# Global Adaptation meets Local Generalization: Unsupervised Domain Adaptation for 3D Human Pose Estimation

Wenhao Chai[1] Zhongyu Jiang[2] Jenq-Neng Hwang[2] Gaoang Wang[1] ✉
[1] Zhejiang University [2] University of Washington

## Abstract

*When applying a pre-trained 2D-to-3D human pose lifting model to a target unseen dataset, large performance degradation is commonly encountered due to domain shift issues. We observe that the degradation is caused by two factors: 1) the large distribution gap over global positions of poses between the source and target datasets due to variant camera parameters and settings, and 2) the deficient diversity of local structures of poses in training. To this end, we combine* ***global adaptation*** *and* ***local generalization*** *in PoseDA, a simple yet effective framework of unsupervised domain adaptation for 3D human pose estimation. Specifically, global adaptation aims to align global positions of poses from the source domain to the target domain with a proposed global position alignment (GPA) module. And local generalization is designed to enhance the diversity of 2D-3D pose mapping with a local pose augmentation (LPA) module. These modules bring significant performance improvement without introducing additional learnable parameters. In addition, we propose local pose augmentation (LPA) to enhance the diversity of 3D poses following an adversarial training scheme consisting of 1) a augmentation generator that generates the parameters of pre-defined pose transformations and 2) an anchor discriminator to ensure the reality and quality of the augmented data. Our approach can be applicable to almost all 2D-3D lifting models. PoseDA achieves 61.3 mm of MPJPE on MPI-INF-3DHP under a cross-dataset evaluation setup, improving upon the previous state-of-the-art method by 10.2%.*

## 1. Introduction

3D human pose estimation is an essential computer vision task which aims to estimate the coordinates of 3D joints from single-frame images or videos. This task can be further used for several downstream tasks in multiple object tracking [1, 15, 13, 21], person re-identification [45], action recognition [44], robot [47], human body reconstruction [14], sports application [64], *etc*. However, large-scale

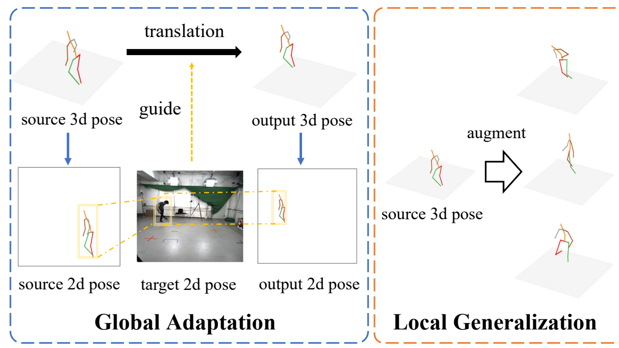


Figure 1. *PoseDA* addresses 3D human pose domain adaptation problem through global adaptation and local generalization. The 2D poses from target dataset are used to guide the translation of 3D poses from source dataset. And the local root-relative poses also are augmented to achieve better generalization ability.

3D-annotated datasets are hard to obtain. Existing methods are usually built on an off-the-shelf 2D pose estimators [9, 46] following two-stage schemes.

Deep learning methods [37, 41] have achieved great success in the mapping from 2D to 3D in the past decade. Despite their success in in-distribution data, these fully-supervised methods show poor performance in cross-dataset inference [18]. We argue that the real bottleneck lies in the domain gap of 3D pose data rather than the 2D-3D lifting network architecture or training strategy. Existing datasets either lack enough diversity in laboratorial environments [23] or lack adequate quantity and accuracy in the wild [38] due to the complex visual condition [59, 57, 58, 34, 33, 24]. We model the pose domain gap in terms of global position and local pose separately shown in Figure 1. As for the global position, the camera intrinsic and extrinsic parameters are completely different in different datasets, resulting in performance degradation in cross-dataset evaluation. And for the local pose, the lack of action diversity also limits the generalization ability of the model. Addressing the domain adaptation or generalization problem is a crucial step for 3D human pose estimation to move from toy experiments to real-world applications. Recent works focus more on enhancing the generalization ability of the 2D-3D lifting networks or set camera view prediction as an auxiliary task [52, 54] to address adaptation problems. Some methods apply data augmentation in training images through image transformations [42, 39, 40] or human synthetics [8, 22, 49]. However, our proposed method does not rely on RGB or temporal information. Specifically, our method first generates transformed pose pairs from source dataset and then use them to train the 2D-3D lifting network, thus can fit any off-the-shelf model.

In this paper, we propose *PoseDA*, an unsupervised domain adaptation framework for 3D human pose estimation. Our method only requires non-sequence 2D poses (not images) and camera intrinsic parameters in target dataset as well as a large-scale 3D-annotated human pose dataset (*e.g.,* Human3.6M [23]). In real-world scenarios, obtaining prior knowledge of the camera intrinsic parameters is often not a concern. This is due to the fact that such information is readily available from camera specifications or can be inferred from the input images or videos alone [48]. The basic idea behind the proposed method is to combine global adaptation and local generalization to address the issue of unsupervised domain adaptation for 3D human pose estimation. Global adaptation aims to align global positions of poses from source domain to target domain, and local generalization aims to enhance the diversity of local structures of poses. Therefore, our proposed method applies global position alignment strictly but only enhances the diversity of local poses. To be specific, we take a sample from the source dataset and apply transformations in terms of bone angle, bone length, and rotation. We use an augmentation generator to generate the parameters for these transformations and an anchor discriminator to ensure the realisticity and quality of these transformed pose pairs. As for the global position, we apply 2D global position alignment to ensure the alignment in both scales and 2D root positions between the projected 3D poses from the source domain with sampled 2D poses from the target domain. This process is solvable through geometry constraints with no additional learnable parameters. Finally, we use the transformed pose pairs to fine-tune the pre-trained 2D-3D lifting network and thus boost model performance on the target dataset without any use of 3D annotations. Our contributions are summarized as follows:

- We reduce the domain gap by separately applying global position alignment and local pose augmentation, where two major domain gaps are effectively decoupled.
- We align the global position through geometry constraint with no additional learnable parameters, which can boost model performance significantly. And we apply local pose augmentation to enhance the diversity of local structures of 3D poses.
- Our approach is applicable to almost all 2D-3D lifting models. We achieve the state-of-the-art performance in Human3.6M-3DHP cross-dataset evaluation with 61.3 mm of MPJPE.

## 2. Related Work

### 2.1. Two-stage 3D Human Pose Estimation

Inspired by the rapid development of 2D human pose estimation algorithms, many works have tried to utilize 2D pose estimation results for 3D human pose estimation to improve in-the-wild performance [53, 62]. Two-stage 3D human pose estimation approaches, which first estimate 2D poses and then lift 2D poses to 3D poses, have been developed. Chen *et al*. [5] present a simple approach to 3D human pose estimation by performing 2D pose estimation, followed by 3D exemplar matching. Martinez *et al*. [37] propose a baseline focusing on lifting 2D poses to 3D with a simple yet effective neural network, which popularizes the research on lifting 2D pose to 3D space. Recently, semi/self-supervised learning based on geometry constraint [41, 6, 12, 25] has been used to train models without 3D annotations or by auxiliary losses.

### 2.2. Data Augmentation on 3D Human Poses

Data augmentation is widely used to improve deep model generalization ability by enhancing training data diversity. Some methods apply pose data augmentation on images [42, 40] or generate 3D synthetic data using graphics engines [8, 56, 17]. Other methods directly generate 2D-3D pose pairs by applying transformations on 3D skeleton[31, 18, 16]. Gong *et al*. [18] make this augmentation process differentiable and further learnable. Those transformations consist of bone angle, bone length, and rigid-body transformation.

### 2.3. Unsupervised Domain Adaptation for 3D Human Pose Estimation

Unsupervised domain adaptation [55] aims to transfer models from a fully-labeled source domain to an unlabeled target domain. Kundu *et al*. [30] address unsupervised domain adaptation problem by modeling pose uncertainty based on RGB images. Li *et al*. [31] are the first to generate corresponding 2D-3D pose pairs by applying skeleton transformations. Gong *et al*. [18] develop this data augmentation module with the differentiable form to jointly optimize the augmentation process with a end-to-end trained model. Gholami *et al*. [16] further utilize 2D pose in target domain to address the domain adaptation problem. However, all existing methods focus more on local pose augmentation or adaptation. In this paper, *PoseDA* utilizes both local pose augmentation and global position alignment, and achieves state-of-the-art performance in cross-domain tasks.

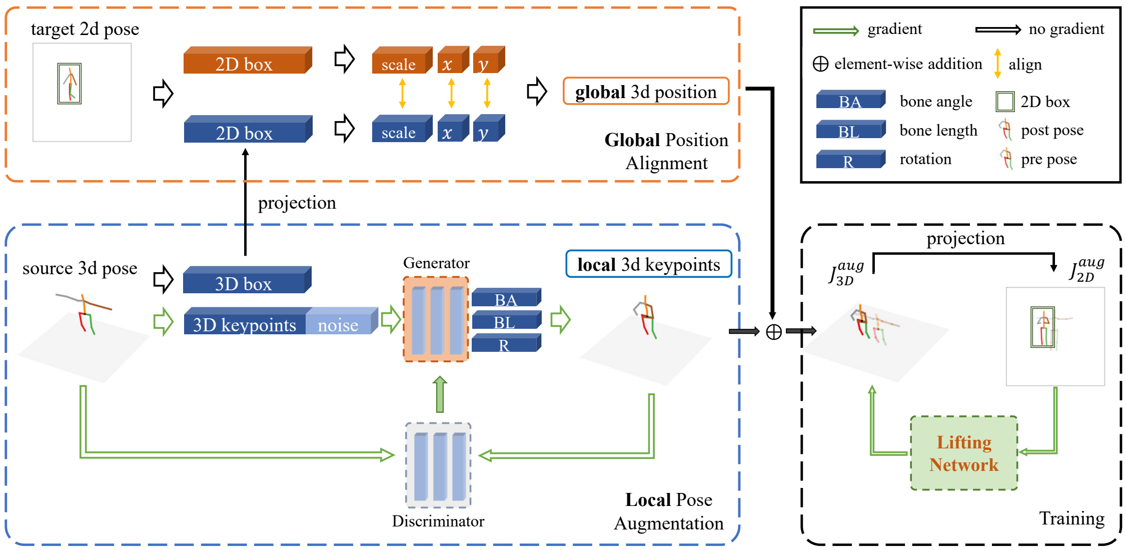


Figure 2. Our proposed unsupervised domain adaptation framework *PoseDA* consists of global position alignment (GPA) and local pose augmentation (LPA). The augmentation bone angle (BA), bone length (BL), and rotation (R) are applied on source 3D poses through an adversarial augmentation framework consists of an augmentation generator $\mathcal{G}$ and an anchor discriminator $\mathcal{D}_{3D}$. Meanwhile, we also sample a 2D pose from target dataset and align the scale, x coordination, and y coordination in 2D screen between the target 2D box and projected source 2D box. Thus we solve the global 3D position by those geometric constraint. Finally, the augmented pose pairs combining with global 3D position and local 3D keypoints are used to train the lifting network $\mathcal{P}$.

# 3. Method

In this section, we introduce *PoseDA*, an unsupervised domain adaptation framework, as summarized in Figure 2, which consists of global position alignment (GPA) in Section 3.2 and local pose augmentation (LPA) in Section 3.3. Global position alignment module aims to align the 2D pose spatial distribution of both scale and location, $(x, y)$ coordinatse, between source and target datasets, and local pose augmentation is designed to enhance the diversity of 3D-2D pose mapping. Finally, we formulate the training process with several loss functions. The pseudo-code for the overall training process of our method is given in Algorithm 1, and the transformation pipeline visualization of corresponding 2D-3D pose pairs is shown in Figure 3.

## 3.1. Problem Formulation

Let $S = (\boldsymbol{J}_{2D}^{src}, \boldsymbol{J}_{3D}^{src})$ denote corresponding 2D-3D pose pairs from the source dataset, and $\boldsymbol{J}_{2D}^{tar}$ denote 2D pose from target dataset extracted by an off-the-shelf 2D pose estimator. Note that we only use $\boldsymbol{J}_{3D}^{tar}$ for evaluation but not for training. All the 3D poses $\boldsymbol{J}_{3D}$ in this paper are root-relative since we do not predict the global position.

Our method conducts data augmentation on pose pairs $S = (\boldsymbol{J}_{2D}^{src}, \boldsymbol{J}_{3D}^{src})$ from source dataset. The 2D pose $\boldsymbol{J}_{2D}^{tar}$ and the camera intrinsic parameters from target dataset is used to guide this process:

$$\boldsymbol{J}_{3D}^{aug} = \mathcal{G}(\boldsymbol{J}_{3D}^{src}; \boldsymbol{\theta}_{cam}),\ \boldsymbol{J}_{2D}^{aug} = f_p(\boldsymbol{J}_{3D}^{aug}; \boldsymbol{\theta}_{cam}) \quad (1)$$

where $(\boldsymbol{J}_{2D}^{aug}, \boldsymbol{J}_{3D}^{aug})$ denotes augmented pose pairs, $\boldsymbol{\theta}_{cam}$ denotes the camera intrinsic parameters from target dataset, and $f_p$ denotes the projection function from 3D camera coordinates to 2D image coordinates. The augmented pose pairs are further used to train the pose lifting network.

We use several strong baseline methods (*e.g.,* VideoPose3D [41] and SimpleBaseline [37]) for lifting 2D poses to 3D poses. Let $\mathcal{P} : \boldsymbol{J}_{2D} \mapsto \boldsymbol{J}_{3D}$ denotes the lifting network with parameters $\boldsymbol{\theta}_{\mathcal{P}}$, which can be trained in fully-supervised paradigm as:

$$\min_{\boldsymbol{\theta}_{\mathcal{P}}} \mathcal{L}_{\mathcal{P}}(\mathcal{P}(\boldsymbol{J}_{2D}; \boldsymbol{\theta}_{\mathcal{P}}), \boldsymbol{J}_{3D}) \quad (2)$$

where $(\boldsymbol{J}_{2D}, \boldsymbol{J}_{3D})$ denotes paired 2D-3D pose pairs, which consist of both augmented pose pairs $(\boldsymbol{J}_{2D}^{aug}, \boldsymbol{J}_{3D}^{aug})$ and original pose pairs $(\boldsymbol{J}_{2D}^{src}, \boldsymbol{J}_{3D}^{src})$, the loss function $\mathcal{L}_{\mathcal{P}}$ is typically defined as Mean Square Error (MSE), which is corresponding to the evaluation metric Mean Per Joint Position Error (MPJPE).

## 3.2. Global Position Alignment (GPA)

Global position alignment (GPA) is designed to eliminate the domain gap in viewpoints, which is simple yet efficient. By applying Monte Carlo sampling [7], the scale and location distributions of the 2D poses of the source dataset can be migrated to distributions of target dataset. We first construct pose pairs $(\boldsymbol{J}_{3D}^{src}, \boldsymbol{J}_{2D}^{tar})$, which are pairs of root-relative 3D poses and 2D poses randomly sampled from source and target domains respectively.

Given a 2D pose $\boldsymbol{J}_{2D}^{tar} = [\boldsymbol{x}^{tar}, \boldsymbol{y}^{tar}]^T \in \mathbb{R}^{2\times J}$ and 3D pose $\boldsymbol{J}_{3D}^{src} = [\boldsymbol{X}^{src}, \boldsymbol{Y}^{src}, \boldsymbol{Z}^{src}]^T \in \mathbb{R}^{3\times J}$ with the root at the origin $[0,0,0]^T$, GPA aims to estimate the translated 3D root position $\boldsymbol{J}_r = [X_r, Y_r, Z_r]^T \in \mathbb{R}^{3\times 1}$ to ensure the reprojected 2D pose $\boldsymbol{J}_{2D}^{proj} = [\boldsymbol{x}^{proj}, \boldsymbol{y}^{proj}]^T$ from $\boldsymbol{J}_{3D}^{src} + \boldsymbol{J}_r$ is close to $\boldsymbol{J}_{2D}^{tar}$ as much as possible with respect to both position and scale. Denote the operation of GPA as $\mathcal{F}$,

$$\hat{\boldsymbol{J}}_r = \mathcal{F}(\boldsymbol{J}_{2D}^{tar}, \boldsymbol{J}_{3D}^{src}), \tag{3}$$

where $\hat{\boldsymbol{J}}_r$ is the estimated 3D root position after translation.

We demonstrate how to obtain $\hat{\boldsymbol{J}}_r$ as follows. Assume that the camera intrinsic parameters are given. The projection from 3D joints to 2D joints after translation should obey the perspective projection function as follows:

$$\begin{aligned} x_i^{proj} &= \frac{f_x(X_i^{src} + X_r)}{Z_i^{src} + Z_r} + c_x, \\ y_i^{proj} &= \frac{f_y(Y_i^{src} + Y_r)}{Z_i^{src} + Z_r} + c_y, \end{aligned} \tag{4}$$

where $i$ denotes the $i$-th joint, $(f_x, f_y), (c_x, c_y)$ denote focal length and principal point respectively. Note that $Z_i^{src} \ll Z_r$ since the absolute depth of the root joint of the person $Z_r$ is usually much larger than depth offset $Z_i^{src}$ of a certain joint relative to the root joint. Therefore, we assume that $Z_i^{src} + Z_r \approx Z_r$ holds. Then Eq. (4) becomes

$$\begin{aligned} x_i^{proj} &\approx \frac{f_x(X_i^{src} + X_r)}{Z_r} + c_x, \\ y_i^{proj} &\approx \frac{f_y(Y_i^{src} + Y_r)}{Z_r} + c_y, \end{aligned} \tag{5}$$

To achieve the similar scale with $\boldsymbol{J}_{2D}^{tar}$ for $\boldsymbol{J}_{2D}^{proj}$, we ensure that the 2D boxes should have similar size with the following constraint,

$$\Delta x^{proj} + \Delta y^{proj} = \Delta x^{tar} + \Delta y^{tar}, \tag{6}$$

where $\Delta$ denotes the difference between the max-min coordinates of 2D joints, *i.e.*, the width and height of the 2D box. Combine Eq. 5 and Eq. 6, we can get the approximated $Z_r$,

$$\hat{Z}_r \approx \frac{f_x \Delta X^{src} + f_y \Delta Y^{src}}{\Delta x^{tar} + \Delta y^{tar}}. \tag{7}$$

Then, we can align the global (root) position between $\boldsymbol{J}_{2D}^{proj}$ and $\boldsymbol{J}_{2D}^{tar}$ by

$$x_r^{proj} = x_r^{tar}, \quad y_r^{proj} = y_r^{tar}, \tag{8}$$

where $[x_r, y_r]^T$ represents the 2D root joint. Combined with Eq. (5), we can get estimated $\hat{X}_r$ and $\hat{Y}_r$,

$$\hat{X}_r = \frac{\hat{Z}_r(x_r^{tar} - c_x)}{f_x}, \quad \hat{Y}_r = \frac{\hat{Z}_r(y_r^{tar} - c_y)}{f_y}. \tag{9}$$

Finally, we can obtain the translated root position $\hat{\boldsymbol{J}}_r = [\hat{X}_r, \hat{Y}_r, \hat{Z}_r]^T$. The ultimate goal of Global Position Alignment is to align the 2D pose distributions of the target domain and the generated domain (after going through the full data augmentation pipeline) in terms of scale and position. There will be cases of large discrepancies between $J_{3D}^{src}$ and $J_{2D}^{tar}$ when pairing randomly on individuals. However, this is mitigated because 1) we only use box information rather than complete poses and 2) we randomly shuffle the pairing method between each epoch, which further increases diversity and avoids individual discrepancies in a statistical sense.

### 3.3. Local Pose Augmentation (LPA)

Inspired by *PoseAug* [18], We also apply local pose augmentation (LPA) to enhance the diversity of 2D-3D pose mappings. The augmentation transformation of 3D pose can be decoupled into perturbations of bone vector, bone length, and rotation. We design an adversarial augmentation framework, which consists of an augmentation generator to generate 3D pose transformation parameters and an anchor discriminator to ensure the realisticity, quality, and diversity of generated pose pairs. The generator and discriminator are jointly end-to-end trained following the GAN style.

**Augmentation generator**. Following *PoseAug* [18], we propose an augmentation generator denoted as $\mathcal{G}$ with parameters $\boldsymbol{\theta}_{\mathcal{G}}$. Unlike vanilla GAN-style generator, we take a sample 3D pose from source dataset $J_{3D}^{src}$ as the condition $\mathcal{G}_{cond}$, which is concatenated with a noise vector $\boldsymbol{z}$ as the input of $\mathcal{G}$, according to the suggestion in [16, 3]. The input 3D pose is converted to bone direction vectors representing the joint angle and bone length. Augmentation generator $\mathcal{G}$ generates three types of 3D pose transformations: bone angle difference, bone length difference, and global rotation. The augmentation process can be represented as

$$\boldsymbol{J}_{3D}^{aug} = \mathcal{G}(\boldsymbol{J}_{3D}^{src}, \boldsymbol{z}; \boldsymbol{\theta}_{\mathcal{G}}), \quad \boldsymbol{J}_{2D}^{aug} = f_p(\boldsymbol{J}_{3D}^{aug}; \boldsymbol{\theta}_{cam}) \tag{10}$$

where $(\boldsymbol{J}_{3D}^{aug}, \boldsymbol{J}_{2D}^{aug})$ denotes the augmented paired 2D-3D pose pairs, $f_p$ denotes the camera projection function 4, and $\boldsymbol{\theta}_{cam}$ denotes the given camera intrinsic parameters in target dataset.

**Anchor discriminator**. Let $\mathcal{D}_{3D}$ denote the discriminator for 3D pose with parameters $\boldsymbol{\theta}_{\mathcal{D}_{3D}}$. It takes root-relative $\boldsymbol{J}_{3D}$ sampled from both $\boldsymbol{J}_{3D}^{src}$ and $\boldsymbol{J}_{3D}^{aug}$ as input. Discriminator $\mathcal{D}_{3D}$ works as an anchor discriminator to ensure the augmented pose $\boldsymbol{J}_{3D}^{aug}$ is reasonable. Inspired by Kinematic Chain Space (KCS) [52, 51], we use KCS representation of 3D pose as input instead of 3D joints. With the help of KCS representation, the generated 3D poses $\boldsymbol{J}_{3D}^{aug}$ are no longer limited in the source domain.

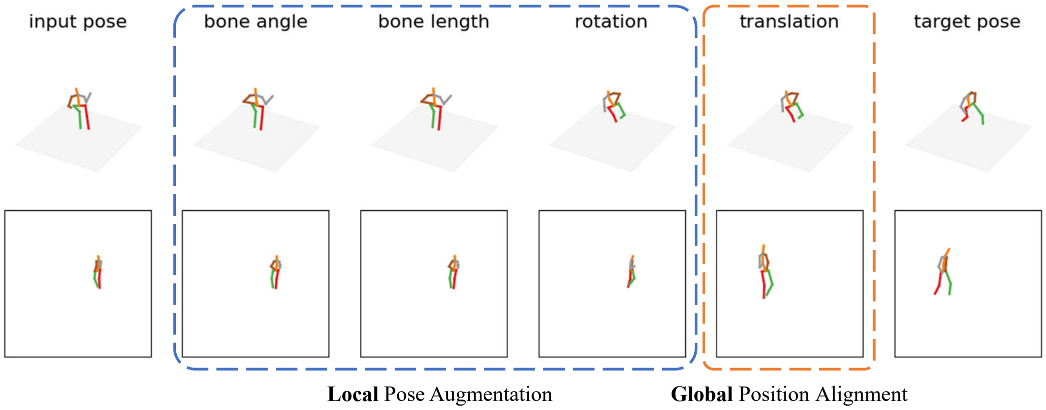


Figure 3. The pose transformations applied on 3D pose from source dataset (column 1). Bone angle (column 2), bone length (column 3), and rotation (column 4) are root-relative transformations given by local pose augmentation (LPA). Translation (column 5) is given by global position alignment (GPA). Target (column 6) is the sampled 2D pose from target dataset used for GPA.

### 3.4. Training

Previous works [18, 16, 56] train the adversarial pose augmentation framework with the loss function of vanilla GAN [19] or least squares GAN (LSGAN) [36]. We argue that training based on Wasserstein distance [43, 11] can be trained stably and provide better augmented poses.

**Discriminator loss**. We adopt WGAN [2] loss for the anchor discriminator $\mathcal{L}_{\mathcal{D}_{3D}}$:

$$\mathcal{L}_{\mathcal{D}_{3D}} = \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{J}_{3D}^{aug}}\left[\mathcal{D}_{3D}(\boldsymbol{x})\right] - \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{J}_{3D}^{src}}\left[\mathcal{D}_{3D}(\boldsymbol{x})\right] \quad (11)$$

**Generator loss**. The adversarial loss of the augmentation generator is the feedback from the anchor discriminator.

$$\mathcal{L}_{\mathcal{G}} = -\mathbb{E}_{\boldsymbol{x}\sim\mathcal{G}(\boldsymbol{J}_{3D}^{src},\boldsymbol{z})}\left[\mathcal{D}_{3D}(\boldsymbol{x}))\right] \quad (12)$$

The discriminator and generator are trained iteratively.

**Lifting network loss**. The standard Mean Squared Error (MSE) loss is adopted to the lifting network $\mathcal{P}$,

$$\mathcal{L}_{\mathcal{P}} = \|\boldsymbol{J}_{GT} - \boldsymbol{J}\|_2^2, \quad (13)$$

where $\boldsymbol{J}_{GT}$ and $\boldsymbol{J}$ are ground truth and estimated 3D joints, respectively. The generator and the discriminator need to be warmed up for $w$ epoch before training lifting network.

**Algorithm 1:** The training pipeline of our method

**Input:** $\boldsymbol{J} = \{\boldsymbol{J}_{3D}^{src}, \boldsymbol{J}_{2D}^{tar}, \boldsymbol{\theta}_{cam}\}$

**for** $t \leftarrow 1$ **to** $T$ **do**
 **for** $i \leftarrow 1$ **to** $I$ **do**
  **freeze** $\mathcal{G}$
  *sample and generate a batch data*
  $\boldsymbol{J}_{3D}^{aug}, \boldsymbol{J}_{2D}^{aug} \leftarrow \mathcal{G}(\boldsymbol{J}_{3D}^{src}, \boldsymbol{z}; \boldsymbol{\theta}_{\mathcal{G}})$
  *train discriminator*
  $\mathcal{L}_{\mathcal{D}_{3D}} = \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{J}_{3D}^{aug}}\left[\mathcal{D}_{3D}(\boldsymbol{x};\boldsymbol{\theta}_{\mathcal{D}_{3D}})\right] - \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{J}_{3D}^{src}}\left[\mathcal{D}_{3D}(\boldsymbol{x};\boldsymbol{\theta}_{\mathcal{D}_{3D}})\right]$
  **update** $\boldsymbol{\theta}_{\mathcal{D}_{3D}}$
  *train generator every $n$ iters*
  **if** *$i$ in every $n$ iters* **then**
   **freeze** $\boldsymbol{\theta}_{\mathcal{D}_{3D}}$
   *generate a batch data*
   $\boldsymbol{J}_{3D}^{aug}, \boldsymbol{J}_{2D}^{aug} \leftarrow \mathcal{G}(\boldsymbol{J}_{3D}^{src}, \boldsymbol{z}; \boldsymbol{\theta}_{\mathcal{G}})$
   $\mathcal{L}_{\mathcal{G}} = -\mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{J}_{3D}^{aug}}\left[\mathcal{D}_{3D}(\boldsymbol{x};\boldsymbol{\theta}_{\mathcal{D}_{3D}})\right]$
   **update** $\boldsymbol{\theta}_{\mathcal{G}}$
 *warmup to ensure stable augmentation*
 **if** $t \geq w$ **then**
  *mixup augmented and source data*
  $\boldsymbol{J}_{2D}^{mix}, \boldsymbol{J}_{3D}^{mix} \leftarrow (\boldsymbol{J}_{2D}^{src}, \boldsymbol{J}_{3D}^{src}), (\boldsymbol{J}_{2D}^{aug}, \boldsymbol{J}_{3D}^{aug})$
  *train lifting network with mixed data*
  $\mathcal{L}_{\mathcal{P}} = \mathrm{MSE}\left(\mathcal{P}\left(\boldsymbol{J}_{2D}^{mix};\boldsymbol{\theta}_{\mathcal{P}}\right), \boldsymbol{J}_{3D}^{mix}\right)$
  **update** $\boldsymbol{\theta}_{\mathcal{P}}$

## 4. Experiments

In this section, we conduct experiments on several popular human pose estimation benchmarks with comprehensive evaluation metrics. Since *PoseDA* is flexible regarding the architecture of the pose lifting network, we perform our framework on different strong baselines to show the universality. GT 2D pose is used as default. We also analyze the contribution of each component in ablation studies.

### 4.1. Datasets and Metrics

The datasets used for quantitative evaluations are several widely used large-scale 3D human pose estimation benchmarks, including Human3.6M [23], MPI-INF-3DHP [38], and 3DPW [50].

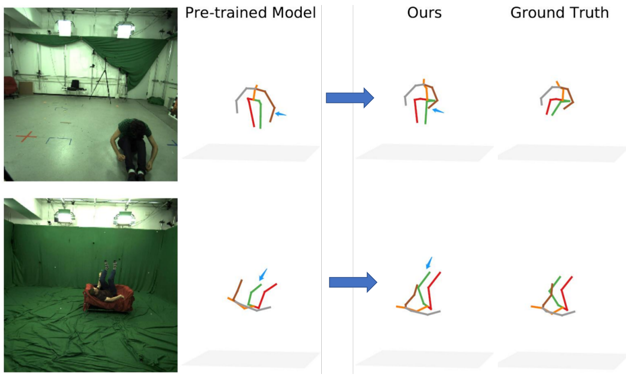


Figure 4. Qualitative results on MPI-INF-3DHP dataset. We use VideoPose3D [41] as pre-trained model. After finetuning with *PoseDA*, the model performs better for some challenging poses.

| Method | S | MPJPE (↓) | PA-MPJPE (↓) |
|---|---|---|---|
| VideoPose3D [41] | Full | 51.8 | 40.0 |
| ST-GCN [4] | Full | 50.6 | 40.2 |
| SimpleBaseline [37] | Full | 45.5 | 37.1 |
| SemGCN [63] | Full | 43.8 | - |
| VideoPose3D [41] | S1 | 64.7 | - |
| Li *et al*. [31] | S1 | 62.9 | - |
| PoseAug [18] | S1 | 56.7 | - |
| AdaptPose [16] | S1 | 54.2 | 35.6 |
| *PoseDA* (Ours) | S1 | **49.9** | **34.2** |

Table 1. **Results on H3.6M**. S denotes source data. MPJPE and PA-MPJPE are used as evaluation metrics. Source: H3.6M-S1. Target: H3.6M-S5, S6, S7, S8.

**Human3.6M (H3.6M)** is one of the largest 3D human pose datasets captured in a laboratorial environment. Following previous works [16, 41], there are two different settings of H3.6M: 1) using the S1, S5, S6, S7 and S8 as our source domain for cross-dataset evaluation. 2) using only S1 as source domain and others (S5, S6, S7, S8) as target domain.
**MPI-INF-3DHP (3DHP)** is a large-scale in-the-wild 3D human pose dataset with more diverse actions and motions. This dataset is closer to real-world scenarios and ideal for evaluating our method. Following previous works [29, 18], we use its test set, which includes 2,929 frames. We report the results of *PoseDA* using metrics of MPJPE, Percentage of Correct Keypoints (PCK), and Area Under the Curve (AUC).
**3DPW** is an in-the-wild dataset, unlike H3.6M or 3DHP, with uncontrolled motion and scene. Since it is a much more challenging dataset, we train each method on H3.6M and evaluate it on the 3DPW test set with the PA-MPJPE and MPJPE metric.

### 4.2. Implementation Details

**Details of discriminators**. For the anchor discriminator, we split 3D keypoints into five parts following [18], pass them through each of the five 4-layer residual MLPs with LeakyReLU, and finally concatenate the output. We use RMSProp optimizer for the anchor discriminators.
**Details of generators**. We sequentially build three 3-layer residual MLPs with LeakyReLU to generate bone angle, bone length, and rotation. Each network receives the output of the previous network as well as noise vector as input. These three networks are trained jointly by weighted adversarial losses. For every 6 iterations, we train the generator once and the discriminator 5 times.
**Details of pose lifting network**. We use VideoPose3D [41] (1-frame) as pose lifting network as well as other strong baselines [4, 37, 63] in ablation study. The pre-trained weights from the source dataset are used as initial weights for all the experiments.
**Details of training**. Starting with pre-trained weight, it takes 5 to 10 epochs (50 secs per epoch) to get convergence. In the experiments, we also found that WGAN not only has more stable training than LSGAN, but also prevents mode collapse, but the performance is not much different.We only train the generator and discriminator at the first 5 epochs for stable augmentation. Then, we mix the augmented data and original data in an 1:1 ratio for pose lifting network training. The learning rate for all the networks is $1e^{-4}$. *PoseDA* is trained on NVIDIA RTX 3090. The training process takes two hours and consumes 2GB of GPU memory.

### 4.3. Results

**Results on H3.6M**. We compare *PoseDA* with state-of-the-art methods [41, 31, 18, 16] using ground truth 2D keypoints as input. The results on H3.6M are shown in Table 1, focusing on the generalization ability of the model over actions, since the distributions over global positions are relatively consistent between the different parts of the same dataset. Even though AdaptPose [16] utilizes temporal information, our method still achieves state-of-the-art performance.
**Results on 3DHP**. Our method achieves remarkable performance in all the metrics. We use single frame ground truth 2D keypoints as input and therefore compare against various recent methods with the same setting. *PoseDA* improve upon the previous state-of-the-art method by 10.2%, and is even competitively compared with some state-of-the-art fully supervised method.
**Results on 3DPW** [50]. We use ground truth 2D keypoints as input. *PoseDA* achieves the state-of-the-art performance without using any 3D annotations in 3DPW, as shown in Table 3, even is favorably compared with video-based methods.
**Qualitative evaluation**. Figure 4 shows the qualitative evaluation on 3DHP dataset. Compared with the baseline model without training with *PoseDA*, our method performs well for unusual human positions and challenging poses.

| Method | CD | MPJPE (↓) | PCK (↑) | AUC (↑) |
|---|---|---|---|---|
| Multi Person [39] | | 122.2 | 75.2 | 37.8 |
| Mehta *et al.* [38] | | 117.6 | 76.5 | 40.8 |
| VNect [40] | | 124.7 | 76.6 | 40.4 |
| OriNet [35] | | 89.4 | 81.8 | 45.2 |
| SimpleBaseline [37] | | 84.3 | 85.0 | 52.0 |
| MixSTE [61] | | 57.9 | 94.2 | 63.8 |
| LCN [10] | ✓ | - | 74.0 | 36.7 |
| HMR [27] | ✓ | 113.2 | 77.1 | 40.7 |
| SRNet [60] | ✓ | - | 77.6 | 43.8 |
| Li *et al.* [31] | ✓ | 99.7 | 81.2 | 46.1 |
| RepNet [52] | ✓ | 92.5 | 81.8 | 54.8 |
| PoseAug [18] | ✓ | 73.0 | 88.6 | 57.3 |
| AdaptPose [16] | ✓ | 68.3 | 90.2 | 59.0 |
| *PoseDA* (Ours) | ✓ | **61.3** | **92.1** | **62.5** |

Table 2. **Results on 3DHP**. CD denotes cross-domain evaluation (no CD denotes fully-supervision, i.e., trained and tested on the same 3DHP dataset). PCK, AUC and MPJPE are used as evaluation metrics. Source: H3.6M. Target: 3DHP.

| Method | CD | V | PA-MPJPE (↓) | MPJPE (↓) |
|---|---|---|---|---|
| VideoPose3D [41] | | ✓ | 68.0 | 105.0 |
| EFT [26] | | | 55.7 | - |
| VIBE [28] | | ✓ | 51.9 | 82.9 |
| Lin *et al.* [32] | | | 45.6 | 74.7 |
| PoseAug [18] | ✓ | | 58.5 | 94.1 |
| VIBE [28] | ✓ | ✓ | 56.5 | 93.5 |
| BOA [20] | ✓ | ✓ | 49.5 | **77.2** |
| AdaptPose [16] | ✓ | ✓ | **46.5** | 81.2 |
| *PoseDA* (Ours) | ✓ | | 55.3 | 87.7 |

Table 3. **Results on 3DPW**. CD denotes cross-domain evaluation (no CD denotes fully-supervision, i.e., trained and tested on the same 3DPW dataset), V denotes video-based method. PA-MPJPE and MPJPE are used as evaluation metrics. Source: H3.6M. Target: 3DPW.

## 4.4. Ablation Study

**Analysis on each components and board lifting network**. Since *PoseDA* is a data augmentation framework, with any pose lifting network architecture. Shown in Tabel 5, we conduct experiments on several strong baselines with different architectures, including MLP [37], Convolutional Network [41], and Graph Convolutional Network [4, 63]. We also check the effectiveness of each component in *PoseDA*. For fair comparison, we use the same weights of generator and discriminators in all the experiments. It shows that both global position alignment (GPA) and local pose augmentation (LPA) benefit the adaptive performance. Moreover, GPA significantly boosts performance without any extra training or use of additional learnable parameters.

**Analysis on global pose alignment (GPA)**. We argue that the crop and normalization on 2D poses is an inaccurate method compared to our GPA module. Because the projection relation is not a linear operator and therefore does not

| Method | MPJPE (↓) | PCK (↑) | AUC (↑) |
|---|---|---|---|
| SemGCN [63] | 95.96 | 80.68 | 48.48 |
| + LPA | 87.64 (-8.3) | 84.21 (+3.5) | 51.24 (+2.8) |
| + GPA | 86.56 (-9.4) | 83.85 (+3.2) | 50.98 (+2.5) |
| + *PoseDA* | **78.37** (-17.6) | **86.17** (+5.5) | **54.74** (+6.3) |
| SimpleBaseline [37] | 81.23 | 85.85 | 53.95 |
| + LPA | 66.56 (-14.7) | 90.16 (+4.3) | 60.41 (+6.5) |
| + GPA | 69.19 (-12.0) | 89.90 (+4.1) | 58.50 (+4.6) |
| + *PoseDA* | **64.79** (-16.4) | **90.55** (+4.7) | **61.32** (+7.4) |
| ST-GCN [4] | 81.19 | 85.92 | 53.78 |
| + LPA | 74.31 (-6.9) | 88.72 (+2.9) | 56.20 (+2.4) |
| + GPA | 74.41 (-6.8) | 88.58 (+2.7) | 55.52 (+1.7) |
| + *PoseDA* | **69.50** (-11.7) | **90.15** (+4.2) | **58.56** (+4.8) |
| VideoPose3D [41] | 82.55 | 85.71 | 53.35 |
| + LPA | 66.65 (-15.9) | 90.05 (+4.3) | 60.24 (+6.9) |
| + GPA | 66.07 (-16.5) | 90.87 (+5.2) | 60.07 (+6.7) |
| + *PoseDA* | **61.36** (-21.2) | **92.05** (+6.3) | **62.52** (+9.2) |

Table 4. Ablation study on components and pose lifting network of our method. LPA denotes local pose augmentation, GPA denotes global position alignment. *PoseDA* combine GPA and LPA in a unified framework. Source: H3.6M. Target: 3DHP.

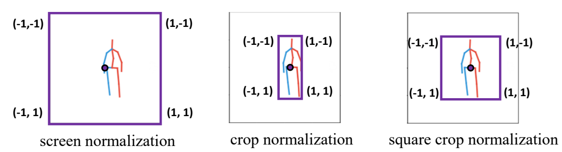


Figure 5. Illustration of different types of normalization operations.

have translation invariance. The crop out loses information about the relative positions of the camera and the person, and when the same 2d pose is cropped in a different position in the picture, the corresponding 3d pose is different (even if it is root-relative), there will be a one-to-many situation, which still increases ambiguity. We conduct experiments under four different pre-processing settings and with two backbones, our GPA based on screen normalized. We train the model on Human3.6M and test on 3DHP with the same pre-processing. Figure 5 shows different normalization operations and Table 5 shows our GPA method outperforms other normalization methods.

**Analysis on local pose augmentation (LPA)**. We also conduct ablation study on different methods of LPA. Comparing with random rotation and random bone length transformation, the proposed adversarial augmentation method achieves better performance as shown in Table 6.

## 4.5. Discussion Compared to SOTA Methods

The most related SOTA works are PoseAug and AdaptPose. Our proposed GPA has significant differences from the two methods in several aspects. In addition, our design of the discriminator is also different from these two methods. PoseAug is a simple domain generalization framework

| Method | MPJPE (↓) | PCK (↑) | AUC (↑) |
|---|---|---|---|
| SimpleBaseline | | | |
| w screen norm | 81.23 | 85.85 | 53.95 |
| w crop norm | 95.37 | 81.40 | 47.66 |
| w square crop | 84.81 | 84.29 | 52.27 |
| w GPA (*ours*) | **69.19** | **89.90** | **58.50** |
| VideoPose3D | | | |
| w screen norm | 82.55 | 85.71 | 53.35 |
| w crop norm | 93.16 | 82.23 | 48.00 |
| w square crop | 85.05 | 84.90 | 51.81 |
| w GPA (*ours*) | **66.07** | **90.87** | **60.07** |

Table 5. Ablation study on Global Position Alignment over other normalization techniques. Source: H3.6M. Target: 3DHP.

| Method | MPJPE (↓) | PCK (↑) | AUC (↑) |
|---|---|---|---|
| No Aug | 66.07 | 90.87 | 60.07 |
| RR | 66.88 | 90.77 | 59.31 |
| BL | 65.81 | 90.94 | 59.91 |
| RR + BL | 64.99 | 91.07 | 60.49 |
| *PoseDA* | **61.36** | **92.05** | **62.52** |

Table 6. Ablation study on local pose augmentation with different pre-defined pose augmentation methods. RR denotes random rotation along vertical axial and BL denotes random bone length transformation. Source: H3.6M. Target: 3DHP.

that uses two discriminators, called $D_{3D}$ and $D_{2D}$. Their goal is to regulate the poses generated by the generator, making them similar to the 3D and 2D poses of the source domain. AdaptPose is a domain adaptation framework that also uses these two discriminators, but the goal of $D_{2D}$ is to make the generated poses more similar to the 2D poses of the target domain. However, the assumption of AdaptPose is not supported after careful and comprehensive experiments. In the human pose domain adaptation problem, we abandon $D_{2D}$ and only use $D_{3D}$ to regulate the quality of the generated poses to achieve better results. The reason we give is that: 1) Compared to PoseAug, making the generated 2D poses unnecessary similar to the source domain brings greater diversity. 2) Compared to AdaptPose, we believe that forcing the generated 2D poses to be similar to the target domain does not guarantee that the corresponding 3D poses are also similar. In other words, this is also because the 2D-3D mapping has large ambiguity. As a result, with our carefully designed GPA and LPA modules, global adaptation and local generalization are well combined, which is also our major contribution.

We conduct extensive and convincing experiments on the input selection of local pose augmentation module. All the experiments in this section are based on pre-trained VideoPose3D and global position alignment. Following [16, 18], we design a 2D pose discriminator of 5-layer MLP. Note that the condition of generator is used only to generate transformation, we still apply those transformation on the

| $\mathcal{G}_{cond}$ | $\mathcal{D}_{3D}$ | $\mathcal{D}_{2D}$ | MPJPE (↓) | PCK (↑) | AUC (↑) |
|---|---|---|---|---|---|
| - | - | - | 66.07 | 90.87 | 60.07 |
| $\mathcal{T}$ | $\mathcal{T}$ | $\mathcal{T}$ | 66.34 | 90.81 | 59.72 |
| $\mathcal{T}$ | $\mathcal{S}$ | $\mathcal{T}$ | 65.91 | 91.02 | 59.92 |
| $\mathcal{T}$ | $\mathcal{S}$ | - | 65.37 | 90.98 | 60.30 |
| $\mathcal{S}$ | $\mathcal{T}$ | $\mathcal{T}$ | 64.20 | 91.64 | 60.85 |
| $\mathcal{S}$ | $\mathcal{T}$ | - | 63.66 | 91.57 | 61.34 |
| $\mathcal{S}$ | $\mathcal{S}$ | $\mathcal{S}$ | 73.55 | 88.96 | 56.41 |
| $\mathcal{S}$ | $\mathcal{S}$ | $\mathcal{T}$ | 65.46 | 91.27 | 60.03 |
| $\mathcal{S}$ | $\mathcal{S}$ | - | **61.36** | **92.05** | **62.52** |

Table 7. The input selection of generator condition $\mathcal{G}_{cond}$, 3D pose discriminator $\mathcal{D}_{3D}$, and 2D pose discriminator $\mathcal{D}_{2D}$ in LPA. $S, T$ denote the pose from source or target domain. Source: H3.6M. Target: 3DHP.

3D pose from the source dataset. The 3D pose of target dataset used in 3D discriminator is the prediction of corresponding 2D pose by the pre-trained lifting network. As shown in Table 7, these experiments lead to two important conclusions: 1) the design of 2D pose discriminator is unnecessary, and 2) there is a performance drop no matter either the target domain information is involved in the generator or the discriminator to adapt the characteristics of local pose.

## 5. Conclusion

This paper addresses the problem of unsupervised cross-domain adaptation for 3D human pose estimation. To reduce the domain gap, we propose global position alignment and local pose augmentation. We argue that global position alignment is simple yet effective, and the local pose augmentation enhances the diversity of 2D-3D pose mapping. The proposed global position alignment module significantly boosts performance with no additional learnable parameters needed. We also show that adversarial pose augmentation based on Wasserstein distance can further obtain stable, diverse, and high-quality pose pairs. With extensive and convincing experiments and ablation studies, *PoseDA* can be flexibly applied on any 2D-3D pose lifting network and make a significant step towards solving domain adaptation problems for 3D human pose estimation.

**Limitations and Future Work** Although we show that generalization performs better than adaptation on domain gap over local pose, it is still possible to design a method that can adapt 3D pose without hurting the diversity of 2D-3D mappings.

**Acknowledgement** This work is supported by the National Key R&D Program of China No.2022ZD0162000, and National Natural Science Foundation of China No.62106219.

## References


[1] Mykhaylo Andriluka, Umar Iqbal, Eldar Insafutdinov, Leonid Pishchulin, Anton Milan, Juergen Gall, and Bernt Schiele. Posetrack: A benchmark for human pose estimation and tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5167–5176, 2018. 1

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 5

[3] Konstantinos Bousmalis, Nathan Silberman, David Dohan, Dumitru Erhan, and Dilip Krishnan. Unsupervised pixel-level domain adaptation with generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3722–3731, 2017. 4

[4] Yujun Cai, Liuhao Ge, Jun Liu, Jianfei Cai, Tat-Jen Cham, Junsong Yuan, and Nadia Magnenat Thalmann. Exploiting spatial-temporal relationships for 3d pose estimation via graph convolutional networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2272–2281, 2019. 6, 7

[5] Ching-Hang Chen and Deva Ramanan. 3d human pose estimation= 2d pose estimation+ matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7035–7043, 2017. 2

[6] Ching-Hang Chen, Ambrish Tyagi, Amit Agrawal, Dylan Drover, Rohith Mv, Stefan Stojanov, and James M Rehg. Unsupervised 3d pose estimation with geometric self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2019. 2

[7] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2781–2790, 2022. 3

[8] Wenzheng Chen, Huan Wang, Yangyan Li, Hao Su, Zhenhua Wang, Changhe Tu, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Synthesizing training images for boosting human 3d pose estimation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 479–488. IEEE, 2016. 2

[9] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7103–7112, 2018. 1

[10] Hai Ci, Chunyu Wang, Xiaoxuan Ma, and Yizhou Wang. Optimizing network structure for 3d human pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2262–2271, 2019. 7

[11] Yicheng Deng, Cheng Sun, Yongqi Sun, and Jiahui Zhu. A synchronized reprojection-based model for 3d human pose estimation. *arXiv preprint arXiv:2106.04274*, 2021. 5

[12] Dylan Drover, Rohith MV, Ching-Hang Chen, Amit Agrawal, Ambrish Tyagi, and Cong Phuoc Huynh. Can 3d pose be learned from 2d projections alone? In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 2

[13] Zhaoxin Fan, Fengxin Li, Hongyan Liu, Jun He, and Xiaoyong Du. Human pose driven object effects recommendation. *arXiv preprint arXiv:2209.08353*, 2022. 1

[14] Zhaoxin Fan, Yuqing Pan, Hao Xu, Zhenbo Song, Zhicheng Wang, Kejian Wu, Hongyan Liu, and Jun He. Furpe: Learning full-body reconstruction from part experts. *arXiv preprint arXiv:2212.00731*, 2022. 1

[15] Zhaoxin Fan, Yazhi Zhu, Yulin He, Qi Sun, Hongyan Liu, and Jun He. Deep learning on monocular object pose detection and tracking: A comprehensive overview. *ACM Computing Surveys*, 55(4):1–40, 2022. 1

[16] Mohsen Gholami, Bastian Wandt, Helge Rhodin, Rabab Ward, and Z Jane Wang. Adaptpose: Cross-dataset adaptation for 3d human pose estimation by learnable motion generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13075–13085, 2022. 2, 4, 5, 6, 7, 8

[17] Kehong Gong, Bingbing Li, Jianfeng Zhang, Tao Wang, Jing Huang, Michael Bi Mi, Jiashi Feng, and Xinchao Wang. Posetriplet: Co-evolving 3d human pose estimation, imitation, and hallucination under self-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11017–11027, 2022. 2

[18] Kehong Gong, Jianfeng Zhang, and Jiashi Feng. Poseaug: A differentiable pose augmentation framework for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8575–8584, 2021. 1, 2, 4, 5, 6, 7, 8

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 5

[20] Shanyan Guan, Jingwei Xu, Yunbo Wang, Bingbing Ni, and Xiaokang Yang. Bilevel online adaptation for out-of-domain human mesh reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10472–10481, 2021. 7

[21] Shenghao Hao, Peiyuan Liu, Yibing Zhan, Kaixun Jin, Zuozhu Liu, Mingli Song, Jenq-Neng Hwang, and Gaoang Wang. Divotrack: A novel dataset and baseline method for cross-view multi-object tracking in diverse open scenes. *arXiv preprint arXiv:2302.07676*, 2023. 1

[22] David T Hoffmann, Dimitrios Tzionas, Michael J Black, and Siyu Tang. Learning to train with synthetic humans. In *German conference on pattern recognition*, pages 609–623. Springer, 2019. 2

[23] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian 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, 2013. 1, 2, 5

[24] Jingxia Jiang, Tian Ye, Jinbin Bai, Sixiang Chen, Wenhao Chai, Shi Jun, Yun Liu, and Erkang Chen. Five a^{+} net-

work: You only need 9k parameters for underwater image enhancement. *arXiv preprint arXiv:2305.08824*, 2023. 1

[25] Zhongyu Jiang, Zhuoran Zhou, Lei Li, Wenhao Chai, Cheng-Yen Yang, and Jenq-Neng Hwang. Back to optimization: Diffusion-based zero-shot 3d human pose estimation. *arXiv preprint arXiv:2307.03833*, 2023. 2

[26] Hanbyul Joo, Natalia Neverova, and Andrea Vedaldi. Exemplar fine-tuning for 3d human model fitting towards in-the-wild 3d human pose estimation. In *2021 International Conference on 3D Vision (3DV)*, pages 42–52. IEEE, 2021. 7

[27] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7122–7131, 2018. 7

[28] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5253–5263, 2020. 7

[29] Nikos Kolotouros, Georgios Pavlakos, Michael J Black, and Kostas Daniilidis. Learning to reconstruct 3d human pose and shape via model-fitting in the loop. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2252–2261, 2019. 6

[30] Jogendra Nath Kundu, Siddharth Seth, Pradyumna YM, Varun Jampani, Anirban Chakraborty, and R Venkatesh Babu. Uncertainty-aware adaptation for self-supervised 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20448–20459, 2022. 2

[31] Shichao Li, Lei Ke, Kevin Pratama, Yu-Wing Tai, Chi-Keung Tang, and Kwang-Ting Cheng. Cascaded deep monocular 3d human pose estimation with evolutionary training data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6173–6183, 2020. 2, 6, 7

[32] Kevin Lin, Lijuan Wang, and Zicheng Liu. Mesh graphormer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12939–12948, 2021. 7

[33] Yun Liu, Zhongsheng Yan, Sixiang Chen, Tian Ye, Wenqi Ren, and Erkang Chen. Nighthazeformer: Single nighttime haze removal using prior query transformer. *arXiv preprint arXiv:2305.09533*, 2023. 1

[34] Yun Liu, Zhongsheng Yan, Aimin Wu, Tian Ye, and Yuche Li. Nighttime image dehazing based on variational decomposition model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 640–649, 2022. 1

[35] Chenxu Luo, Xiao Chu, and Alan Yuille. Orinet: A fully convolutional network for 3d human pose estimation. *arXiv preprint arXiv:1811.04989*, 2018. 7

[36] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017. 5

[37] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 2640–2649, 2017. 1, 2, 3, 6, 7

[38] Dushyant Mehta, Helge Rhodin, Dan Casas, Pascal Fua, Oleksandr Sotnychenko, Weipeng Xu, and Christian Theobalt. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, pages 506–516. IEEE, 2017. 1, 5, 7

[39] Dushyant Mehta, Oleksandr Sotnychenko, Franziska Mueller, Weipeng Xu, Srinath Sridhar, Gerard Pons-Moll, and Christian Theobalt. Single-shot multi-person 3d pose estimation from monocular rgb. In *2018 International Conference on 3D Vision (3DV)*, pages 120–130. IEEE, 2018. 2, 7

[40] Dushyant Mehta, Srinath Sridhar, Oleksandr Sotnychenko, Helge Rhodin, Mohammad Shafiei, Hans-Peter Seidel, Weipeng Xu, Dan Casas, and Christian Theobalt. Vnect: Real-time 3d human pose estimation with a single rgb camera. *Acm transactions on graphics (tog)*, 36(4):1–14, 2017. 2, 7

[41] Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 3d human pose estimation in video with temporal convolutions and semi-supervised training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7753–7762, 2019. 1, 2, 3, 6, 7

[42] Grégory Rogez and Cordelia Schmid. Mocap-guided data augmentation for 3d pose estimation in the wild. *Advances in neural information processing systems*, 29, 2016. 2

[43] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 5

[44] Liangchen Song, Gang Yu, Junsong Yuan, and Zicheng Liu. Human pose estimation and its application to action recognition: A survey. *Journal of Visual Communication and Image Representation*, 76:103055, 2021. 1

[45] Chi Su, Jianing Li, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian. Pose-driven deep convolutional model for person re-identification. In *Proceedings of the IEEE international conference on computer vision*, pages 3960–3969, 2017. 1

[46] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep high-resolution representation learning for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5693–5703, 2019. 1

[47] Mikael Svenstrup, Soren Tranberg, Hans Jorgen Andersen, and Thomas Bak. Pose estimation and adaptive robot behaviour for human-robot interaction. In *2009 IEEE International Conference on Robotics and Automation*, pages 3571–3576. IEEE, 2009. 1

[48] Workman *et al.* Deepfocal: A method for direct focal length estimation. In *ICIP*, 2015. 2

[49] Gul Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J Black, Ivan Laptev, and Cordelia Schmid.

Learning from synthetic humans. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 109–117, 2017. 2

[50] Timo Von Marcard, Roberto Henschel, Michael J Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 601–617, 2018. 5, 6

[51] Bastian Wandt, Hanno Ackermann, and Bodo Rosenhahn. A kinematic chain space for monocular motion capture. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, pages 0–0, 2018. 4

[52] Bastian Wandt and Bodo Rosenhahn. Repnet: Weakly supervised training of an adversarial reprojection network for 3d human pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7782–7791, 2019. 2, 4, 7

[53] Jinbao Wang, Shujie Tan, Xiantong Zhen, Shuo Xu, Feng Zheng, Zhenyu He, and Ling Shao. Deep 3d human pose estimation: A review. *Computer Vision and Image Understanding*, 210:103225, 2021. 2

[54] Zhe Wang, Daeyun Shin, and Charless C Fowlkes. Predicting camera viewpoint improves cross-dataset generalization for 3d human pose estimation. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020. 2

[55] Garrett Wilson and Diane J Cook. A survey of unsupervised deep domain adaptation. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 11(5):1–46, 2020. 2

[56] Wei Yang, Wanli Ouyang, Xiaolong Wang, Jimmy Ren, Hongsheng Li, and Xiaogang Wang. 3d human pose estimation in the wild by adversarial learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5255–5264, 2018. 2, 5

[57] Tian Ye, Sixiang Chen, Yun Liu, Yi Ye, Jinbin Bai, and Erkang Chen. Towards real-time high-definition image snow removal: Efficient pyramid network with asymmetrical encoder-decoder architecture. In *Proceedings of the Asian Conference on Computer Vision*, pages 366–381, 2022. 1

[58] Tian Ye, Sixiang Chen, Yun Liu, Yi Ye, Erkang Chen, and Yuche Li. Underwater light field retention: Neural rendering for underwater imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 488–497, 2022. 1

[59] Tian Ye, Yunchen Zhang, Mingchao Jiang, Liang Chen, Yun Liu, Sixiang Chen, and Erkang Chen. Perceiving and modeling density for image dehazing. In *European Conference on Computer Vision*, pages 130–145. Springer, 2022. 1

[60] Ailing Zeng, Xiao Sun, Fuyang Huang, Minhao Liu, Qiang Xu, and Stephen Lin. Srnet: Improving generalization in 3d human pose estimation with a split-and-recombine approach. In *European Conference on Computer Vision*, pages 507–523. Springer, 2020. 7

[61] Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. Mixste: Seq2seq mixed spatio-temporal encoder for 3d human pose estimation in video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13232–13242, 2022. 7

[62] Zhenyu Zhang, Wenhao Chai, Zhongyu Jiang, Tian Ye, Mingli Song, Jenq-Neng Hwang, and Gaoang Wang. Mpm: A unified 2d-3d human pose representation via masked pose modeling. *arXiv preprint arXiv:2306.17201*, 2023. 2

[63] Long Zhao, Xi Peng, Yu Tian, Mubbasir Kapadia, and Dimitris N Metaxas. Semantic graph convolutional networks for 3d human pose regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3425–3435, 2019. 6, 7

[64] Zhonghan Zhao, Wenhao Chai, Shengyu Hao, Wenhao Hu, Guanhong Wang, Shidong Cao, Mingli Song, Jenq-Neng Hwang, and Gaoang Wang. A survey of deep learning in sports applications: Perception, comprehension, and decision. *arXiv preprint arXiv:2307.03353*, 2023. 1