Geometry-Driven Self-Supervised Method for 3D Human Pose Estimation

Yang Li,1,2 Kan Li,1* Shuai Jiang,1,2 Ziyue Zhang,2 Congzhentao Huang,2 Richard Yi Da Xu2

1School of Computer Science and Technology, Beijing Institute of Technology, China
2Faculty of Engineering and Information Technology, University of Technology Sydney, Australia
{yanglee, likan}@bit.edu.cn, {shuai.jiang-1, ziyue.zhang-2, congzhentao.huang}@student.uts.edu.au, yida.xu@uts.edu.au

Abstract

The neural network based approach for 3D human pose estimation from monocular images has attracted growing interest. However, annotating 3D poses is a labor-intensive and expensive process. In this paper, we propose a novel self-supervised approach to avoid the need of manual annotations. Different from existing weakly/self-supervised methods that require extra unpaired 3D ground-truth data to alleviate the depth ambiguity problem, our method trains the network only relying on geometric knowledge without any additional 3D pose annotations. The proposed method follows the two-stage pipeline: 2D pose estimation and 2D-to-3D pose lifting. We design the transform re-projection loss that is an effective way to explore multi-view consistency for training the 2D-to-3D lifting network. Besides, we adopt the confidences of 2D joints to integrate losses from different views to alleviate the influence of noises caused by the self-occlusion problem. Finally, we design a two-branch training architecture, which helps to preserve the scale information of re-projected 2D poses during training, resulting in accurate 3D pose predictions. We demonstrate the effectiveness of our method on two popular 3D human pose datasets, Human3.6M and MPI-INF-3DHP. The results show that our method significantly outperforms recent weakly/self-supervised approaches.

Introduction

3D human pose estimation has attracted substantial interest for its vast potential on various applications including human-computer interaction, virtual reality and action recognition. With the great success of deep learning, many researchers (Martinez et al. 2017; Pavllo et al. 2019) applied the neural network to predict 3D human poses from monocular images. Estimating 3D poses using neural networks mainly faces two main challenges. Firstly, a typical neural network model needs a large amount of training data. 3D pose annotations are collected through the marker-based Motion Capture (MoCap) system, which is an expensive process. Secondly, there are well-founded geometrical theories on how to project 2D images to 3D skeletons. Simply using a neural network to approximate this projection may lead to the network subject to overfitting training data.

To alleviate the above challenges, weakly/self-supervised learning paradigms have been increasingly explored in recent works (Rhodin et al. 2018; Wu et al. 2016; Chen et al. 2019; Rhodin, Salzmann, and Fua 2018). Re-projection loss (Tung et al. 2017), which does not require explicit 3D ground-truth, has become a commonly used technique. It reprojects estimated 3D poses back to the 2D space and calculates the loss between the input and re-projected 2D poses as supervision. However, due to the depth ambiguity problem where multiple 3D body configurations can explain the same 2D projection, the re-projection loss cannot yield accurate and realistic 3D poses. For example, as shown in Figure 1, since re-projection loss only constrains the estimated 3D pose at a specific camera angle, it may result in an invalid human pose when observed from another angle. Although some techniques such as adversarial loss (Tung et al. 2017; Wandt and Rosenhahn 2019) and kinematic constraints (Habibie et al. 2019; Pavllo et al. 2019), have been proposed to constrain the estimated 3D poses into a semantic sub-space, they usually require some extra unpaired 3D pose annotations.

Figure 1: (a) The network trained with re-projection loss. (b) The proposed network trained with transform re-projection loss. (c) The comparisons of results estimated by the above two networks. The results are shown in two different views.
The contributions of our work are summarized as follows:

- We design the transform re-projection loss, which is an effective technique to exploit multi-view consistency information and constrain the estimated 3D poses during training. Moreover, we integrate it with the 2D joint confidences of different camera views to alleviate the self-occlusion problem.
- The proposed method achieves state-of-the-art results on two popular 3D pose benchmarks compared with recent weakly/self-supervised methods.

Related Work

3D Human Pose Estimation

3D human pose estimation is a long-standing problem and has been considerably studied in the past few years. Recently, following the great success of deep learning, modern 3D human pose estimation techniques are usually formulated as learning-based frameworks. These works can be generally classified into two categories. The first class of methods (Tekin et al. 2017; Pavlakos et al. 2017a; Mehta et al. 2017; Habibie et al. 2019; Sun et al. 2018) directly predict the depth from monocular images through the deep convolutional neural networks (DCNNs). The second category (Iqbal et al. 2018; Fang et al. 2018; Tome, Russell, and Agapito 2017; Martinez et al. 2017; Fang et al. 2018; Chen and Ramanan 2017) is the two-stage pipeline, which first obtains 2D joint locations through the advanced 2D keypoint detector such as Stacked Hourglass (SH) network (Newell, Yang, and Deng 2016) and Cascaded Pyramid Network (CPN) (Chen et al. 2018), and then lifts them into 3D space. In order to learn the mapping between 2D and 3D joint positions, various 2D-to-3D lifting network backbones were designed. For example, (Martinez et al. 2017) proposed a simple baseline using a simple neural network with only two fully connected layers, while achieved surprising results. Since human skeletons are with the graph-like structure, several works (Zhao et al. 2019) also attempted to exploit the novel Graph Convolution Networks (GCNs) to capture the semantic relationships between human joints for accurate 3D human pose regression. Besides, there are also works (Pavllo et al. 2019; Arnab, Doersch, and Zisserman 2019; Zhou and De la Torre 2016) that considers temporal information from frame sequence to produce more robust predictions. In our work, we follow the two-stage pipeline. Moreover the proposed approach is compatible with any recent 2D-to-3D lifting network backbone.

Weakly/Self-supervised Approaches

Recently, weakly/self-supervised approaches have received much attention due to the difficulty of gathering 3D pose annotations. In order to train the network without explicit 3D pose annotations, the prior of camera projection geometry was commonly explored, and some geometry-driven methods were proposed. Among them, re-projection loss is one of the most widely used technique (Kanazawa et al. 2018; Wu et al. 2016; Wandt and Rosenhahn 2019; Pavllo et al. 2019; Wang et al. 2019; Brau and Jiang 2016). However, using re-projection loss alone cannot accurately con-
strain the depth of skeletons due to the depth ambiguity problem. (Habibie et al. 2019; Pavllo et al. 2019; Rhodin et al. 2018) alleviated this problem by enforcing the bone length similarity between predicted and ground-truth skeletons. Adversarial loss (Yang et al. 2018; Tung et al. 2017; Wandt and Rosenhahn 2019; Kanazawa et al. 2018) is another popular technique to regularize the predicted 3D poses. It encourages the output 3D poses on the real human manifold by introducing a real/fake 3D skeleton discriminator. For example, (Tung et al. 2017) proposed the Adversarial Inverse Graphical Network (AIGN), which uses the adversarial prior to match the distribution between the predictions and a collection ground-truth for the task such as 2D-to-3D lifting and image-to-image translation. (Wandt and Rosenhahn 2019) proposed a weakly-supervised method with the adversarial supervision for 3D human pose estimation. (Chen et al. 2019) exploits the geometric self-consistency of the lift-reproject-lift process with the adversarial prior of 2D poses. As we analyzed above, the bone length constraint and adversarial loss still require unpaired 3D pose annotations for counting the bone length or training the real/fake 3D skeleton discriminator. Differently, we proposed a self-supervised approach that solely relies on the camera geometry prior, which can result in better generalization ability.

Multi-view Approaches

Early methods (Belagiannis et al. 2014) reconstructed 3D poses from multi-view inputs through the triangulation or the 3D pictorial structures model. Recent works (Pavlakos et al. 2017b; Tome et al. 2018; Dong et al. 2019) combined the traditional techniques with the novel CNNs to improve the robustness of the framework. Different from these methods requiring multi-view inputs at both training and testing stages, our method only requires multi-view inputs during training and requires monocular images during testing.

There are two latest methods that also exploit multi-view information for 3D pose estimation in a self-supervised way. (Rhodin, Salzmann, and Fua 2018) pre-trained an encoder-decoder network that predicts an image from one view to another to learn a geometry-aware body representation, and then use a small amount of supervision to learn a mapping from 2D to 3D poses. (Kocabas, Karagoz, and Akbas 2019) applied triangulation on the detected 2D joint positions to generate ‘ground-truth’ 3D poses for training the 3D pose estimation network. In comparisons, our method directly adopts the multi-view consistency information for training rather than generating the pseudo 3D ground-truth, which is a simpler way and more robust to the noisy 2D detections. Moreover, our method does not require any additional 3D annotations.

Method

Overall, the proposed method follows a two-stage pipeline, as shown in Figure 2. First, we use the state-of-the-art 2D pose estimation network to predict 2D poses from input frames. Here, we denote \( \mathbf{X} \in \mathbb{R}^{N \times 2} \) as \( N \) detected 2D joint locations. Meanwhile, we obtain their corresponding confidence scores \( \mathbf{w} \in \mathbb{R}^{N} \) from estimated keypoint heatmaps through the \( \max \) operation. With the detected 2D poses, we learn a neural network \( \mathcal{N} \) to project them into 3D space. Similarly, we define \( \mathbf{Y} \in \mathbb{R}^{N \times 3} \) as the output 3D joint locations. Following the protocol with previous works, we estimate zero-centered 3D poses where the values of \( \mathbf{Y} \) are the 3D positions relative to the fixed root joint (pelvis).

The architecture of the lifting network \( \mathcal{N} \) is designed inspired by (Martinez et al. 2017). The input layer takes the concatenated coordinates of \( N \) human joints and applies a fully connected layer with 1024 output channels. Then it is followed by four blocks that are surrounded by residual connections. For each block, several fully connected layers (1024 channels) followed by Batch Normalization, rectified linear units, and dropout, are stacked for efficiently mapping the 2D pose features to high-level features. Finally, the features extracted by the last residual block are fed into an extra linear layer \( (N \times 3 \text{ channels}) \) to output 3D poses.

Self-supervised Approach

In this section, we introduce the proposed self-supervised approach for training the lifting network. The training process takes as input the detected 2D poses of a pair of frames that are captured from two different views at the same time. With the paired frames, we first detect their 2D poses \( \mathbf{X}^{v1} \) and \( \mathbf{X}^{v2} \) and their corresponding confidence weights of each joint \( \mathbf{w}^{v1} \) and \( \mathbf{w}^{v2} \). Then, we feed the 2D poses into the lifting network and obtain their estimated 3D poses \( \mathbf{Y}^{v1} \) and \( \mathbf{Y}^{v2} \).
Two-branch Training Architecture  

For training the lifting network without 3D ground-truth annotations, we design the transform re-projection loss. It involves the perspective projection and view transformation operations, which require global 3D joint positions. Without global 3D joint positions, we can not obtain the absolute depth of the person in the camera coordinate, which results in unknown scale when re-projecting 3D poses back to 2D space. Existing methods commonly normalize the scale of 2D skeletons to overcome the scale ambiguity problem. However, it must be used in conjunction with the kinematic constraint or adversarial loss to output realistic 3D poses.

In our work, we design another branch, named root position branch, to help train the lifting network. It predicts root joint positions, $r^{v1}$ and $r^{v2}$, which are added to relative 3D poses predicted by the lifting network to restore global 3D poses, $Y^{v1}$ and $Y^{v2}$. The root position network has the same architecture with the lifting network, and they do not share poses, predicted by the lifting network to restore global 3D joint positions, we can not obtain the absolute depth of the person in the camera coordinate, which results in unknown scale when re-projecting 3DSkeleton should be the same with the 2D input of the target view, we design the transform re-projection loss as follows:

$$L_{t-reproj} = \sum_{i} w_{i}^{v1} \| X^{v1} - \rho(Y^{v1}) \|^2 + w_{i}^{v2} \| X^{v2} - \rho(Y^{v2}) \|^2,$$

where $R_{1to2}, R_{2to1} \in \mathbb{R}^{3 \times 3}$ are the rotation matrixes, and $t_{1to2}, t_{2to1} \in \mathbb{R}^{3}$ are the transformation vectors. With the extrinsic parameters of two cameras $R_{1}, t_{1}$ and $R_{2}, t_{2}$, we can directly obtain the rigid transformation parameters,

$$R_{1to2} = R_{2} R_{1}^{T}; \quad t_{1to2} = R_{1} (t_{2} - t_{1}),$$

where $r^{v1}$ and $r^{v2}$ are the confidence scores of the $i^{th}$ joints of two views. Here, we use the confidence scores of detected 2D poses to integrate the re-projection loss of different views. The view with smaller 2D confidence value makes less contribution to the loss value, which reduces the impact of the noisy 2D detections for the lifting network training.

However, simply using the re-projection consistency will encounter the depth ambiguity problem. To overcome the problem, we design the transform re-projection loss, which constrains the predicted 3D skeletons from multiple perspectives. Specifically, we transform the estimated 3D pose from one view to another through the rigid transformation $\tau$ as follows:

$$\tau(Y^{v1}) = R_{1to2} (Y^{v1} - t_{1to2}),$$

$$\tau(Y^{v2}) = R_{2to1} (Y^{v2} - t_{2to1}),$$

For training the lifting network without 3D ground-truth annotations, we find that the
network cannot converge if we train it with random initialization. Thus, we design a pre-training technique to warm-up the network. As shown in Figure 4, the pre-training loss can be formulated as:

\[ L_{\text{pre-train}} = \sum_{i} \| \tau(\tilde{Y}_i^1) - \tilde{Y}_i^2 \|^2 + \| \tau(\tilde{Y}_i^2) - \tilde{Y}_i^1 \|^2. \]  

(6)

It is designed according to multi-view consistency that the transformed 3D pose and the estimated 3D pose of the target view should be the same. Although this loss is not able to guide the lifting network to produce valid 3D poses, it effectively regularizes the output space of the root position branch. It can be regarded as an advanced initialization of the root position branch, which greatly reduces the difficulty of network convergence.

After pre-training, the network is fine-tuned using the re-projection loss and transform re-projection loss,

\[ L_T = L_{\text{reproj}} + \lambda L_{\text{t-reproj}}, \]

where \( \lambda \) is a hyper-parameter that is adapted to set under different datasets.

**Experiments**

**Datasets**

We perform extensive evaluations on two publicly available benchmarks. Human3.6M (H36M) (Ionescu et al. 2013) is one of the largest datasets for 3D human pose estimation, which is captured by MoCap system. It consists of 3.6 million images with 11 actors performing 15 actions such as eating, sitting and walking. They are captured from 4 calibrated camera views. Training with more camera views can improve model performance and generalization ability. We follow the technique proposed by (Fang et al. 2018) to simulate a series of virtual camera views. We extend the H36M dataset from 4 views to 12 views containing 8 virtual camera views, and we obtain the corresponding 2D pose of each sample through perspective projection to augment the training set. The detailed analysis will be shown in the following sections.

**Evaluation Metrics**

For the H36M dataset, we consider two popular evaluation protocols. Protocol 1 is the Mean Per Joint Position Error (MPJPE) in millimeters (mm). MPJPE is the mean euclidean distance between the ground-truth and predicted positions of the joints. Protocol 2 is the Procrustes MPJPE (P-MPJPE), which aligns the estimated 3D pose to the ground-truth by a rigid transformation called Procrustes Analysis before computing the MPJPE.

The evaluation metrics for the 3DHP dataset include the adapted Percentage of Correct Keypoints (PCK) and corresponding Area Under Curve (AUC) (Mehta et al. 2017). The PCK indicates the percentage of joints whose estimated position is within 15cm of the ground-truth.

**Data Augmentation**

The H36M dataset has only four calibrated camera views. Training with more camera views can improve model performance and generalization ability. We follow the technique proposed by (Fang et al. 2018) to simulate a series of virtual camera views. We extend the H36M dataset from 4 views to 12 views containing 8 virtual camera views, and we obtain the corresponding 2D pose of each sample through perspective projection to augment the training set. The detailed analysis will be shown in the following sections.

**Implementation Details**

In order to enable the proposed two-branch network to converge without any explicit 3D pose supervision, the training procedure contains two stages. First, we pre-train the network using the \( L_{\text{pre-train}} \) loss. We use the Adam as the optimizer and train the network for 200 epochs with learning rate 0.001. Next, the network is trained using the \( L_T \) loss for 300 epochs. The learning rate starts from 0.001 and drops by 0.1 each 100 epochs. During evaluation, for consistency with other works, we only use the 2D-to-3D lifting branch to predict the relative 3D poses in the camera space, and not use the root position branch. We implement our method using the deep learning toolbox Pytorch.
Ablation Study

Analysis of Transform Re-projection Loss  In order to evaluate the effectiveness of the proposed transform re-projection loss, we compare it with the existing popular technique, adversarial loss. We design several variants and compare the results under Protocol #1 (MPJPE) and Protocol #2 (P-MPJPE) on the H36M dataset. All variants use 2D poses extracted by the CPN network as inputs. Table 1 presents the quantitative results, and Figure 5 shows the results of different variants on several hard samples, i.e., with serious self-occlusion or far from the camera. It is obvious that only using the re-projection loss will obtain strange 3D skeletons that do not conform the human kinematics. Although adversarial loss can constrain the 3D poses using unpaired 3D pose annotations, it still cannot produce precise 3D poses, especially when encountering samples with serious self-occlusion. Compared with adversarial loss, our method achieves significant performance improvements, and the MPJPE and P-MPJPE decrease by 47.6 and 32.7 (mm). This shows that the transform re-projection loss can effectively help the network learn geometric knowledge, which further constrains the estimated 3D poses to get more accurate results. Moreover, the MPJPE and P-MPJPE decrease by extra 2.0 and 1.6 (mm) when using data augmentation, which verifies that training with more camera views can effectively facilitate the model performance.

Analysis of Backbones  Our method does not depend on any particular backbone. In this part, we investigate the performance of our method with different 2D-to-3D network backbones. ResLinear (Martinez et al. 2017) is the earliest and most commonly used backbone, which consists of fully connected layers and residual connections. TemporalDilated (Pavllo et al. 2019) is the latest proposed backbone that can explore the temporal information using dilated temporal convolutions. We feed it with 243 neighboring frames as inputs during training and testing. As shown in Table 2, our approach can achieve competitive results when using the simple ResLinear backbone. Therefore, the improvements of our method are not merely due to the better backbone. When using the TemporalDilated, our method gains obvious improvements, which benefit from the exploration of temporal information. These results illustrate that the proposed self-supervised training technique has strong versatility and is suitable for any novel 2D-to-3D network architecture.

Analysis of Network Pre-training  In this section, we show the effectiveness of the network pre-training. Since the proposed network is trained without any 3D pose annotation, the pre-training is very important for our two-branch networks. As shown in Figure 6, the loss and MPJPE curves without pre-training violently oscillate. The network fails to converge despite our best efforts at tuning the hyperparameters. In contrast, the loss curve rapidly decreases, and we achieve low MPJPE value when using the proposed pre-training technique. It illustrates that pre-training technique is vital and effective in our work.

Analysis of Generalization Ability  To demonstrate the generalization ability of our model, we train the network on the H36M dataset and evaluate it on the test split of the 3DHP dataset, which includes challenging outdoor scenes. We present some examples in Figure 7. It shows that our approach can successfully recover 3D poses on the datasets without being trained on them.

Comparisons with State-of-the-art Methods  In this section, we compare our method with recent weakly/self-supervised methods. First, we compare with them on the H36M dataset using protocol #1 and protocol #2 in Table 3. (Tung et al. 2017; Wandt and Rosenhahn 2019; Zhou et al. 2017) are based on re-projection loss and require additional unpaired 3D pose annotations. Compared with them, our method has an explicit improvement and obtains average errors of 59.0mm and 45.7mm under two evaluation protocols. Our method also outperforms (Kocabas, Karagoz, and Akbas 2019) that adopts multi-view information. We present its result obtained in the case of using
Table 3: Comparisons with recent weakly/self-supervised methods on the H36M dataset under evaluation Protocol #1 and Protocol #2.

<table>
<thead>
<tr>
<th>Protocol #1</th>
<th>Direct.</th>
<th>Discuss</th>
<th>Eating</th>
<th>Greet</th>
<th>Phone</th>
<th>Photo</th>
<th>Pose</th>
<th>Purch.</th>
<th>Sitting</th>
<th>SittingD</th>
<th>Smoke</th>
<th>Wait</th>
<th>WalkD</th>
<th>Walk</th>
<th>WalkT</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pavlakos et al. CVPR’17</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>118.4</td>
<td></td>
</tr>
<tr>
<td>Tung et al. ICCV’17 (†)</td>
<td>77.6</td>
<td>91.4</td>
<td>89.9</td>
<td>88</td>
<td>107.3</td>
<td>110.1</td>
<td>75.9</td>
<td>107.5</td>
<td>124.2</td>
<td>137.8</td>
<td>102.2</td>
<td>90.3</td>
<td>-</td>
<td>-</td>
<td>97.2</td>
<td></td>
</tr>
<tr>
<td>Wandt et al. CVPR’19 (†)</td>
<td>77.5</td>
<td>85.2</td>
<td>82.7</td>
<td>93.9</td>
<td>93.9</td>
<td>101.0</td>
<td>82.9</td>
<td>102.6</td>
<td>100.5</td>
<td>125.8</td>
<td>88.0</td>
<td>84.8</td>
<td>72.6</td>
<td>78.8</td>
<td>79.0</td>
<td>89.9</td>
</tr>
<tr>
<td>Wang et al. PAMI’19 (⋆; †)</td>
<td>50.0</td>
<td>60.0</td>
<td>54.7</td>
<td>56.6</td>
<td>65.7</td>
<td>52.7</td>
<td>54.8</td>
<td>85.9</td>
<td>118.0</td>
<td>62.5</td>
<td>79.6</td>
<td>59.6</td>
<td>41.5</td>
<td>65.2</td>
<td>48.5</td>
<td>63.7</td>
</tr>
<tr>
<td>Kocabas et al. CVPR’19</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>76.6</td>
</tr>
<tr>
<td>Ours</td>
<td>49.7</td>
<td>54.5</td>
<td>58.0</td>
<td>56.8</td>
<td>63.4</td>
<td>80.0</td>
<td>52.4</td>
<td>52.7</td>
<td>71.4</td>
<td>78.3</td>
<td>58.9</td>
<td>55.2</td>
<td>60.0</td>
<td>43.8</td>
<td>49.6</td>
<td>59.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Protocol #2</th>
<th>Direct.</th>
<th>Discuss</th>
<th>Eating</th>
<th>Greet</th>
<th>Phone</th>
<th>Photo</th>
<th>Pose</th>
<th>Purch.</th>
<th>Sitting</th>
<th>SittingD</th>
<th>Smoke</th>
<th>Wait</th>
<th>WalkD</th>
<th>Walk</th>
<th>WalkT</th>
<th>Avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhou et al. ICCV’17 (†)</td>
<td>54.8</td>
<td>60.7</td>
<td>58.2</td>
<td>71.4</td>
<td>62.0</td>
<td>65.5</td>
<td>53.8</td>
<td>55.6</td>
<td>75.2</td>
<td>111.6</td>
<td>64.1</td>
<td>66.0</td>
<td>5.4</td>
<td>63.2</td>
<td>55.3</td>
<td>64.9</td>
</tr>
<tr>
<td>Drover et al. ECCV’18</td>
<td>60.2</td>
<td>60.7</td>
<td>59.2</td>
<td>65.1</td>
<td>65.5</td>
<td>63.8</td>
<td>59.4</td>
<td>59.4</td>
<td>69.1</td>
<td>88.0</td>
<td>64.8</td>
<td>60.8</td>
<td>64.9</td>
<td>63.9</td>
<td>65.2</td>
<td>64.6</td>
</tr>
<tr>
<td>Rhodin et al. ECCV’18 (†)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>98.2</td>
</tr>
<tr>
<td>Wandt et al. CVPR’19 (†)</td>
<td>53.0</td>
<td>58.3</td>
<td>59.6</td>
<td>66.5</td>
<td>72.8</td>
<td>71.0</td>
<td>56.7</td>
<td>69.6</td>
<td>78.3</td>
<td>95.2</td>
<td>66.6</td>
<td>65.9</td>
<td>63.4</td>
<td>63.9</td>
<td>49.9</td>
<td>65.1</td>
</tr>
<tr>
<td>Kocabas et al. CVPR’19</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>67.5</td>
</tr>
<tr>
<td>Chen et al. CVPR’19 (⋆)</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>68.0</td>
</tr>
<tr>
<td>Ours</td>
<td>39.6</td>
<td>42.6</td>
<td>45.7</td>
<td>46.0</td>
<td>47.6</td>
<td>57.1</td>
<td>41.0</td>
<td>39.2</td>
<td>55.4</td>
<td>59.9</td>
<td>46.4</td>
<td>42.5</td>
<td>47.1</td>
<td>34.4</td>
<td>41.0</td>
<td>45.7</td>
</tr>
</tbody>
</table>

1 (⋆) denotes that it takes advantage of the temporal information; (†) denotes that it requires unpaired or part of 3D pose annotations.

Table 4: Comparisons with recent weakly/self-supervised methods on the 3DHP dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>PCK</th>
<th>AUC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Zhou et al. ICCV’17</td>
<td>69.2</td>
<td>32.5</td>
</tr>
<tr>
<td>Kocabas et al. CVPR’19</td>
<td>64.7</td>
<td>-</td>
</tr>
<tr>
<td>Chen et al. CVPR’19</td>
<td>71.1</td>
<td>36.3</td>
</tr>
<tr>
<td>Ours</td>
<td>74.1</td>
<td>41.4</td>
</tr>
</tbody>
</table>

In this work, we proposed a new self-supervised approach for 3D human pose estimation. The approach explored multi-view consistency to construct supervision signals for training a 2D-to-3D lifting network, which can effectively overcome the depth ambiguity problem. Note that our method simply applied multi-view information during training, and required only single view inputs during inference. Meanwhile, we designed a two-branch training architecture and pre-training technique to ensure the network can successfully converge and achieve excellent performance. Extensive ablation studies on the H36M and 3DHP datasets illustrate the effectiveness and generalization ability of our approach. The experiment results showed that our method obtained a superior performance over recent weakly/self-supervised methods.

Acknowledgments
This research was supported by National Key R & D Program of China (No. 2016YFB0801100), Beijing Natural Science Foundation (No. 4172054, L181010), National Basic Research Program of China (No. 2013CB329605), and Graduate Technological Innovation Project of Beijing Institute of Technology (No.2018CX20026).

References


