FLEXPOSE: POSE DISTRIBUTION ADAPTATION WITH FEW-SHOT GUIDANCE

Anonymous authors
Paper under double-blind review

ABSTRACT

Annotating human pose images can be costly. Meanwhile, there is an unavoidable performance drop when a pre-trained pose estimation model is directly evaluated on a new dataset. We observe that pose distributions from different datasets share similar pose hinge-structure priors with different geometric transformations, which inspires us to learn a pose generator that can be flexibly adapted to generate the pose of a new pose distribution with prior and transformation disentangling. In this paper, we treat human poses as skeleton images and propose a scheme to transfer a pre-trained pose annotation generator with only a few annotation guidances. By finetuning a limited number of linear layers, the transferred generator is able to generate any number of pose annotations that are similar to the target pose distribution. We evaluate the proposed FlexPose on several cross-dataset settings qualitatively and quantitatively. FlexPose surprisingly achieves around 41.8% average performance improvement on the Unsupervised Pose Estimation task when it transfers the pose distribution of COCO, 3DHP and Surreal dataset to that of the H36M dataset.

1 INTRODUCTION

In this decades, Deep Learning achieves great success on various computer vision tasks. Deep generative models such as Generative Adversarial Networks (GAN) (Goodfellow et al., 2014), Variational Anto-Encoder (VAE) (Kingma & Welling, 2014; Vahdat & Kautz, 2020) and Diffusion Models (Ho et al., 2020) have been well developed. With the complex structure and high model capacity, modern generative methods (Karras et al., 2019; Brock et al., 2018) can estimate complicated data distribution and generate high-quality images.

As data-hunger methods, deep neural networks require large-scale datasets with high-quality human annotations in training. However, annotating on large-scale datasets is costly and time-consuming, especially on regression-based tasks such as pose estimation. To bypass labor-intense annotating, machine learning techniques *e.g.,* unsupervised learning (Chen et al., 2020), semi-supervised learning (Berthelot et al., 2019), and meta-learning (Finn et al., 2017) have been heatedly discussed in the community. Among these methods, Domain Adaptation (DA) introduces additional information from existing annotated datasets to a target dataset and is verified effective on several computer vision tasks. Unfortunately, things become different in the human-related dataset, *e.g.*, human pose (Ionescu et al., 2014), human face (Wu et al., 2018). As the source human appearance is required in DA-based methods, they may import unexpected data distribution bias Buolamwini & Gebru (2018), *e.g.*, gender or color, from the source. Besides, the direct exposure of private portraits may raise the privacy issue.

On the other side, Pose Distribution Adaptation (PDA), which transfers human pose keypoints only, can be a choice in the scenario of human datasets when transferring knowledge from one domain to another. It can avoid directly utilizing the human appearance images and thus tackle the above issues. A common observation is that different human poses share a similar hinge-structure prior. Typically, poses in a target dataset can be transferred from poses in a pre-collected source set by applying geometric transformations. Therefore, pose adaptation can be much easier and safer than DA, intuitively. Inspired by this observation, we propose FlexPose to transfer source pose distribution to target and generate a new pose distribution with only few-shot guidances in this paper.

The proposed FlexPose aims to estimate the pose distribution on the target dataset given few-shot annotations. We believe that the pose prior can be learned from the source dataset and be utilized with transformations to approximate the target pose distribution. In FlexPose, we treat pose annotations as *skeleton images* to well align the annotations with their RGB appearance correspondences, and to improve the model's ability on prior learning. We first learn the pose prior and fit the empirical distribution from a source human pose dataset by a multi-layer generative model. Thereafter, we calibrate specific layers by inserting learnable lightweight transformation modules to transfer the distribution to the target domain. Considering that only few-shot poses are given, we introduce three regularizations to avoid a collapse transfer solution. By generating credible pose interpolation by Pose-mixup and by strictly limiting the complexity of the transfer module, we minimize the data requirement of FlexPose in pose adaptation. The advantages of FlexPose are obvious. First, FlexPose is computation-efficient. It operates on the pose domain, and hence the convergence of the training procedure is much faster than on both the pose and image domain together. Second, FlexPose is data-efficient. We only need few-shot poses from the target dataset to finetune the transfer modules. At last, FlexPose performs well. Extensive experiments on three pose-based tasks, *i.e.,* human pose estimation, human face landmarks detection, and pose-conditional image generation, show that FlexPose outperforms baselines by a large margin both quantitatively and qualitatively.

Our contributions can be summarized as follows:

1. We propose to treat the transfer task of human pose distribution as transfer of skeleton image generator and demonstrate that a target pose distribution can be well approximated from a well-learned pose prior.
2. We propose FlexPose, a novel few-shot pose adaptation framework. With three well-designed regularizations, FlexPose can efficiently transfer a pose distribution to a target one by referring to few-shot guidance with low computation and storage costs.
3. Extensive experiment results on three pose-related tasks show that training with poses generated by FlexPose achieves remarkable improvement over baseline methods.

2 Related Works

Deep Generative Model for Image Generation. Deep generative models such as GAN, Variational AutoEncoder (VAE) and Diffusion model achieve great success in realistic/artificial image generating and natural image distribution modeling. Recently proposed generative models such as StyleGAN (Karras et al., 2019), DDPM (Ho et al., 2020), NVAE (Vahdat & Kautz, 2020) introduce new mechanisms, new architecture and new regularizations into image generation. VAEs (Kingma & Welling, 2014) learn to maximize the variational lower bound of likelihood. Diffusion probabilistic models (Sohl-Dickstein et al., 2015) treat images synthesis by a denoising procedure. GANs (Goodfellow et al., 2014) train two networks in an adversarial manner to learn how to generate realistic images. Among them, Karras *et al.* (Karras et al., 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 & 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 as pre-training techquni. 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 & Harada (2019) transfers knowledge from a large dataset to a small dataset by re-computing batch statistics. Existing methods focus on either image domain or neural language processing domain (Shin et al., 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.

Human Pose Estimation. Human pose estimation is a task that predicts the 2D human pose from a single image. Fully-supervised methods (Andriluka et al., 2009; Bai & Wang, 2019; Be-

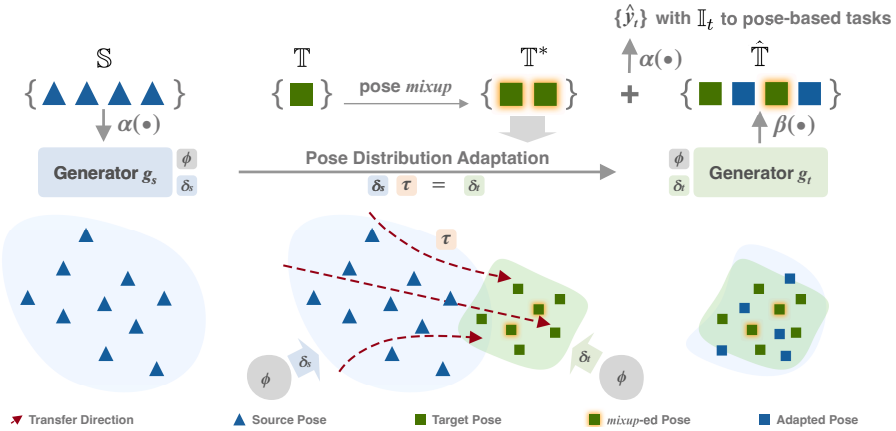

Figure 1: An Illustration of the FlexPose Framework for Pose Distribution Adaptation.

lagiannis & Zisserman, 2017) utilize large 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 et al., 2019) 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 human pose annotation. These methods can train a model with prior knowledge learned from weak unpaired annotations or can utilize unpaired image and annotation sets to generate meaningful landmarks. We choose KeypointGAN (Jakab et al., 2020) as the baseline for comparison in this paper. KeypointGAN is trained in an unpaired way, which only requires the pose annotations and the images lie in the same distribution, but not one-to-one aligned. However, when the distribution of pose and images are not aligned, the estimation performance drops a lot (Jakab et al., 2020). We use this characteristic as an indicator to show how close the pose distribution transferred by FlexPose is to the target distribution.

3 METHOD

Given a newly collected set of human pose images $\mathbb{I}_t = \{\boldsymbol{x}_n\}_{n=1}^N$ and its 2D pose annotations $\mathbb{T} = \{\boldsymbol{t}_m\}_{m=1}^M, \boldsymbol{t} \in \mathbf{R}^{J\times 2}$ on few-shot level, *i.e.* $M \ll N$, our goal is to estimate the distribution $\mathcal{T}$ of the corresponding pose annotation of $\mathbb{I}_T$ by only referring a few pose annotations $\mathbb{T}$. J here is the number of joints in each pose. This task setting is challenging and seems impracticable at the first glance. However, we believe that with the prior from off-the-shelf annotations $\mathbb{S} = \{\boldsymbol{s}_k\}_{k=1}^K$ ($M \ll K$, $\boldsymbol{s} \in \mathbf{R}^{J\times 2}$), the distribution $\mathcal{T}$ can be estimated and well shaped. We transfer the source distribution $\mathcal{S}$ estimated from $\mathbb{S}$ to the target domain to estimate the target distribution $\mathcal{T}$, with the guidance of $\mathbb{T}$. To evaluate the performance of the distribution estimation, we randomly generate pose annotations from the predicted distribution $\hat{\mathcal{T}}$, and utilize the generated annotations to train a neural network for pose estimation from single image $\boldsymbol{x}$.

3.1 OVERVIEW

As illustrated in Figure 1, our framework consists of three steps: (1) **Generic Pose Distribution Estimation**. We learn a generator $g_s(\cdot)$ on the pose annotation set $\mathbb{S}$ to estimate the pose distribution $\hat{\mathcal{S}}$ of source domain. The generator takes a latent code $\boldsymbol{z}$ as input and outputs a pose annotation $\hat{\boldsymbol{s}}$, *i.e.* $\hat{\boldsymbol{s}} = g_s(\boldsymbol{z})$. We take the distribution of generated $\hat{\boldsymbol{s}}$ to mimic that of the generic pose $\boldsymbol{s}$. (2) **Few-shot Distribution Transformation**. Given few-shot target annotation set $\mathbb{T}$, we transfer $g_s(\cdot)$ to fit the pose distribution of the set $\mathbb{I}_T$, and learn a new generator $g_t(\cdot)$ of the target domain. Considering the limited knowledge can be acquired from few-shot target pose annotation $\mathbb{T}$, we introduce three regularizations to avoid reaching collapse solution. (3) **Target Pose Sampling**. The transferred generator $g_t(\cdot)$ can flexibly generate any number of fake pose annotations by randomly sampling in the latent space. This generated annotation set $\hat{\mathbb{T}}$ will be treated as an extension of given annotations set $\mathbb{T}$, and guide the training of a keypoints detector on target domain images $\mathbb{I}_T$ in an unpaired way.

3.2 GENERIC POSE DISTRIBUTION ESTIMATION

Deep generative models have been widely verified that they have rich capacity to well approximate image distributions when given sufficient training data. Motivated by the success of these generative models Karras et al. (2019) on natural/artificial image generation, we treat 2D pose annotations $\boldsymbol{s}, \boldsymbol{t} \in \mathbf{R}^{J\times 2}$ as skeleton images $\boldsymbol{y}_s, \boldsymbol{y}_t \in \mathbf{R}^{C\times W\times H}$ and extend an image generator to synthesize skeleton images. The transformation between the 2D key points and the skeleton images can be implemented by functions $\alpha(\cdot)$ and $\beta(\cdot)$, namely $\boldsymbol{y}_s = \alpha(\boldsymbol{s})$ and $\boldsymbol{s} = \beta(\boldsymbol{y}_s)$. To achieve precise semantic alignment with the appearance correspondence, each bone in the skeleton image is assigned with an unique color. Therefore, C of each skeleton image is set as three (RGB channels).

A general generator can be formulated as a mapping function $g(\cdot)$ which maps a latent code $\boldsymbol{z}$ to a skeleton images $\boldsymbol{y}$. The probability distribution of skeleton images hence is estimated by $p(\boldsymbol{y}) = p(\boldsymbol{z})p_g(\boldsymbol{y}|\boldsymbol{z})$. We assume that the pose distributions of different datasets share similar pose prior. Therefore their distributions can transfer to one another by geometric transformations. On the basis of this assumption, we further factorize the generator $g(\cdot)$ as $g = \delta \circ \phi$. Therefore, skeleton image generator of source domain can be formulated as

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

in which $\phi(\cdot)$ preserves the learned pose prior and $\delta(\cdot)$ records the transformation to a certain pose domain. $\boldsymbol{h}$ is the learned prior and $\delta(\cdot)$ maps $\boldsymbol{h}$ to the skeleton image $\boldsymbol{y}_s$ in source domain. Similarly, the distribution of target domain can be formulated as $p(\hat{\boldsymbol{y}}_t) = p(\boldsymbol{z})\; p_\phi(\boldsymbol{h}|\boldsymbol{z})\; p_\delta^s(\hat{\boldsymbol{y}}_t|\boldsymbol{h})$. With the prior sharing assumption, the pose distribution adaptation aims at transferring pre-trained conditional probability $p_\delta^s(\hat{\boldsymbol{y}}_s|\boldsymbol{h})$ to $p_\delta^t(\hat{\boldsymbol{y}}_t|\boldsymbol{h})$ with few-shot guidance $\mathbb{T}$:

$$p_\delta^s(\hat{\boldsymbol{y}}_s|\boldsymbol{h}) \xrightarrow{\mathbb{T}} p_\delta^t(\hat{\boldsymbol{y}}_t|\boldsymbol{h}). \tag{2}$$

Considering the ability of StyleGAN in seperating high-level attributes and the quality of attribute interpolation, we utilize the network architecture of StyleGAN to disentangle the pose prior and the transformation of the source domain generator,

$$g_s = \phi \circ \delta_s = \phi \circ (A \circ f)_s, \tag{3}$$

where f 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 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 target domain by adjusting the style code.

3.3 FEW-SHOT DISTRIBUTION TRANSFORMATION

To transfer $p_\delta^s(\hat{\boldsymbol{y}}_s|\boldsymbol{h})$ to $p_\delta^t(\hat{\boldsymbol{y}}_t|\boldsymbol{h})$, we adjusting 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

$$g_t = \phi \circ \delta_t = \phi \circ (\tau \circ \delta_s). \tag{4}$$

To learn the transfer function τ, we first randomly sample M latent code $\mathbb{Z}$ from the latent space,and map these codes to skeleton images to be close to the guidances. This transferring procedure can be achieved by minimizing the follow perceptual loss

$$\min_{\theta_\tau} \; \mathcal{L}_{s\to t} = \min_{\theta_\tau} \sum_{m=1}^{M} \left\| \Gamma\Big(g_t(\boldsymbol{z}_m); \theta_\tau\Big) - \Gamma\Big(\alpha(\boldsymbol{t}_m)\Big) \right\|_2^2, \tag{5}$$

where θ_τ is the parameter of $\tau(\cdot)$, Γ is an image feature extractor (VGG-net in our experiment), $\boldsymbol{z}$ is from the set of $\mathbb{Z}$, and $\boldsymbol{t}$ is from the few-shot 2D pose annotation set $\mathbb{T}$.

The problem however is, we only have few-shot guidance $\mathbb{T}$ 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, recent

GAN inversion technique shows that the layer-wise style code in StyleGAN leads to the hierarchical disentanglement of local and global attributes, which well aligns with our motivation of adapting pose distribution by considering the global geometric transformation between poses. We thus adjusting 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,

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

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, where only the l-th block is not an identity matrix $\mathbf{I}$. We experimented with changing different layers and found that setting $l = 3$, *i.e.,* transferring the third coarsest layer, forces the transformation to mainly occur on the pose action level. We have also tried the combination of different layers. But the best of them gets a similar result with the case when $l = 3$.

Pose-mixup Regularization. Intuitively, most of pose interpolated between two real poses physically exists, 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. *Pose-mixup* regularizes the neural network to learn the simple linear behavior in-between 2D poses, and thus avoids the model to generate unrealistic human pose annotations. By mixing up the corresponding joints of any two 2D pose with mixup ratio λ, the extended guidance set $\mathbb{T}^*$ of size E is then defined as,

$$\mathbb{T}^* = \{\boldsymbol{t}^* \mid \boldsymbol{t}^* = \lambda\, \boldsymbol{t}_i + (1-\lambda)\, \boldsymbol{t}_j,\ \lambda \in [0,1],\ \boldsymbol{t}_i, \boldsymbol{t}_j \in \mathbb{T}\}\,. \tag{7}$$

3.4 Target Pose Sampling

Once the transferred generator g_t is obtained, we can generate theoretically as many target skeleton images as possible by randomly sampling latent codes in the estimated target distribution $\hat{\mathcal{T}}$. Unfortunately, the generated target skeleton images is not perfect and may bring in artifacts which mislead the training of a neural network. To address this issue, we introduce $\alpha(\cdot)$ and $\beta(\cdot)$, and design a information bottleneck to filter out the random noise.

Once a fake skeleton image $\tilde{\boldsymbol{y}}_t$ is generated from $\hat{\mathcal{T}}$, we extract the coordinates of interpretable 2D keypoints $\hat{\boldsymbol{t}}$ from it, *e.g.*, nose, hands *et al.* by applying $\hat{\boldsymbol{t}} = \beta(\tilde{\boldsymbol{y}}_t)$. $\beta(\cdot)$ is a neural network regressor pre-trained on $\mathcal{S}$. Then, a rule-based render $\alpha(\cdot)$ simply draws keypoints from $\hat{\boldsymbol{t}}$ and connects them with straight lines on a blank figure, *i.e.* $\hat{\boldsymbol{y}}_t = \alpha(\hat{\boldsymbol{t}})$. The visual effect is similar to the stick man. This re-render process is given by

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

where the keypoint coordinates $\hat{\boldsymbol{t}}$ act as a tight information bottleneck that preserves skeleton information and ignore random noise.

With the synthsized skeleton images $\{\hat{\boldsymbol{y}}_t\} = \alpha(\hat{\mathbb{T}})$ and the extended pose-mixuped images $\alpha(\mathbb{T}^*)$ as unpaired supervision, we train a KeypointGAN (Jakab et al., 2020) as our keypoint detector for unsupervised pose/landmark estimation and pose-conditioned image generation.

4 Experiments

Our method is evaluated on three tasks: human pose estimation, human face landmark detection and pose-conditional image generation. The training and testing setting. We train a StyleGAN (Karras et al., 2019) by using the annotations from the source datasets, *e.g., COCO-2017* (Lin et al., 2014), *300-VW* (Sagonas et al., 2013), to estimate the distribution of source human pose/face, respectively. Meanwhile, several target datasets, *e.g., Human3.6M* (Ionescu et al., 2014) and *WFLW* (Wu et al., 2018) are considered. The training and testing settings are kept the same in all experiments for fair comparison. At last, we also show FlexPose's potential application on pose-conditioned image generation.

4.1 HUMAN POSE

Source Datasets. *COCO-2017* (Lin et al., 2014) is a large-scale dataset including multiple tasks. We only keep the annotated people instances with full pose annotations to construct a training set of 32 thousand samples. *MPI-INF-3DHP (3DHP)* (Mehta et al., 2017) contains 2D and 3D human pose annotations from eight subjects and covers eight complex exercise activities. Its training set consists of more than 1.8 million frames. *SURREAL* (Varol et al., 2017) is a synthetic dataset containing more than six million frames of people in motion.

Target Datasets. In the large-scale dataset *Human3.6M* (Ionescu et al., 2014), there are 3.6 million accurate 2D and 3D human pose frames captured from 11 actors in 17 scenarios. The background is static. As previous works (Jakab et al., 2020; Gholami et al., 2022) do, we use the frist five subjects (S1, S5, S6, S7, S8) for training and the remaining subjects (S9, S11) for evaluation. The *Simplified Human3.6M* dataset collected in Zhang et al. (2018) is a subset of Human3.6M, containing six activities where human bodies are mostly upright. The dataset consists of 800 thousand training and around 90k testing images.

In our experiments, the RGB images in the source datasets are not considered in the pose adaptation task. We only keep their keypoint annotations and crop them into a compact rectangle skeleton image according to the provided annotations. To align different pose annotation format across different datasets, we use 15 shared keypoints across different datasets in this paper (three for each limb in addition to the head, neck, and pelvis). Thanks to the data efficiency of our method, the disk storage is reduced to one of the thousands of its original size (*e.g.,* $\mathbf{1/1200}\times$ in 3DHP). During training, we use the RGB images and only keep few-shot annotations of the target dataset in the training set as a clue for pose adaption. Evaluation is conducted on the test set with paired pose annotations.

Evaluation Metrics. We report 2D landmark detection performance in each experiment for evaluation. Two standard evaluation matrices is considered to compare our method with baselines. The MSE reports a mean square error in pixels over the 15 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.

Experiment Settings. Our method focuses on pose adaptation and we transfer the pose distribution from different source datasets to each target dataset to align with the distribution of the target human pose image. As mentioned in Section 3.4, we feed the generated skeleton images $\hat{\boldsymbol{y}}_t$ and RGB human pose images $\boldsymbol{x}$ from the target dataset into an off-the-shelf pose estimation algorithm KeypointGAN (Jakab et al., 2020) to evaluate the effectiveness of FlexPose. As a baseline, we train the keypoint detector, KeypointGAN, on each target dataset by directly using the pose annotations $\mathbb{S} = \{\boldsymbol{s}_k\}_{k=1}^{K}$ from the source dataset, which we denotes as ***Direct*** in the comparison. Given that our proposed FlexPose utilizes both the source annotation $\mathbb{S}$ and the mixup-ed few-shot target annotation $\mathbb{T}^{*}* = \{\boldsymbol{t}_e^*\}_{e=1}^{E}$, in addition to $\mathbb{S}$, we also involve the pose annotations $\mathbb{T}^*$ from the target dataset in the pose annotations of baseline for fair comparison, unless especially marked. Besides, we also introduce the version of KeypointGAN trained when the distribution of pose annotations is well aligned with human pose images, and we denote this version as ***Aligned***.

In our experiments, we use 30 annotations ($M = 30$) from target dataset when the target dataset is Human3.6M (two for each class) and 12 annotations ($M = 12$) when the target dataset is Simplified Human3.6M. The number of interpolated pose (pose-mixup) E is set as 1000 for all experiments. With the spirit of fair comparison, in the settings of both baseline and FlexPose, the size of annotation set is set as 33 thousand (32 thousand to line up with the source COCO and one thousand to line up with the target mixup) when the source dataset is COCO, and the size is set as 121 thousand when the source dataset is 3DHP or SURREAL. By this way, the impact of the scale of annotation set is excluded, even though FlexPose can theoretically generate almost infinite annotations with little cost. All other hyper-parameters are kept the same as the original work of KeypointGAN Jakab et al. (2020).

Performance Comparisons. Quantitatively, we compare the performance of baselines with FlexPose on human pose estimation in Table 1. We highlight the ideal result on KeypointGAN by green when the source dataset is the same as the target dataset. The performance gap between the method and the ideal case can be treated as a referenced distance between the current pose distribution and target pose distribution. As shown in Table 1, the keypoint detector has much lower performance on the target dataset when the pose annotations are from different datasets, especially when some

Table 1: Results on Human Pose Estimation Task. S-H36 and H36 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. K stands for *kilo* and M stands for *million*.

Method	Target	Source	Annotation Set Size	MSE-Pixel (↓)	PCK (↑)
Aligned	Human3.6M	H3.6M	3.6M	12.07	0.443
Direct		COCO	33K	17.86	0.015
Ours		COCO	33K	**13.19**	**0.585**
Aligned	Simplified Human3.6M	S-H3.6M	800K	2.96	0.843
Direct		COCO	33K	5.47	0.685
Direct		3DHP	121K	12.66	0.000
Direct		SURREAL	121K	11.18	0.000
Ours		COCO	33K	**4.61**	**0.706**
Ours		3DHP	121K	**5.98**	**0.467**
Ours		SURREAL	121K	**6.47**	**0.499**

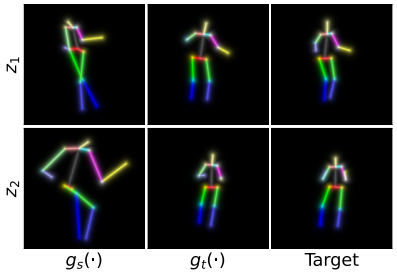

Figure 2: Visualization of Pose Adaptation. The left column is sampled from the generic distribution (generated from $g_s(\cdot)$). The middle column is sampled from the adapted distribution (generated from $g_t(\cdot)$). In right column, the two poses are from few-shot guidance of the target dataset, corresponding to the pose in the middle column. Each row of left and middle columns are generated from the same random noise.

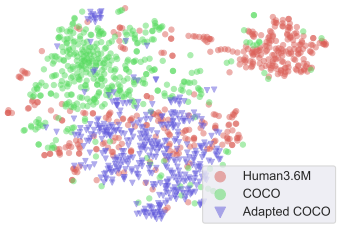

Figure 3: t-SNE Visualization of Human Poses before and after Adaptation. We visualize the pose distribution ($\mathbf{R}^{J\times 2}$) in a two-dimensional space.

of them have a distinct pose distributions 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 much 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 poses with similar distribution to the target dataset, even when less to only two poses per class in the target dataset are given.

Visualization. Qualitatively, we show the visual result of pose transformation in Figure 2. For each row, we show one skeleton (Left) that was randomly sampled from generic pose distribution, one skeleton (Middle) that was sampled from transferred distribution by using the same latent noise with Left, and one skeleton (Right) in the few-shot samples set $\mathcal{T}$ from target dataset. For each dataset, we can see that the skeletons Left and the skeleton Middle generated from the same randomly sampled high-dimensional noise are visually quite different, and the transferred one (Middle) is more similar to the skeleton (Right) in the target dataset.

In Figure 3, we plot the t-SNE embedding of the poses generated by FlexPose (Adapted COCO), comparing it with the embedding of poses from the source dataset (COCO) and target dataset (Human3.6M). As can be seen, the embedding of poses from the source dataset and target dataset are separated, and the distribution of generated poses significantly overlaps with the target ones. Since only two shots are utilized as guidance for each class from the target dataset, some poses distribution are 'missed' (UPPER RIGHT).

Ablation Study & Parameter Sensitivity Analysis. We conduct ablation studies to evaluate the impact of:

a) Annotation Set Scale. We study the effect of generated annotation set scale under the human pose detection setting (COCO→Simplified Human3.6M). We enlarge the generated annotation set to 121

Table 2: Ablation study on human pose detection. The target dataset is Simplified-Human3.6M for all experiments. #0 indicates the result of method **Aligned** as a reference. Others are the results of **FlexPose**. CO, 3D and SU are short for COCO, 3DHP and SURREAL. K stands for *kilo* and M stands for *million*.

#	Source Dataset	Annotation Set Size	Channel	Shots	l	MSE-Pixel	PCK10
0	N/A	800K	1	N/A	N/A	2.96	0.84
1	CO	33K	3	12	3	4.61	0.71
2	CO	61K	3	12	3	4.96	0.69
3	CO	121K	3	12	3	3.79	0.77
4	CO	121K	3	24	3	3.80	0.75
5	CO	121K	3	48	3	3.73	0.70
6	CO	121K	3	12	1,3	3.82	0.78
7	CO	121K	3	12	3,5	4.02	0.61
8	CO	121K	3	12	all	4.50	0.66
9	3D	121K	3	12	3	5.98	0.47
10	SU	121K	3	12	3	6.47	0.50
11	CO+3D	121K	3	12	3	5.28	0.59
12	CO+3D+SU	121K	3	12	3	5.19	0.59
13	CO	121K	1	12	3	4.15	0.79

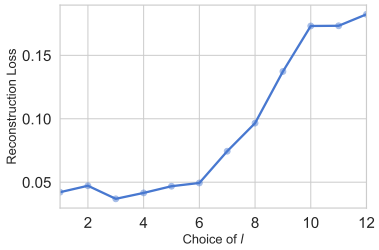

Figure 4: Ablation Study on the Number of Finetuned Layer l.

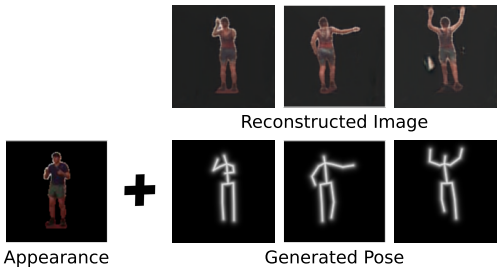

Figure 5: Application on Pose-conditioned Human Image Generation. The transferred generator can generate new poses as conditions for image generation.

thousand in our method and evaluate the result as shown in Table 2 (#1 to #3). The MSE-Pixel drops by 0.8 (4.6→3.8) and PCK10 raise by 0.06 (0.71→0.77). The experiment result shows that FlexPose can bring further potential improvement to pose-related tasks by generating a large number of fake annotations with minimal cost.

b) Number of Shots. We study the effect of the number of poses from the target dataset. Under the setting of Simplified Human3.6M → COCO, we increase the number of shots from 12 (2 per class) to 48 (8 per class) and found that the performance of the pose detector has no obvious difference. The results can be found in Table 2 (#3 to #5). There is an explanation 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. Only 12 samples are enough for target distribution localization.

c) Number of Finetuned Layers. In previous experiments, we empirically choose $l = 3$ in Equation (4) for all experiment and get significant improvement. We found that the choice of l is not strictly fixed. We record the reconstruction loss in Equation (5) on Figure 4 when choosing different l. We get a similar reconstruction loss in the earlier layer ($l \leq 6$), and the loss rises when we choose a later layer ($l > 6$), which may indicate the decomposition property of layers in the generative model. We have also tried a composition of multi-layer, and the results can be found in Table 2 (#3, #6, #7 and #8). The result in #8 shows the necessity of sparse regularization. We leave the best choice of l to future work.

d) Multi-source Datasets. To study the effect of the setting where the annotations in the source set are from different datasets, we conduct two additional experiments (#11 and #12) in Table 2 and compare them with existing experiments (#3, #9 and #10). In #11 and #12, we use the union of different source datasets to train the generic generator. The result indicates that the mixture of source datasets brings worse result compared with the result in COCO but better result compared with the results in 3DHP and SURREAL. The result keeps the same with our previous observation: FlexPose performs better when the gap between source distribution and target distribution is closer.

Table 3: Results on Human Face Detection Task.

Method	Target	Source	Annotation Set Size	MSE-Pixel (↓)	PCK20 (↑)
Aligned	WFLW	WFLW	7.5K	8.25	0.425
Direct		300-VW	96K	18.78	0.101
Ours		300-VW	96K	**11.64**	**0.482**

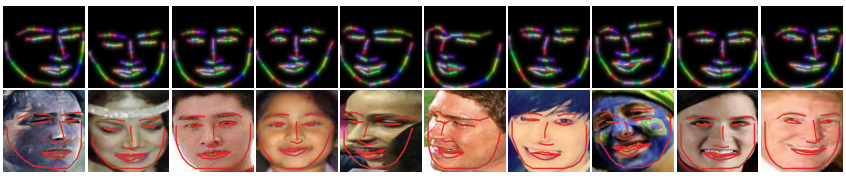

Figure 6: Visualization of Human Face Landmark Detection on WFLW dataset. Upper Row: Detected Landmarks Unpairedly Learned from Transferred Poses. Bottom Row: Detected Landmarks in the upper row with their corresponding Human Face.

e) Color of skeleton. We use colorful skeletons in our FlexPose to embed different human parts. And we compare it (#3) with the result of the black-and-white version (#13). The result can be found in Table 2. The results show that the colorful encoding brings marginal improvement to FlexPose in MSE but slightly harms PCK compared to the black-and-white skeleton.

4.2 Human Face

It is straightforward to expand FlexPose to human face landmarks transformation, since both human pose and human face consist of a set of pre-determined keypoints. Therefore in this section, we extend our method to solve face landmark detection task.

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 face in the training set with 68 facial landmarks.

We treat 300-VW as the source dataset and only use its annotations for transferring. All the images and few-shot annotations in the target dataset WFLW are utilized. 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 number of few-shot guidance is set as 12 (M = 12). We report the evaluation results on the validation set of WFLW in Table 3 and Figure 6. FlexPose still outperforms baseline by a large margin (MSE $18.78 \rightarrow 11.64$ and PCK $0.101 \rightarrow 0.482$), which shows the outstanding task generality of FlexPose.

4.3 Application on Pose-conditioned Image Generation

Recently, there have been studies work on how to generate human images with given poses based on one or more reference images. A large amount of reasonable human pose in a certain style or distribution may be needed to evaluate their performance. FlexPose is born for this job and can generate infinite suitable human poses with few-shot human poses in the needed style. Figure 5 show some examples. The backbone we used is a pre-trained CycleGAN (Zhu et al., 2017), and the channel of the skeleton is set as one here.

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

Mykhaylo Andriluka, Stefan Roth, and Bernt Schiele. Pictorial structures revisited: People detection and articulated pose estimation. In *CVPR*, pp. 1014–1021. IEEE, 2009.

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

Vasileios Belagiannis and Andrew Zisserman. Recurrent human pose estimation. In *FG*, pp. 468–475. IEEE, 2017.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. *NeurIPS*, 32, 2019.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2018.

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

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, pp. 1126–1135. PMLR, 2017.

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pp. 1180–1189. PMLR, 2015.

Baris Gecer, Stylianos Ploumpis, Irene Kotsia, and Stefanos Zafeiriou. Ganfit: Generative adversarial network fitting for high fidelity 3d face reconstruction. In *CVPR*, pp. 1155–1164, 2019.

Zhenglin Geng, Chen Cao, and Sergey Tulyakov. 3d guided fine-grained face manipulation. In *CVPR*, pp. 9821–9830, 2019.

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 *CVPR*, pp. 13075–13085, 2022.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 33: 6840–6851, 2020.

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, jul 2014.

Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks through conditional image generation. *NeurIPS*, 31, 2018.

Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Self-supervised learning of interpretable keypoints from unlabelled videos. In *CVPR*, pp. 8787–8797, 2020.

Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, pp. 7122–7131, 2018.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pp. 4401–4410, 2019.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *ICLR*, 2014.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, pp. 740–755. Springer, 2014.

Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. Learning transferable features with deep adaptation networks. In *ICML*, pp. 97–105. PMLR, 2015.

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)*, pp. 506–516. IEEE, 2017.

Atsuhiro Noguchi and Tatsuya Harada. Image generation from small datasets via batch statistics adaptation. In *ICCV*, pp. 2750–2758, 2019.

Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *CVPR*, pp. 1717–1724, 2014.

Christos Sagonas, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: The first facial landmark localization challenge. In *ICCVW*, pp. 397–403, 2013.

Sungho Shin, Kyuyeon Hwang, and Wonyong Sung. Generative knowledge transfer for neural language models. *arXiv preprint arXiv:1608.04077*, 2016.

Zhixin Shu, Mihir Sahasrabudhe, Riza Alp Guler, Dimitris Samaras, Nikos Paragios, and Iasonas Kokkinos. Deforming autoencoders: Unsupervised disentangling of shape and appearance. In *ECCV*, pp. 650–665, 2018.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, pp. 2256–2265. PMLR, 2015.

Arash Vahdat and Jan Kautz. Nvae: A deep hierarchical variational autoencoder. *NeurIPS*, 33: 19667–19679, 2020.

Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, 2017.

Yaxing Wang, Chenshen Wu, Luis Herranz, Joost van de Weijer, Abel Gonzalez-Garcia, and Bogdan Raducanu. Transferring gans: generating images from limited data. In *ECCV*, pp. 218–234, 2018.

Wayne Wu, Chen Qian, Shuo Yang, Quan Wang, Yici Cai, and Qiang Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *CVPR*, 2018.

Fei Yin, Yong Zhang, Xiaodong Cun, Mingdeng Cao, Yanbo Fan, Xuan Wang, Qingyan Bai, Baoyuan Wu, Jue Wang, and Yujiu Yang. Styleheat: One-shot high-resolution editable talking face generation via pretrained stylegan. *arXiv preprint arXiv:2203.04036*, 2022.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *ICLR*, 2017.

Yuting Zhang, Yijie Guo, Yixin Jin, Yijun Luo, Zhiyuan He, and Honglak Lee. Unsupervised discovery of object landmarks as structural representations. In *CVPR*, pp. 2694–2703, 2018.

Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A Efros. Generative visual manipulation on the natural image manifold. In *ECCV*, pp. 597–613. Springer, 2016.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *CVPR*, pp. 2223–2232, 2017.