FlexPose: Pose Distribution Adaptation with Limited Guidance

Zixiao Wang[1] **Junwu Weng**[2*] **Mengyuan Liu**[3] **Bei Yu**[1*]

[1]The Chinese University of Hong Kong [2]ByteDance Inc. [3]Peking University
zxwang22@cse.cuhk.edu.hk, we0001wu@e.ntu.edu.sg, nkliuyifang@gmail.com, byu@cse.cuhk.edu.hk

Abstract

Numerous well-annotated human key-point datasets are publicly available to date. However, annotating human poses for newly collected images is still a costly and time-consuming progress. Pose distributions from different datasets share similar pose hinge-structure priors with different geometric transformations, such as pivot orientation, joint rotation, and bone length ratio. The difference between Pose distributions is essentially the difference between the transformation distributions. Inspired by this fact, we propose a method to calibrate a pre-trained pose generator in which the pose prior has already been learned to an adapted one following a new pose distribution. We treat the representation of human pose joint coordinates as skeleton image and transfer a pre-trained pose annotation generator with only a few annotation guidance. By fine-tuning a limited number of linear layers that closely related to the pose transformation, the adapted generator is able to produce any number of pose annotations that are similar to the target poses. We evaluate our proposed method, FlexPose, on several cross-dataset settings both qualitatively and quantitatively, which demonstrates that our approach achieves state-of-the-art performance compared to the existing generative-model-based transfer learning methods when given limited annotation guidance.

1 Introduction

Deep neural networks are data-hungry and rely on large-scale datasets with high-quality human annotations for training. However, the process of annotating these datasets can be expensive and time-consuming, particularly when dense annotation are required, as is often the case in pose estimation tasks (Wang and Zhang 2022; He et al. 2017). To overcome this challenge, AI-aided labeling methods have become increasingly popular, where a pre-trained model's prediction serves as a reference to reduce human workload. However, when there is a domain shift (Luo et al. 2019), where the distribution of the training dataset and test dataset are not aligned not only on the input image domain but on the pose annotation domain as well, the accuracy of the model can significantly decline.

Considerable efforts have been devoted to tackling this issue. Among them, domain adaptation (DA) (Daumé III 2009; Csurka 2017) introduces knowledge from existing annotated datasets to a target dataset and is verified effective on several computer vision tasks (Cao et al. 2019; Inoue et al. 2018). However, things become different in the human-related dataset, *e.g.*, human pose (Ionescu et al. 2014) and human face (Wu et al. 2018). As the source human appearance is usually required in DA-related methods for input image domain adaptation, they may import unexpected data distribution bias (Buolamwini and Gebru 2018), *e.g.*, gender or color, from the source. Besides, the direct exposure of private portraits may raise the privacy issue.

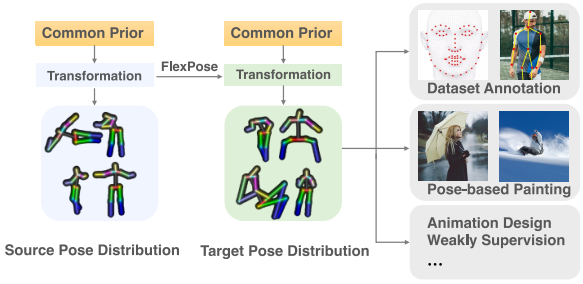

Figure 1: The illustration of how poses can be adapted between different domains. Although various pose datasets may differ in their transformations, they share a common hinge-structure prior. FlexPose's adaptation process focused on transformation, and the resulting poses can be effectively used in a wide range of downstream pose-related tasks.

On the other hand, it is commonly observed that different human poses share a similar hinge-structure prior. Typically, poses in a target dataset can be transferred from poses of a pre-collected source set by applying geometric transformations on for example pivot orientation, joint rotation, and bone length ratio. Therefore, adapting the pose distribution *only* can be a viable option in the case of human-related datasets. Pose Domain Adaptation (PDA) avoids the direct use of human appearance images, effectively addressing the aforementioned issues. Motivated by this observation, we propose FlexPose (shown in Figure 1), a method that transfers the source pose distribution to a target distribution with limited pose annotation guidance. After pose distribution transfer by FlexPose, each *input image* can be matched with the most related *generated pose* in estimated pose distribution by uti-

*Corresponding Author

lizing a matching algorithm (Jakab et al. 2020) for weakly supervised pose estimation and pose annotation. Besides, the generated poses can also be utilized in plenty of downstream tasks such as pose-conditioned image generation (Zhang and Agrawala 2023).

In FlexPose, we treat pose annotations as *skeleton images* to well align the annotations with their RGB appearance correspondences, and to improve the learnability of pose prior as the skeleton images well preserve the spatial structure of joint connection on image plane. We first learn the pose prior and fit the empirical distribution from a source human pose dataset by a multi-layer generative model. Thereafter, specific layers of the generative model are calibrated by inserting learnable lightweight linear modules to transfer the source distribution to the target domain. Considering that only a limited number of poses are given, we introduce three regularizations to avoid the collapse of the transfer solution. By generating credible pose interpolations with *Pose-mixup* regularization and by strictly limiting the complexity of the transfer module with linear and sparse regularization, we minimize the requirement of sample amount but maximize the sample diversity in FlexPose. FlexPose is computation-efficient. It operates on the pose domain, and hence the training convergence is much faster than methods in domain adaptation, which focuses adaptation on both the pose and image domains together. FlexPose is also data-efficient. We only need limited pose annotations from the target dataset to finetune the transfer modules. Extensive experiments on three pose-based tasks, *i.e.,* human pose annotation, human face landmarks annotation, and pose-conditional human image generation, demonstrate that FlexPose outperforms baselines by a large margin both quantitatively and qualitatively. Our contributions can be summarized as follows:

- We propose to treat the task of Pose Domain Adaptation as the transfer of skeleton image generator and demonstrate that a target pose distribution can be well approximated from a well-learned pose prior.
- We introduce FlexPose, a PDA framework that employs three regularizations to efficiently transfer a pose distribution to a target one by utilizing a limited number of guiding poses with low computation and storage costs.
- Extensive experimental results on three pose-related tasks show that FlexPose achieves remarkable improvement over existing methods.

2 Related Works

Deep Generative Model for Image Generation. Deep generative models such as GAN, Variational AutoEncoder (VAE), and Diffusion models achieve great success in realistic/artificial image generating and natural image distribution modeling. Recently proposed generative models such as StyleGAN (Karras, Laine, and Aila 2019), DDPM (Ho, Jain, and Abbeel 2020), NVAE (Vahdat and Kautz 2020) introduce new mechanisms, new architecture, and new regularizations into image generation. VAEs (Kingma and Welling 2014) learn to maximize the variational lower bound of likelihood. Diffusion probabilistic models (Sohl-Dickstein et al. 2015) synthesize images by a denoising procedure. GANs (Goodfellow et al. 2014) are trained in an adversarial manner to learn how to generate realistic images. Among them, Karras *et al.* (Karras, Laine, and Aila 2019) proposed an architecture StyleGAN that can learn a hierarchical decoupled style code and controls image synthesis. Our method is based on generators with multi-layer architecture and leverages StyleGAN as the backbone. We are inspired by the recent works (Zhu et al. 2016; Yin et al. 2022), which manipulate the latent code in the generative model to edit the output images. These works motivate us to transfer the pose distribution to the target domain by transferring style codes with few-shot guidance.

Transfer Learning for Generative Models. The literature on transfer learning has been extensively studied in recent years (Oquab et al. 2014; Long et al. 2015; Ganin and Lempitsky 2015). Transfer learning learns to transfer the knowledge from a large-scale source dataset to a small target dataset to enhance model performance on the target dataset. The methodology of transfer learning is also treated as a pre-training technique. It is utilized to accelerate the learning on the target dataset. (Wang et al. 2018) finetunes a pre-trained GAN on a target dataset to get better performance. (Noguchi and Harada 2019) transfers knowledge from a large dataset to a small dataset by re-computing batch statistics. Existing methods focus on either the image domain or the neural language processing domain (Shin, Hwang, and Sung 2016). For these methods, hundreds of training samples are still required. Compared with these approaches, we focus on pose domain adaptation, and our method only requires few-shot guidance for transferring. After LoRA (Hu et al. 2021) is widely used in Large Language Model finetuning, the researchers in Content Generation are inspired to introduce or extend this technique in generative model (Mou et al. 2024). Compared with the light-weight but global model finetuning in LoRA, FlexPose only focuses on calibrating specific layers with semantics of pose geometric transformation locally to satisfy the linear and sparse finetuning requirements.

Human Pose Estimation. 2D Human pose estimation is a task that predicts the 2D pose from a single image. Fully-supervised methods (Andriluka, Roth, and Schiele 2009; Bai and Wang 2019; Belagiannis and Zisserman 2017) utilize large-scale annotated datasets such as COCO (Lin et al. 2014), Human3.6M (Ionescu et al. 2014) and 3DHP (Mehta et al. 2017) for model training. Weakly-supervised (Kanazawa et al. 2018; Gecer et al. 2019; Geng, Cao, and Tulyakov 2019; Wang et al. 2023) and unsupervised (Shu et al. 2018; Jakab et al. 2018) methods such as KeypointGAN (Jakab et al. 2020) have been proposed to reduce the dependence on the expensive pose annotation. These methods require supervised post-training or additional prior knowledge to generate meaningful landmarks, which can serve as a distance measurement between the provided prior knowledge and the target distribution. To match poses generated by FlexPose with unlabelled images in the target dataset, we employ an unsupervised method (Jakab et al. 2020) in addition to supervision from adversarial training. This matching procedure serves as an evaluation method for FlexPose and is further detailed in Section 4. Recently, test-time adaptation (Li et al. 2021; Cui et al. 2023; Hu et al. 2024) has proven to be an effective way

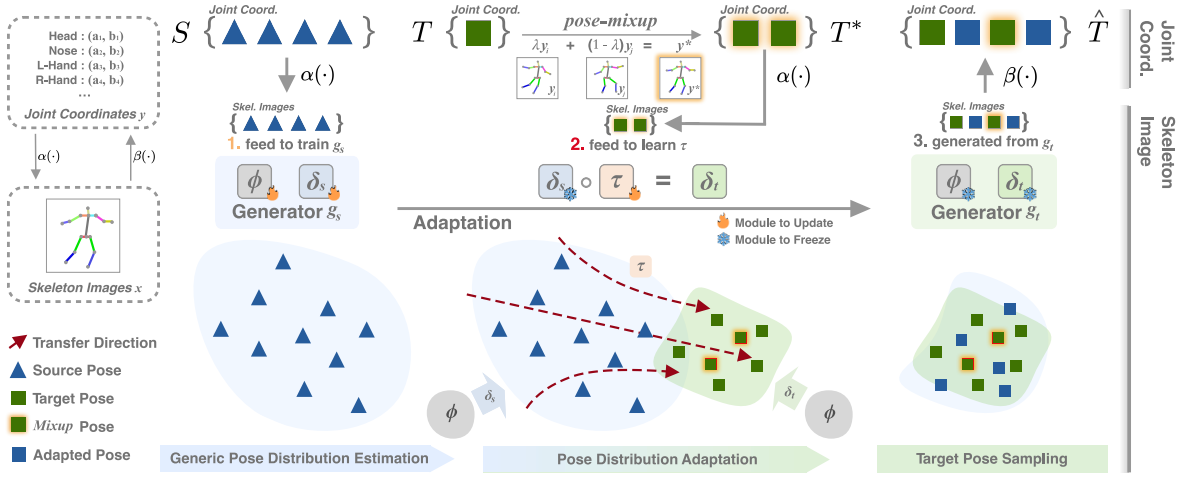

Figure 2: An illustration of the FlexPose framework for pose distribution adaptation. There are three main steps in our framework: ① We train a skeleton image generator to learn the pose prior from the source pose distribution; ② The source generator is transferred to a target generator with limited target pose guidance to achieve pose distribution adaptation; ③ We utilize the target generator to generate target pose annotations for downstream tasks.

to deal with domain shift in pose estimation. It utilizes self-supervised learning during inference to adapt model to the input human appearance distribution. Compared with FlexPose which focuses on PDA, Test-time adaptation tackle the issues in input image domain shift.

3 Method

Given a limited number of 2D pose annotations set $T = \{\boldsymbol{y}_t | \boldsymbol{y}_t \in \mathbb{R}^{M\times 2}\}$ of a newly collected human pose images, FlexPose aims to estimate the whole distribution $\mathcal{D}_t$ which pose annotation T belongs to, and generate any number of new pose annotations that follow the distribution. M here is the number of joints in each pose. This task setting is challenging. However, we believe that with the prior from sufficient off-the-shelf annotations $S = \{\boldsymbol{y}_s | \boldsymbol{y}_s \in \mathbb{R}^{M\times 2}\}$, the distribution $\mathcal{D}_t$ can be estimated and well shaped. In this paper, we transfer the distribution $\mathcal{D}_s$ estimated from S to the target pose domain to estimate the target distribution $\mathcal{D}_t$ by considering the guidance from 2D pose annotations set T.

3.1 Overview

As illustrated in Figure 2, our framework consists of three phases: ① **Generic Pose Distribution Estimation**. We learn a generator $g_s(\cdot)$ on the pose set S to estimate the pose distribution $\mathcal{D}_s$. The generator takes a latent code $\boldsymbol{z}$ as input and outputs a skeleton $\hat{\boldsymbol{x}}_s$, *i.e.* $\hat{\boldsymbol{x}}_s = g_s(\boldsymbol{z})$. We take the distribution of generated $\hat{\boldsymbol{x}}_s$ to mimic that of the generic pose $\boldsymbol{x}_s$. Here, $\boldsymbol{x}$ is the corresponding *skeleton image* of an annotation $\boldsymbol{y}$ as shown in the left part of Figure 2. ② **Pose Distribution Adaptation**. Given the limited target annotation set T, we transfer $g_s(\cdot)$ to fit the pose distribution $\mathcal{D}_t$ and learn a new generator $g_t(\cdot)$ of the target pose domain. Considering the limited knowledge acquired from target pose annotation T, we introduce three regularizations, *Linear, Sparse* and *Pose-mixup*, to avoid reaching a collapse solution. ③ **Target Pose Sampling**. The transferred generator $g_t(\cdot)$ can flexibly synthesize any number of fake pose annotations by randomly sampling in the latent space. This generated annotation set $\hat{T}$ will be treated as an extension of given annotations set T in the downstream tasks, *e.g.,* Keypoints Annotation and Pose-conditional Human Image Generation, since poses within both of them follow the distribution $\mathcal{D}_t$.

3.2 Generic Pose Distribution Estimation

Deep generative models have been widely verified that they have a rich capacity to well approximate image distributions when given sufficient training data. Motivated by the success of these generative models (Karras, Laine, and Aila 2019) on natural/artificial image generation, we treat 2D pose annotations $\boldsymbol{y}_s, \boldsymbol{y}_t \in \mathbb{R}^{M\times 2}$ as skeleton images $\boldsymbol{x}_s, \boldsymbol{x}_t \in \mathbb{R}^{C\times W\times H}$ and extend an image generator to generate 2D pose annotations by synthesizing corresponding skeleton images. As shown in the left part of Figure 2, the transformation from the 2D keypoints to the skeleton images can be implemented by functions $\alpha(\cdot)$, namely $\boldsymbol{x} = \alpha(\boldsymbol{y})$, where $\alpha(\cdot)$ simply draws keypoints from $\boldsymbol{y}$ and connects them with straight lines on a blank figure. The visual effect is similar to the stick man. To achieve precise semantic alignment with the appearance correspondence, each bone in the skeleton image is assigned a unique color. Therefore, C of each skeleton image is set as three (RGB channels). Compared with Black&White, the colorful embedding brings marginal improvement in the quality of generated skeletons.

A generator can be formulated as a mapping function $g(\cdot)$, which gets a latent code $\boldsymbol{z}$ and outputs a skeleton image $\boldsymbol{x}$. The probability distribution of skeleton images hence is estimated by $p(\boldsymbol{x}) = p(\boldsymbol{z})p_g(\boldsymbol{x}|\boldsymbol{z})$. We assume that the pose distributions of different datasets share similar pose prior, and their distributions can transfer to one another by geometric transformations. Based on this assumption, we

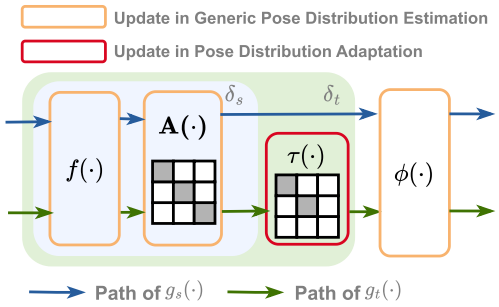

Figure 3: An illustration of the generator decomposition. We use $\tau(\cdot)$ to adjust the source generator for pose distribution adaptation.

further factorize the generator $g(\cdot)$ as $g = \phi \circ \delta$. Therefore, the source skeleton image generator can be formulated as

$$p(\hat{\boldsymbol{x}}_s) = p(\boldsymbol{z})\; p_g^s(\hat{\boldsymbol{x}}_s|\boldsymbol{z}) = p(\boldsymbol{z})\; p_\delta^s(\boldsymbol{h}_s|\boldsymbol{z})\; p_\phi(\hat{\boldsymbol{x}}_s|\boldsymbol{h}_s), \quad (1)$$

in which $\phi(\cdot)$ preserves the learned pose prior and $\delta(\cdot)$ records the mapping from the learned prior $\boldsymbol{h}$ to the skeleton image $\boldsymbol{x}_s$ of a certain pose domain. Similarly, the distribution of target domain can be formulated as $p(\hat{\boldsymbol{x}}_t) = p(\boldsymbol{z})\; p_\delta^t(\boldsymbol{h}_t|\boldsymbol{z})\; p_\phi(\hat{\boldsymbol{x}}_t|\boldsymbol{h}_t)$. With the prior sharing assumption, the pose distribution adaptation aims at transferring pretrained conditional probability $p_\delta^s(\boldsymbol{h}_s|\boldsymbol{z})$ to $p_\delta^t(\boldsymbol{h}_t|\boldsymbol{z})$ with guidance from the pose annotation set T:

$$p_\delta^s(\boldsymbol{h}_s|\boldsymbol{z}) \xrightarrow{T} p_\delta^t(\boldsymbol{h}_t|\boldsymbol{z}). \quad (2)$$

Considering the ability of StyleGAN in separating high-level attributes and in the interpolation between these attributes, we utilize StyleGAN network architecture to disentangle the pose prior and the transformation of the source skeleton image generator,

$$g_s = \phi \circ \delta_s = \phi \circ (A \circ f)_s, \quad (3)$$

where $f(\cdot)$ is a non-linear mapping that takes random noise as input and outputs a random vector. $A(\cdot)$ is a learned affine transformation and can be treated as a block diagonal matrix with L blocks, where L is the number of layers. The output of $A(\cdot)$ is the style code to modulate the synthesis network $\phi(\cdot)$ by adaptive instance normalization. Due to the ability of StyleGAN in style control, we can directly adapt the distribution of source skeleton image to the target domain by adjusting the style code.

3.3 Pose Distribution Adaptation

As illustrated in Figure 3, to transfer $p_\delta^s(\boldsymbol{h}_s|\boldsymbol{z})$ to $p_\delta^t(\boldsymbol{h}_t|\boldsymbol{z})$, we adjust the style code by introducing a transfer function $\tau(\cdot)$ at the output of $\delta(\cdot)$, and therefore the target domain generator is defined as

$$g_t = \phi \circ \delta_t = \phi \circ (\tau \circ \delta_s). \quad (4)$$

To learn the transfer function τ, we first randomly sample $|T|$ latent codes $\boldsymbol{z}$ (one for each pose in T) from the latent space, and require the generator maps each code to corresponding skeletons in T. This transferring procedure can be achieved by minimizing the following perceptual loss,

$$\min_{\theta_\tau} \mathcal{L}_{s\to t} = \min_{\theta_\tau} \sum \left\| \Gamma\big(g_t(\boldsymbol{z}); \theta_\tau\big) - \Gamma\big(\boldsymbol{x}\big) \right\|_2^2, \quad (5)$$

where θ_τ is the parameter of $\tau(\cdot)$, Γ is a pre-trained feature extractor, $\boldsymbol{z}$ is from the set of sampled latent codes, and $\boldsymbol{x}$ is the skeleton image drawn from the pose annotation set T.

However, the problem is we only have few-shot guidance T from the target domain distribution. Given a data-starving deep learning model, the guidance is insufficient to reach a satisfactory solution. For that reason, we introduce three regularizations to alleviate the data-insufficient issue.

Linear & Sparse Regularization. Compared with finetuning the whole transformation function δ_s to reach δ_t, only adjusting the affine transformation from A_s to A_t, *i.e.* $A_t = \tau \circ A_s$, can efficiently shrink the searching space of transfer solution, and therefore avoid overfitting. Meanwhile, the recent GAN inversion technique shows that the layer-wise style code in StyleGAN leads to the hierarchical disentanglement of local and global attributes, which aligns well with our motivation of adapting pose distribution by considering the global geometric transformation between poses. We thus adjust the source affine transformation A_s from the perspective of layer level, and limit the number of to-be-adjusted layers as small as possible. Considering the form of the affine transformation A and the layer decoupling characteristics of StyleGAN, we empirically define the transfer function $\tau(\cdot)$ as a block diagonal matrix,

$$\tau \triangleq \mathrm{diag}(\boldsymbol{I}, ..., \boldsymbol{I}, \boldsymbol{U}_{l_0}, \boldsymbol{I}, ..., \boldsymbol{I}, \boldsymbol{U}_{l_1}, \boldsymbol{I}, ..., \boldsymbol{I}), \quad (6)$$

where only a limited number of block is defined by $\boldsymbol{U}$, *i.e.* l_0 and l_1 in this case, to follow the sparse regularization. We experimentally find that the earlier layers are most related to the geometric transformation. And we only learn those layers in our experiments. Meanwhile, other blocks are set as identity matrix $\boldsymbol{I}$. We investigated how the choice of layer l affects the transformation procedure in Section 4.4.

Pose-mixup Regularization. Most poses interpolated between two real poses physically exist, and their convex combinations build the real-world pose distribution. Inspired by the *mixup* regularization (Zhang et al. 2017) on images, we therefore extend it to 2D pose annotations and propose the *Pose-mixup* to enrich the guidance set. The main difference between *mixup* and *Pose-mixup* is that the *mixup* works on *image* space and the *Pose-mixup* works on *keypoint* space. Given that the mixup on skeleton space may lead to unreasonable results, *Pose-mixup* regularizes the neural network to learn the simple linear behavior in-between 2D poses and thus prevents the model from generating unrealistic human pose annotations. By mixing up the corresponding joints of any two 2D poses with mixup ratio $\lambda \in [0, 1]$, the extended annotation set T^* from T is then defined as,

$$T^* = \{\boldsymbol{y}^* \mid \boldsymbol{y}^* = \lambda\, \boldsymbol{y}_i + (1-\lambda)\, \boldsymbol{y}_j, \quad \boldsymbol{y}_i, \boldsymbol{y}_j \in T\}. \quad (7)$$

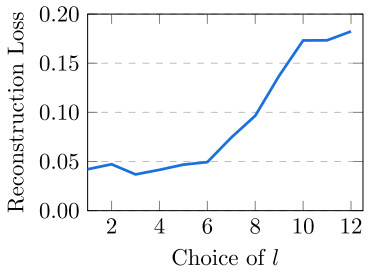

Figure 4: Reconstruction loss with different choice of layer l.

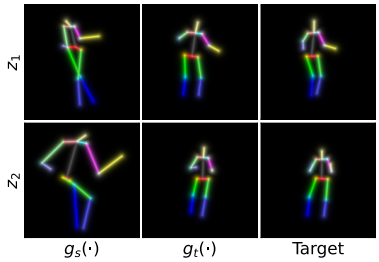

Figure 5: Visualization of pose adaptation. The left and middle of each row are generated from the same random noise. The middle aims to mimic the right.

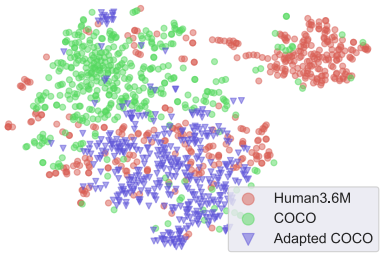

Figure 6: t-SNE visualization of human poses before and after adaptation. We visualize the pose distribution ($\mathbb{R}^{M\times 2}$) in a two-dimensional space.

Method	Target	Source	MMD^2 (↓)	FD (↓)
FreezeD	H3.6M	COCO	0.081	3.77
AdaGAN		COCO	0.052	2.67
LoRA		COCO	0.035	1.36
FlexPose		COCO	**0.029**	**0.80**

Table 1: The results of distribution distance measurement.

3.4 Target Pose Sampling

Once the transferred generator $g_t(\cdot)$ is obtained, we can generate theoretically as many target skeleton images $\tilde{\boldsymbol{x}}_t$ as possible by randomly sampling latent codes in the estimated target distribution $\mathcal{D}_t$. Unfortunately, the generated target skeleton images are not perfect and may bring in artifacts which mislead the training of a neural network. To address this issue, we utilize $\beta(\cdot)$ to filter out the random noise. $\beta(\cdot)$ is a neural network regressor pre-trained on S and acts as a tight information bottleneck that preserves skeleton information and ignores random noise. Following the generation of the fake skeleton images $\tilde{\boldsymbol{x}}_t$ from $g_t(\cdot)$, we extract the coordinates of interpretable 2D keypoints $\hat{T} = \{\hat{\boldsymbol{y}}_t\}$ from it, *e.g.*, hands, by applying $\hat{\boldsymbol{y}}_t = \beta(\tilde{\boldsymbol{x}}_t)$. Thereafter, we can get a clean generated skeleton $\hat{\boldsymbol{x}}_t$ by a re-render process,

$$\hat{\boldsymbol{x}}_t = \alpha(\hat{\boldsymbol{y}}_t) = \alpha\big(\ \beta(\tilde{\boldsymbol{x}}_t)\ \big). \tag{8}$$

These generated skeleton images and the corresponding generated 2D pose annotations can be further utilized to assist any pose-related down-stream tasks.

4 Experiments

In this section, we first evaluate the distribution similarity between transferred distribution and target distribution via two standard metrics. Then, we show how FlexPose can improve the performance of existing unsupervised landmark detection algorithms and benefit unlabelled human pose dataset annotation. At last, we extensively discussed how each part of FlexPose works.

4.1 Pose Distribution Transformation

In this subsection, we conduct a transformation experiment between COCO (Lin et al. 2014) and Human3.6M (Ionescu et al. 2014) to show how FlexPose works.

Experiment Setting. We train a StyleGAN (Karras, Laine, and Aila 2019) using the skeleton images from the source datasets to estimate the distribution of source human pose. And then we transform the estimated distribution to the target one according to several samples from target dataset. For source dataset COCO, we only keep the annotated people instances with full pose annotations to construct a training set of 32k samples. The training of StyleGAN follows standard protocol in the original work. In the transformation phase, we only use 30 annotations from the target dataset Human3.6M (two for each class). The size of interpolated pose set ($|T^*|$) is set as 1000. We experimented with changing different layers and found that setting l=3, *i.e.,* transferring the third coarsest layer, usually gets the lowest reconstruction loss in Equation (5) as shown in Figure 4. So, we set l=3 in all experiments. A detailed setting and deeper analysis can be found in appendix. When adaptation phase ends, we sample new poses from generator and treat them as Adapted COCO.

Evaluation & **Visualization**. Qualitatively, we show the visual result of pose transformation in Figure 5. For each row, we show one skeleton (Left) that was randomly sampled from the generator before transformation, one skeleton (Middle) that was sampled from the transformed generator by using the same latent noise as the Left, and one skeleton (Right) in the few-shot annotation set T from target dataset. We can see that the Left and the Middle generated from the same random noise are visually quite different, and the Middle is more similar to the Right.

In Figure 6, we plot the t-SNE embedding of the poses generated by FlexPose (Adapted COCO), comparing it with the embedding of poses from the source (COCO) and target (Human3.6M) dataset. As can be seen, the embedding of poses from the source and target dataset are separated, and the distribution of generated poses significantly overlaps with the target ones. We also noted that the pose distributions are 'mismatched' in the upper right region. Considering that only two shots are utilized as guidance for each class during the transformation, such a mismatch is reasonable.

Quantitatively, we measure the similarity between the transferred distribution and target distribution using the Fréchet distance (FD), which follows from the Wasserstein-

Method	Target	Source	MSE (↓)	PCK (↑)
Baseline	H3.6M	COCO	17.86	0.015
FreezeD		COCO	20.60	0.081
AdaGAN		COCO	14.88	0.395
LoRA		COCO	13.85	0.430
FlexPose		COCO	**13.19**	**0.585**
Baseline	S-H3.6M	COCO	5.47	0.685
FreezeD		COCO	7.63	0.003
AdaGAN		COCO	5.36	0.455
LoRA		COCO	5.02	0.512
FlexPose		COCO	**3.79**	**0.770**
Baseline		3DHP	12.66	0.000
AdaGAN		3DHP	7.23	0.215
FreezeD		3DHP	6.28	0.206
LoRA		3DHP	6.15	0.314
FlexPose		3DHP	**5.98**	**0.467**
Baseline		SURREAL	11.18	0.000
FreezeD		SURREAL	11.38	0.006
AdaGAN		SURREAL	6.63	0.228
LoRA		SURREAL	6.52	0.337
FlexPose		SURREAL	**6.47**	**0.499**

Table 2: Results on human pose annotation task. S-H3.6M and H3.6M are short for Simplified Human3.6M dataset and Human3.6M dataset respectively. The threshold of PCK is 10% for S-H3.6M and 20% for H3.6M in this table.

Method	Target	Source	MSE (↓)	PCK (↑)
Baseline	WFLW	300-VW	18.78	0.679
AdaGAN		300-VW	11.95	**0.785**
FreezeD		300-VW	11.66	0.779
LoRA		300-VW	11.77	0.760
FlexPose		300-VW	**11.64**	0.766

Table 3: Results on human face annotation task.

based definition of FID (Heusel et al. 2017) without the application of the pre-trained Inception network. We also measure the square of Maximum Mean Discrepancy (MMD) to provide more insights. The measurements are conducted on the keypoint coordinates space.

We compare our method with three strong competitors. AdaGAN (Noguchi and Harada 2019), FreezeD (Mo, Cho, and Shin 2020) and LoRA (Hu et al. 2021) (rank r=8). Both AdaGAN and FreezeD suggest finetuning-based strategies with regularization. LoRA is the most related work to our FlexPose, introducing low-rank regularization to the generative model. For all methods, we generate 50k samples from the transferred generative model and compare them with all samples in the target dataset Human3.6M. In Table 1, experiment results suggest that FlexPose gives superior performance on both MMD and FD evaluation, indicating the transferred distribution shares more similar characteristics to the target distribution. The finding remains the same as that of the observation on qualitative evaluation.

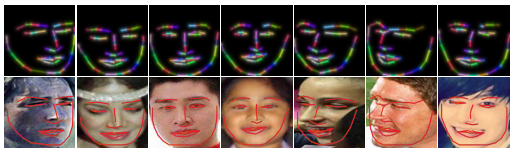

Figure 7: Visualization of human face landmark annotation on WFLW dataset. Upper Row: landmarks given by matching algorithm. Bottom Row: landmarks in the upper row with their corresponding human faces.

4.2 Unlabelled Human Pose Dataset Annotation

To further evaluate the quality of transformed pose quantitatively and show potential downstream application of FlexPose, we show how can we annotated unlabelled human-related dataset with the help of FlexPose.

Pose-Image Matching Algorithm. In dataset annotation tasks, our goal is to assign each image in the target dataset to the most closely related pose in the estimated target distribution $\mathcal{D}_t$. Existing self-supervised human pose detection methods (Jakab et al. 2018; Lorenz et al. 2019; Thewlis, Bilen, and Vedaldi 2017) are usually constrained by the high relevance between model prediction and input image. Among them, KeypointGAN (Jakab et al. 2020) can match unpaired images and annotations by forcing the distribution of detector predictions to align with the existing poses. We train KeypointGAN by using human images from target dataset and generated pose set $\hat{T}$. Once the training process is completed, the model prediction on samples can be treated as the best-matched annotations in given distribution.

Source Datasets. Apart from COCO, we also use MPI-INF-3DHP (3DHP) (Mehta et al. 2017) which contains more than 1.8 million human pose annotations from eight subjects and covers eight complex exercise activities. SURREAL (Varol et al. 2017) is a synthetic dataset containing more than six million frames of people in motion.

Target Datasets. The large-scale dataset *Human3.6M* (Ionescu et al. 2014) has 3.6 million samples. The *Simplified Human3.6M* dataset (Zhang et al. 2018) contains 800k training and around 90k testing images. We use all human images in the target dataset and randomly select several guide poses.

Evaluation Metrics. Since dataset annotation shares similar targets with 2D landmark detection. We report 2D landmark detection performance for evaluation. Two standard evaluation metrics are considered to compare our method with baselines. The MSE column reports a mean square error in pixels overall pre-defined common joints. The Percentage of Correct Key-points (PCK-ρ) is used as an accuracy metric that measures if the distance between the predicted keypoint and the true joint is within a certain threshold ρ.

Performance Comparisons. We feed the generated skeleton images $\hat{\boldsymbol{x}}_t$ and RGB human images from target dataset into KeypointGAN to evaluate the effectiveness of each pose transformation algorithm. As a baseline, we train the detector on each target dataset by directly using the pose annotations set S from the source dataset, which we denote as ***Baseline*** in the comparison and can be roughly treated as the worst case. We also employ three strong competitors, AdaGAN,

#	Source	Layer	Mixup	Linear	Shots	MSE	PCK
1	**C**	3	√	√	12	3.79	0.77
2	**C**	1,3	√	√	12	3.82	0.78
3	**C**	3,5	√	√	12	4.02	0.61
4	**C**	ALL	√	√	12	4.50	0.66
5	**D**	3	√	√	12	5.98	0.44
6	**D**	ALL	×	√	12	9.82	0.01
7	**D**	ALL	×	×	12	12.32	0.00
8	**C**	3	√	√	12	3.79	0.77
9	**C**	3	√	√	24	3.80	0.75
10	**C**	3	√	√	48	3.73	0.70
11	**D**	3	√	√	12	5.98	0.47
12	**DC**	3	√	√	12	5.28	0.59
13	**DCS**	3	√	√	12	5.19	0.59

Table 4: Ablation study on human pose annotation. The target dataset is Simplified-Human3.6M for all experiments. **C**, **D**, **S** are short for COCO, 3DHP, SURREAL dataset.

FreezeD, and LoRA, for comparison.

Quantitatively, we compare their performance with FlexPose on human pose estimation in Table 2. As shown in Table 2, the ***Baseline*** has much lower performance on the target dataset as the pose annotations are from different datasets, especially when some of them have a distinct pose distribution from that of the target dataset, *e.g.,* when 3DHP or SURREAL is the source dataset and Simplified Human3.6M is the target one. FlexPose gets better results on all settings under both metrics. FlexPose largely reduces the performance gap when the pose distribution of the source dataset is very different from that of the target distribution, *e.g.,* MSE 12.7→6.0 and PCK10 0.00→0.47 when adaptation occurs from 3DHP to Simplified Human3.6M. The results show that FlexPose is effective at generating similar poses with the target dataset, even with less to only two poses per class in the target dataset.

4.3 Unlabelled Human Face Dataset Annotation

Introducing FlexPose to human face landmarks transfer is straightforward since both the human pose and the human face consist of a set of pre-determined keypoints.

Datasets. *WFLW* (Wu et al. 2018) has 10 thousand samples with 98 facial landmarks, where 7.5 thousand for training and 2.5 thousand for testing. *300-VW* (Sagonas et al. 2013) consists of 300 Videos in the wild and contains ∼95 thousand annotated human faces in the training set. We treat 300-VW as the source dataset and only use its annotations for training StyleGAN. And few-shot annotations in the target dataset WFLW are utilized for transformation. We only keep the shared 68 facial landmarks in two datasets.

Experiments Settings and Results. The evaluation metrics and the experiment protocols are the same as that in the human pose. The size of the few-shot guidance set from target dataset is set as 30. We report the evaluation results on the validation set of WFLW in Table 3. FlexPose still outperforms the baseline by a large margin (MSE $18.78 \rightarrow 11.64$ and PCK $0.679 \rightarrow 0.766$). Given that the human face can be treated as a rigid body approximately and are easier to transfer, FlexPose achieves comparable performance with previous SOTA methods, AdaGAN, FreezeD and LoRA.

We show the detected human face landmarks in Figure 7. The human face detector trained with generated face landmarks can handle human faces in different directions well.

4.4 Ablation Study & Parameter Sensitivity

In Table 4, ablation studies are conducted:

Effect of Regularization. We proposed three kinds of regularization in Section 3.3 to alleviate the extreme data-insufficient issue. We remove part of them from our FlexPose, and the results are #1 to #7. From #1 to #4, we gradually relax the sparsity regularization by allowing more blocks in the diagonal matrix τ not to be an identity matrix $\boldsymbol{I}$. The performance only drops by an acceptable level thanks to the Linear and Mixup regularization. Furthermore, in #5, #6, and #7, we further relax the Mixup and Linear regularization, which significantly hurt the quality of generated images and lower the model accuracy in downstream tasks.

Number of Shots from Target Dataset. Under the setting of COCO→S-H3.6M, we increase the number of shots from 12 to 48 and found that the performance of the pose detector has no obvious difference. The results can be found in #8 to #10. An explanation is that the increment of few-shot samples from the target dataset brings a limited gain of information compared with the strong prior trained on large-scale datasets. Few samples are enough for target distribution localization.

Choice of Layers l. In previous experiments, we empirically choose l=3 in Equation (4) for all experiments and get significant improvement. We found that the choice of l is not strictly fixed. We have also tried a composition of multi-layer, and the results can be found in #2, #3, and #4. The result in #4 shows the necessity of sparse regularization. We leave the best choice of l to future work.

Multi-source Datasets. To study the effect of the setting where the source annotations are from different datasets, we conduct two additional experiments (#12 and #13) in Table 4 and compare them with existing experiments (#11). In #12 and #13, we use the union of different source datasets to train the generic generator. The result indicates that the increasing diversity on the source dataset (#11→#12→#13) brings better results on the target dataset. By utilizing FlexPose, the performance of downstream task models can benefit from collecting a more diverse pose dataset, which is much easier compared with collecting a realistic human dataset with accurate landmarks. However, the result of #13 is still worse than that of #1, which indicates the trade-off between diversity and similarity to the target dataset when choosing the source.

5 Conclusion

We aim to transfer knowledge in the pose domain and propose an effective method named FlexPose. Our approach allows us to adapt an existing pose distribution to a different target one by using a few poses from the target dataset and generating theoretically infinite poses following the target distribution. FlexPose can be used on several pose-related works. In future work, we hope to extend our method to a more generic pose domain adaptation approach.

References

Andriluka, M.; Roth, S.; and Schiele, B. 2009. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, 1014–1021. IEEE.

Bai, Y.; and Wang, W. 2019. Acpnet: anchor-center based person network for human pose estimation and instance segmentation. In *ICME*, 1072–1077. IEEE.

Belagiannis, V.; and Zisserman, A. 2017. Recurrent human pose estimation. In *FG*, 468–475. IEEE.

Buolamwini, J.; and Gebru, T. 2018. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, 77–91. PMLR.

Cao, J.; Tang, H.; Fang, H.-S.; Shen, X.; Lu, C.; and Tai, Y.-W. 2019. Cross-domain adaptation for animal pose estimation. In *ICCV*, 9498–9507.

Csurka, G. 2017. Domain adaptation for visual applications: A comprehensive survey. *arXiv preprint arXiv:1702.05374*.

Cui, Q.; Sun, H.; Lu, J.; Li, W.; Li, B.; Yi, H.; and Wang, H. 2023. Test-time Personalizable Forecasting of 3D Human Poses. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 274–283.

Daumé III, H. 2009. Frustratingly easy domain adaptation. *arXiv preprint arXiv:0907.1815*.

Ganin, Y.; and Lempitsky, V. 2015. Unsupervised domain adaptation by backpropagation. In *ICML*, 1180–1189. PMLR.

Gecer, B.; Ploumpis, S.; Kotsia, I.; and Zafeiriou, S. 2019. Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction. In *CVPR*, 1155–1164.

Geng, Z.; Cao, C.; and Tulyakov, S. 2019. 3d guided fine-grained face manipulation. In *CVPR*, 9821–9830.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A. C.; and Bengio, Y. 2014. Generative Adversarial Nets. In *NeurIPS*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *ICCV*, 2961–2969.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 30.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *NeurIPS*, 33: 6840–6851.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Hu, S.; Sun, H.; Li, B.; Wei, D.; Li, W.; and Lu, J. 2024. Fast Adaptation for Human Pose Estimation via Meta-Optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1792–1801.

Inoue, N.; Furuta, R.; Yamasaki, T.; and Aizawa, K. 2018. Cross-domain weakly-supervised object detection through progressive domain adaptation. In *CVPR*, 5001–5009.

Ionescu, C.; Papava, D.; Olaru, V.; and Sminchisescu, C. 2014. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *TPAMI*, 36(7): 1325–1339.

Jakab, T.; Gupta, A.; Bilen, H.; and Vedaldi, A. 2018. Unsupervised learning of object landmarks through conditional image generation. *NeurIPS*, 31.

Jakab, T.; Gupta, A.; Bilen, H.; and Vedaldi, A. 2020. Self-supervised learning of interpretable keypoints from unlabelled videos. In *CVPR*, 8787–8797.

Kanazawa, A.; Black, M. J.; Jacobs, D. W.; and Malik, J. 2018. End-to-end recovery of human shape and pose. In *CVPR*, 7122–7131.

Karras, T.; Laine, S.; and Aila, T. 2019. A style-based generator architecture for generative adversarial networks. In *CVPR*, 4401–4410.

Kingma, D. P.; and Welling, M. 2014. Auto-encoding variational bayes. *ICLR*.

Li, Y.; Hao, M.; Di, Z.; Gundavarapu, N. B.; and Wang, X. 2021. Test-time personalization with a transformer for human pose estimation. *Advances in Neural Information Processing Systems*, 34: 2583–2597.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*, 740–755. Springer.

Long, M.; Cao, Y.; Wang, J.; and Jordan, M. 2015. Learning transferable features with deep adaptation networks. In *ICML*, 97–105. PMLR.

Lorenz, D.; Bereska, L.; Milbich, T.; and Ommer, B. 2019. Unsupervised part-based disentangling of object shape and appearance. In *CVPR*, 10955–10964.

Luo, Y.; Zheng, L.; Guan, T.; Yu, J.; and Yang, Y. 2019. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *CVPR*, 2507–2516.

Mehta, D.; Rhodin, H.; Casas, D.; Fua, P.; Sotnychenko, O.; Xu, W.; and Theobalt, C. 2017. Monocular 3d human pose estimation in the wild using improved cnn supervision. In *2017 international conference on 3D vision (3DV)*, 506–516. IEEE.

Mo, S.; Cho, M.; and Shin, J. 2020. Freeze the discriminator: a simple baseline for fine-tuning gans. *CVPR Workshop*.

Mou, C.; Wang, X.; Xie, L.; Wu, Y.; Zhang, J.; Qi, Z.; and Shan, Y. 2024. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 4296–4304.

Noguchi, A.; and Harada, T. 2019. Image generation from small datasets via batch statistics adaptation. In *ICCV*, 2750–2758.

Oquab, M.; Bottou, L.; Laptev, I.; and Sivic, J. 2014. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, 1717–1724.

Sagonas, C.; Tzimiropoulos, G.; Zafeiriou, S.; and Pantic, M. 2013. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *ICCVW*, 397–403.

Shin, S.; Hwang, K.; and Sung, W. 2016. Generative knowledge transfer for neural language models. *arXiv preprint arXiv:1608.04077*.

Shu, Z.; Sahasrabudhe, M.; Guler, R. A.; Samaras, D.; Paragios, N.; and Kokkinos, I. 2018. Deforming autoencoders: Unsupervised disentangling of shape and appearance. In *ECCV*, 650–665.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2256–2265. PMLR.

Thewlis, J.; Bilen, H.; and Vedaldi, A. 2017. Unsupervised learning of object landmarks by factorized spatial embeddings. In *ICCV*, 5916–5925.

Vahdat, A.; and Kautz, J. 2020. NVAE: A deep hierarchical variational autoencoder. *NeurIPS*, 33: 19667–19679.

Varol, G.; Romero, J.; Martin, X.; Mahmood, N.; Black, M. J.; Laptev, I.; and Schmid, C. 2017. Learning from Synthetic Humans. In *CVPR*.

Wang, D.; and Zhang, S. 2022. Contextual Instance Decoupling for Robust Multi-Person Pose Estimation. In *CVPR*, 11060–11068.

Wang, Y.; Wu, C.; Herranz, L.; van de Weijer, J.; Gonzalez-Garcia, A.; and Raducanu, B. 2018. Transferring gans: generating images from limited data. In *ECCV*, 218–234.

Wang, Z.; Weng, J.; Yuan, C.; and Wang, J. 2023. Truncate-split-contrast: a framework for learning from mislabeled videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 2751–2758.

Wu, W.; Qian, C.; Yang, S.; Wang, Q.; Cai, Y.; and Zhou, Q. 2018. Look at Boundary: A Boundary-Aware Face Alignment Algorithm. In *CVPR*.

Yin, F.; Zhang, Y.; Cun, X.; Cao, M.; Fan, Y.; Wang, X.; Bai, Q.; Wu, B.; Wang, J.; and Yang, Y. 2022. Styleheat: One-shot high-resolution editable talking face generation via pretrained stylegan. *arXiv preprint arXiv:2203.04036*.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2017. mixup: Beyond empirical risk minimization. *ICLR*.

Zhang, L.; and Agrawala, M. 2023. Adding Conditional Control to Text-to-Image Diffusion Models. arXiv:2302.05543.

Zhang, Y.; Guo, Y.; Jin, Y.; Luo, Y.; He, Z.; and Lee, H. 2018. Unsupervised discovery of object landmarks as structural representations. In *CVPR*, 2694–2703.

Zhu, J.-Y.; Krähenbühl, P.; Shechtman, E.; and Efros, A. A. 2016. Generative visual manipulation on the natural image manifold. In *ECCV*, 597–613. Springer.

Zhu, J.-Y.; Park, T.; Isola, P.; and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *CVPR*, 2223–2232.

6 Training Details

Pre-training Setting of StyleGAN. We follow the settings in StyleGAN (Karras, Laine, and Aila 2019) to learn the source generator. All the hyper-parameters are kept the same, and the generator is trained until convergence. The training set is re-rendered from the pose annotation of the Source Dataset S by the function $\alpha(\cdot)$. Different methods share the common pre-training weights for a fair comparison.

Training Setting of FlexPose. We freeze the weights of the pre-trained source generator and only finetune on the linear block-diagonal matrix $\tau(\cdot)$. $\tau(\cdot)$ is initialized with an identity matrix $\boldsymbol{I}$, and only a few selected blocks (3rd, for example) of it can be adjusted due to the sparse regularization. The training set only includes the skeletons re-rendered from the target pose set T and the extended annotation set T^*. The parameter betas of the Adam optimizer is set as $(0.9, 0.999)$. The learning rate is set to 0.1. The batch size is 128 with a training length of 1000 iterations. It takes less than 5 minutes on a single V100 GPU to finetune the model.

Training Setting of Methods for Comparison. We compare our FlexPose with three similar methods, AdaGAN (Noguchi and Harada 2019), FreezeD (Mo, Cho, and Shin 2020) and LoRA(Hu et al. 2021), in our experiments. We re-implement AdaGAN under our settings according to the official open-source code. FreezeD has an official implementation on StyleGAN and is utilized for the comparison. Similar to our method, the low-rank modules are conducted aside the linear-block-diagonal matrix $\tau(\cdot)$ with the default rank setting (r=8). To have a fair comparison, all the hyper-parameters and the settings including training length, learning rate, batch size, optimizer are well aligned with the competitors.

7 Implementation of $\alpha(\cdot)$ and $\beta(\cdot)$

$\alpha(\cdot)$ is a rule-based function. Given an annotation $\boldsymbol{y}$, we draw each keypoint on an empty black figure, and then connect them with fixed color lines. The choice of color is random and is pre-defined before the experiment. All methods share the same choice of colors. The visual effect is similar to a stick man.

As a reverse function of $\alpha(\cdot)$, $\beta(\cdot)$ is a pre-trained neural network. The input is a skeleton image $\boldsymbol{x}$ and the output is M heatmaps. The locations of keypoints are further obtained by the method introduced in (Jakab et al. 2018) by converting each heatmap into a 2D probability distribution. The training of $\beta(\cdot)$ is offline. The training procedure is achieved by minimizing the reconstruction loss on the given annotation $\boldsymbol{y}$ from the source dataset,

$$\mathcal{L}_{rec} = \| \boldsymbol{y} - \beta(\alpha(\boldsymbol{y})) \|_2^2. \tag{9}$$

8 Skeleton-guided Applications

Recently, there have been studies on generating human images with given 2D poses. A large amount of reasonable human poses in a certain style or distribution may be needed to evaluate their performance. FlexPose is born for this task and can generate infinite suitable 2D human poses with few-shot human poses in the needed style. Figure 8 gives some examples. For the appearance-based method, we used the CycleGAN (Zhu et al. 2017) as the generator. For the prompt-based method, we show the result on Stable Diffusion with ControlNet (Zhang and Agrawala 2023). As can be seen, the 2D human poses outputted from FlexPose diversify the human posture in the generated images.

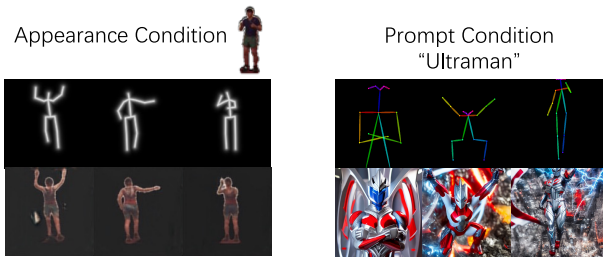

Figure 8: Application on pose-conditional human image generation. FlexPose can synthesize new poses in a certain style, which can be used as conditions for image generation.

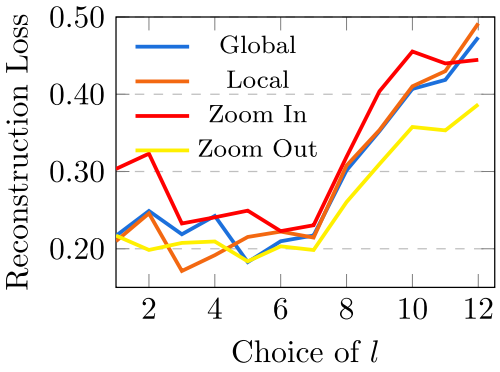

Figure 9: Reconstruction loss with different augmentation methods.

9 Extended Discussion

We expanded our discussion on the fine-tuned layers selection presented in Figure 4 of the main text. The discussion investigates the pose geometric semantics of each layer in the StyleGAN (Karras, Laine, and Aila 2019).

Geometric Rotation Transformation.. Two types of geometric rotation transformation were applied to the poses from the target dataset. The global transformation, $\text{Aug}_G^{\theta}(\cdot)$, rotates the target skeleton image $\boldsymbol{x}$ by a given angle θ. In contrast, the local transformation, $\text{Aug}_L^{\gamma}(\cdot)$, rotates a single leg of the target skeleton image $\boldsymbol{x}$ by an angle γ. The angle θ is randomly selected from $[-45°, 45°]$, while γ is randomly chosen from $[135°, 225°]$. An illustration of these two transformations is provided in Figure 10. The augmented skeleton images can differ significantly from the samples in the source distribution. Our investigation centers on identifying which layer calibration can minimize the reconstruction loss during the transformation phase when the target skeleton images are individually transformed by these two methods.

Scale Transformation.. We also conducted experiments on the scale transformation. We scale the skeleton by η, where η is randomly chosen from $[0.7, 0.9]$ and $[1.1, 1.2]$. We also showed some examples in Figure 10 and report the reconstruction loss.

The quantitative results are presented in Figure 9. The findings indicate that coarser layers (layers 3 and 4) result in lower reconstruction loss with global augmentation, whereas finer layers (layers 5 and 6) are more suitable for local transformations. A noteworthy conclusion is that layers with $l \geq 8$ and $l \leq 2$ have a lesser effect on fitting the distribution of skeleton images. A potential explanation is that the coarsest layers ($l \leq 2$) primarily determine the background, while the finest layers are less associated with skeleton action. This analysis underscores a hierarchical framework in the latent space of skeleton images, extending related insights observed in natural images.

Figure 10: Illustration of augmentations. All of them generate skeleton images that less frequently appear in source targets.