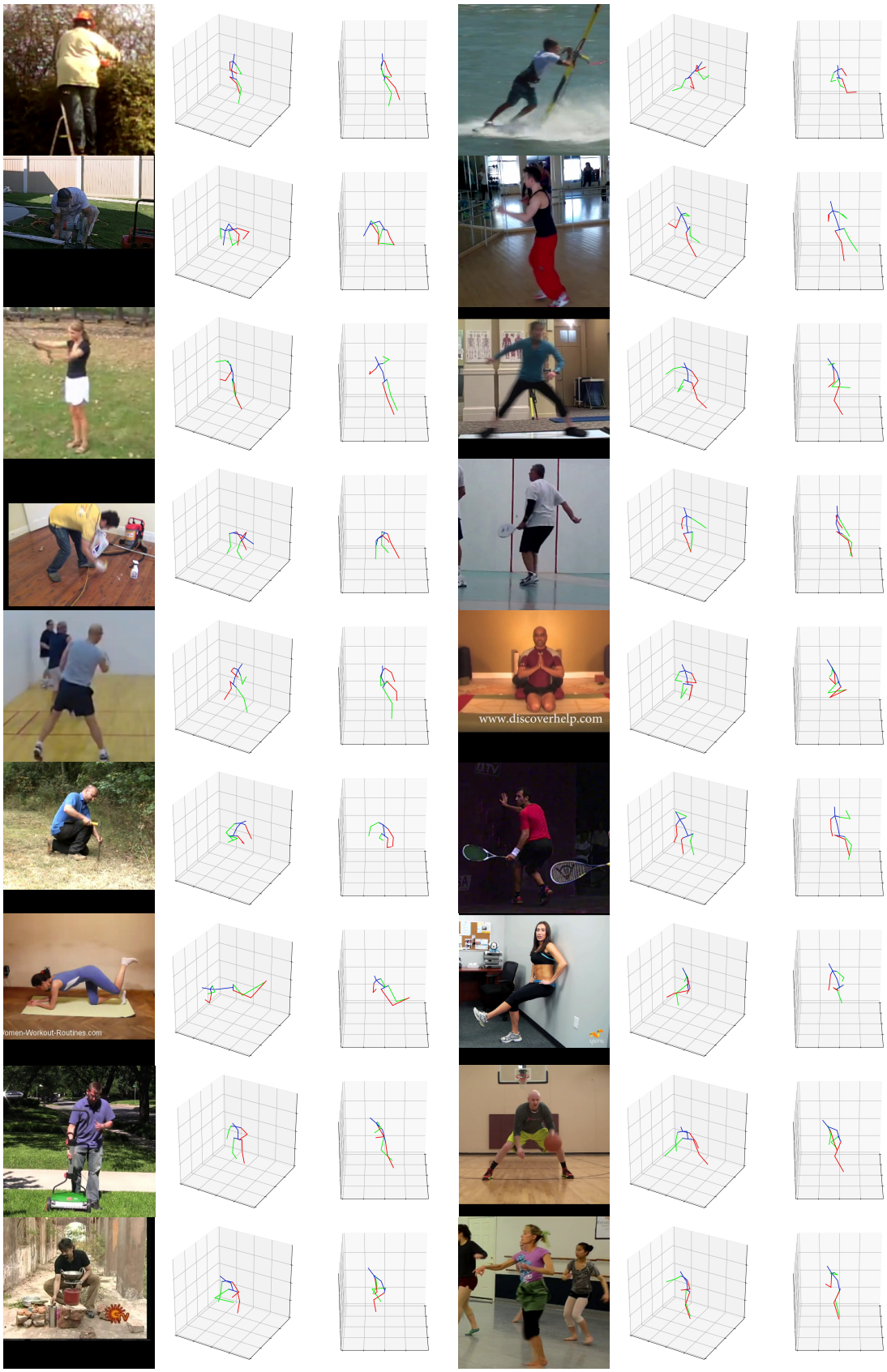

Figure 1. Qualitative results on MPII from two different views.

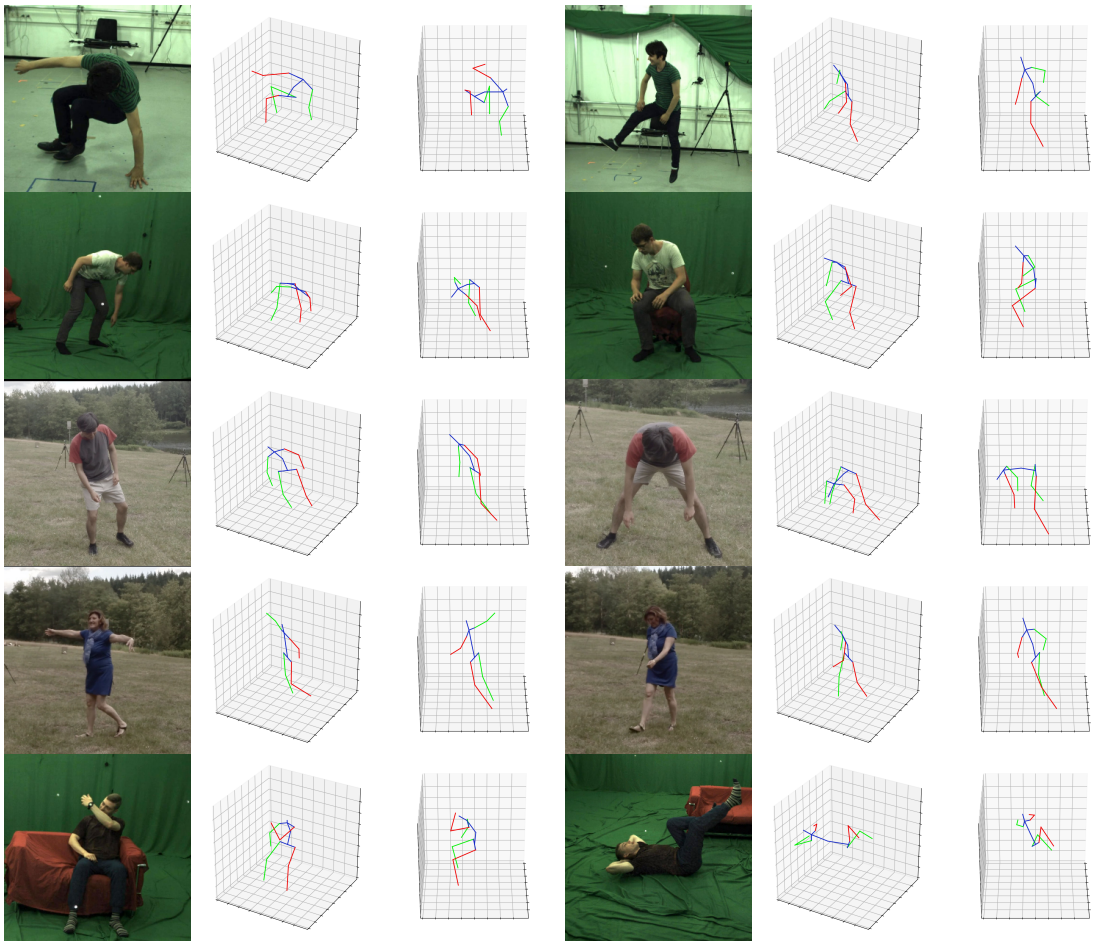

Figure 2. Qualitative results on MPI-INF-3DHP from two different views.

10fps, without using any smoothing techniques. Our results are very close to GT, and unexpectedly, are temporally-smooth in most of the frames.

2.2. Additional qualitative results on in-the-wild images

Additional qualitative results on MPII and MPI-INF-3DHP are shown in Figs. 1 and 2. More examples from these two datasets are provided in the attached video. Our method still performs well in some seemingly-challenging scenarios including complex outdoor environments, blurred images, and unseen poses.

3. 3D pose estimation conditioned on pose attributes

To demonstrate the influence of the proposed pose attributes on 3D pose estimation, we show in Figs. 3 and 4 the 3D pose estimation results from the same 2D pose but conditioned on different attributes settings. In other words, given the same 2D pose estimations, we manually set the pose attributes to different categories, and feed them as input for 3D pose estimation to see the results.

In most cases, changing the attribute on a joint will lead to a change of the 3D predictions of the corresponding joint, as can be seen from rows 1 and 2 of Fig. 3, indicating that the pose attributes are effective and interpretable priors for 3D pose estimation. Although our pose attributes are defined using the 3D joint locations, in a proper camera view, it is possible to infer the attribute of a particular joint from the 2D pose. In these cases, setting the attributes to some improper values may bring conflicts between the 2D pose and the attributes. When the attribute setting of a joint conflicts with the 2D pose, our method is able to make a compromise on them. For example, in Fig. 3, we can tell from the 2D pose that the right hand is in front of the torso, which means, for example, the attributes for r-Elbow should be *Front*. When setting its attribute to *Back*, the 3D location of this joint only moves back slightly, but still lies in front (see row 3 in Fig. 4). In addition, it also implies that combining the 2D pose with image features may facilitate the learning of pose attributes, which could be an interesting future work.

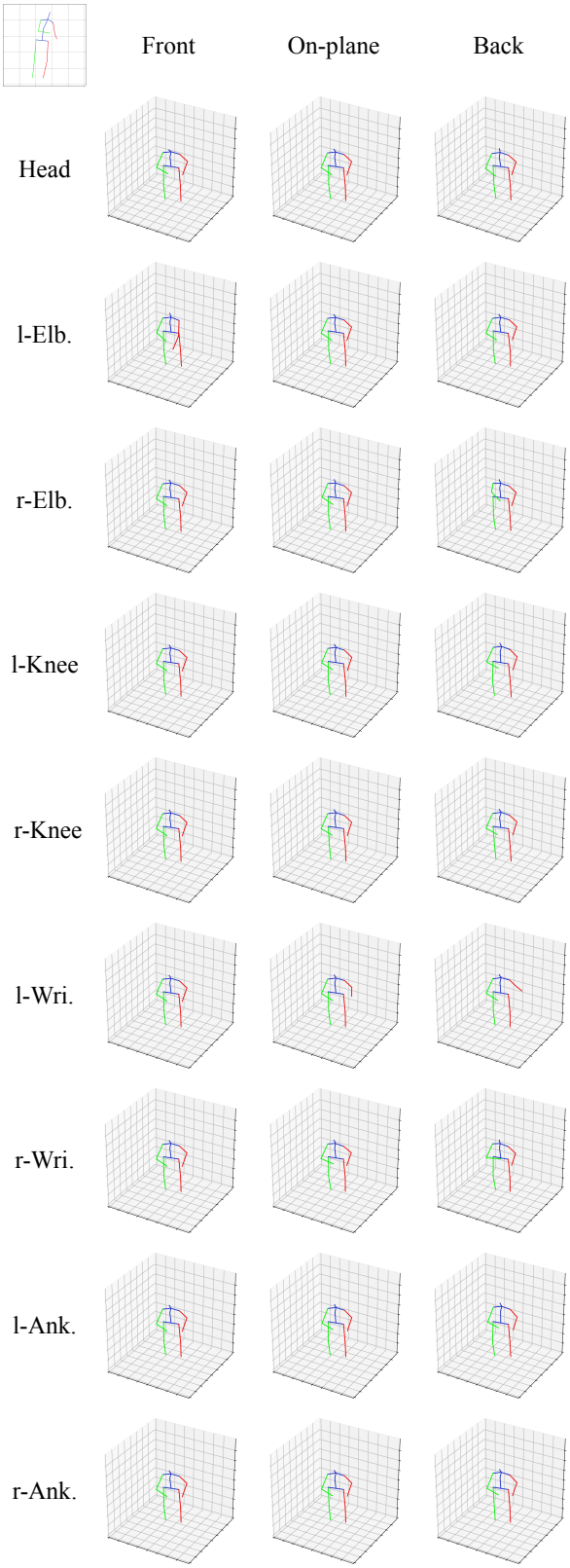

Figure 3. 3D pose estimation from a single 2D pose but conditioned on different pose attribute settings (case #1).

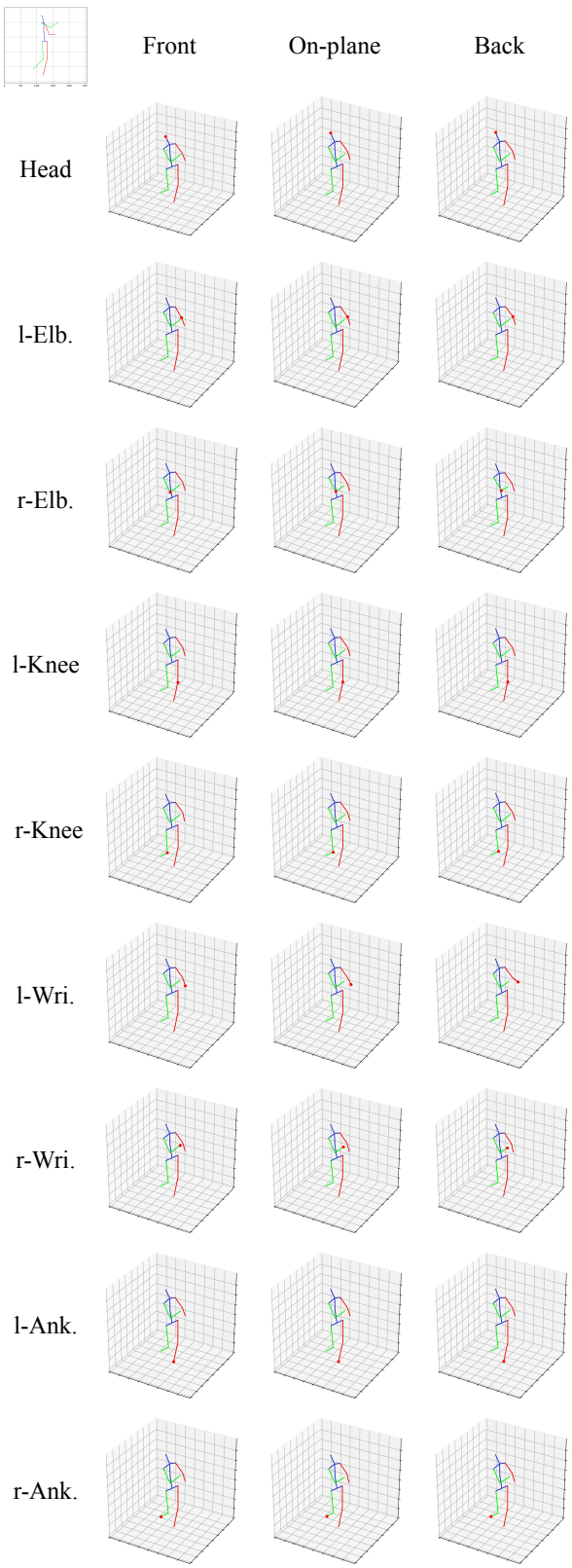

Figure 4. 3D pose estimation from a single 2D pose but conditioned on different pose attribute settings (case #2).