# 2.5D POSE GUIDED HUMAN IMAGE GENERATION

by

KANG YUAN

(Under the Direction of Professor Sheng Li)

Abstract

In this thesis, we propose the 2.5D pose guided human image generation method that integrates depth information with 2D poses. Basically, given a specific 2.5D pose and an image of a person, our model can generate a new image of that person with a target pose. In order to incorporate depth information into the pose structure, we present the Three-Layer Pose Space that allows more accurate pose transfer compared with regular 2D pose structure. Specifically, our pose space enables the generative model to solve the occlusion problems commonly happened in human image generation and helps us easily recognize spatial front-back relations of limbs. Our approach is trained end-to-end on images and the corresponding 3D coordinates. In qualitative experiments on the DeepFashion, Human 3.6M dataset, our model demonstrates significant improvement of visual effect regarding the depth of field.

Index words: Generative model, Autoencoder, Variational Autoencoder, U-Net, Pose estimation, Pose transfer, Computer vision

# 2.5D POSE GUIDED HUMAN IMAGE GENERATION

by

KANG YUAN

B.A., University of Nanjing Forestry, 2014

M.E., University of Southeast, 2017

A Dissertation Submitted to the Graduate Faculty

of The University of Georgia in Partial Fulfillment

of the

Requirements for the Degree

Master of Science

Athens, Georgia

2019

# 2.5D POSE GUIDED HUMAN IMAGE GENERATION

by

KANG YUAN

Approved:

| | |
|---|---|
| Major Professors: | Sheng Li |
| Committee: | Frederick Maier<br>Suchendra Bhandarkar |

Electronic Version Approved:

Suzanne Barbour
Dean of the Graduate School
The University of Georgia
May 2019

# 2.5D POSE GUIDED HUMAN IMAGE GENERATION

KANG YUAN

May 1, 2019

# Acknowledgments

I would like to thank Dr. Li and Dr. Maier for their immense help and patience in assisting me to complete this thesis.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Problem Definition

Generative models for human image generation can be used in many applications such as scene generation (Johnson et al. [2016]; Wang et al. [2017]), person re-identification, that is a very challenging problem because of the lack of cross-view paired training data and view-invariant features. Consequently, a series of methods regarding human image generation have been proposed such as Dense Pose Transfer (Neverova et al. [2018]), PG2 Network (Ma et al. [2017]) and Deformable GANs (Siarohin et al. [2018]).

Inspired by (Ma et al. [2017] and Esser et al. [2018]), our approach uses two different variables to guide the generation process: human appearance and spatial layout. On the one hand, our model needs to preserve the invariant appearance features that are characterized by human skin, hair and the color, texture of clothes, obviously, these appearance features retain while persons change their poses. On the other hand, there are a wide range of poses that include geometrical layout of an object and the viewpoint variation, and our approach can automatically represent the pose with keypoint map, provided that the position of keypoints can be estimated by 2D pose estimation methods (PAFs, Cao et al. [2016]) or extracted from

3D human datasets (Human 3.6M,Ionescu et al. [2014]).

Deep learning based generative models have shown remarkable success in image generation field, Generative Adversarial Networks (Goodfellow et al. [2014]) and Variational Auto-Encoders (VAE) (Kingma and Welling [2013]) are capable of generating realistic-looking images. Recent works have also shown that conditional generative models have the ability to guide image generation process by controlling categorical attributes or layout constraints indicated by conditioning variables such as a label, 2D pose or a sentence (Ma et al. [2017]; Johnson et al. [2016]; Esser et al. [2018]; Jaderberg et al. [2015]). For instance, Ma et al. [2017] proposed PG2 network, in which the person with one pose can be transferred to another pose represented by 2D keypoints map.

However, most of these methods have a big problem while describing posture information, that is neither 2D keypoints map nor 2D skeleton map can accurately describe certain pose of an object, since human body is a relatively complex three-dimensional structure. We need to take in account the depth information in order to prevent the loss of information concerning the spatial relationships between limbs. Depth information can also help us deal with the self-occlusion problem that frequently occurred in human image generation process.

## 1.2 Contributions

In this thesis, we propose the Three-Layer Pose Space that can incorporate depth information with regular 2D skeleton map. This approach allows us to generate 2.5D pose map that accurately represents spatial relation and geometrical layout of a certain pose. We build on Variational U-net that is recently proposed by Esser et al. [2018], which can disentangle human appearance and spatial information from an image. Moreover, we can train this model without providing samples of the same object with varying pose or appearance. The experiments on DeepFashion dataset and Human3.6M dataset demonstrate that our method

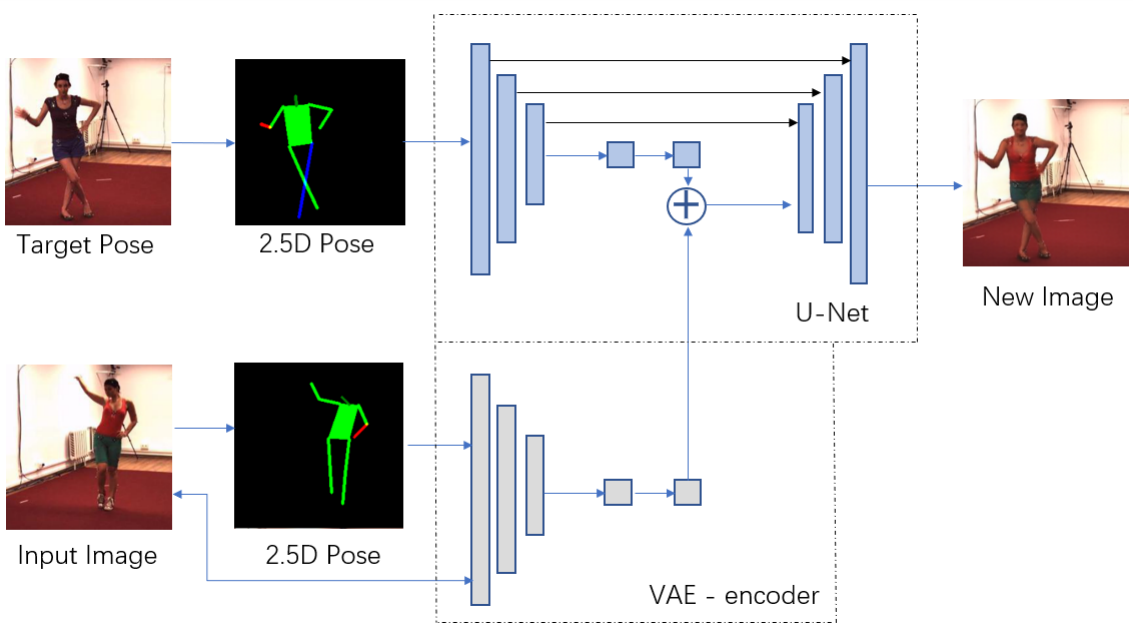


Figure 1.1: Framework of our method.

can generate more realistic human images complying with spatial and depth constraints and commonsense. The framework of our method is shown in Fig. 1.1.

## 1.3 Structure of the Thesis

The structure of this thesis is as follows. Firstly, we introduce the related works in human image generation field in Chapter 2, and then offer a detailed explanation about some popular generative models such VAEs and GAN. We present our methods and experiments on two datasets in Chapter 3. In the end, we conclude the thesis in Chapter 4.

# Chapter 2

# Background and Related Work

## 2.1 Human Pose Estimation

The task of human pose estimation is that given a static image or a video sequence, how to accurately locate the corresponding positions of the main human joints in the image, and then simulate the approximate human pose by connecting adjacent joints. In the common datasets regarding human posture estimation, the positions of 14 joints of human body are usually marked which include the head, neck, left shoulder, left elbow, left hand, left buttock, left knee, left foot, right shoulder, right elbow, right buttock, right knee and right foot. By changing the positions of these 14 joints, countless human posture can be described.

With the development of human pose estimation in recent years, more powerful methods have been proposed to deal with complex situations. However, there are still many difficulties restricting the implementation of the algorithms:

(1) Background interference. In an image, the background may contain many objects with different colors, but if some of the colors are similar to the colors of human skin or clothes, it would be difficult to distinguish between people and other objects which will lead to the wrong estimation of the possible positions of some joints in the human body.

(2) Occlusion problem. The occlusion problem is the primary problem that restricts the accuracy of human pose estimation method, and it is also the problem that researchers have been trying to solve. Because of the high degree of freedom of each joint, they might appear in any positions, which makes the complexity of human posture space very high. The occlusion problem can be divided into self-occlusion and other occlusions. Among them, self-shielding refers to the situation that some joints are covered by oneself, such as crossing hands, pocketing hands, crossing legs, kneeling or observing from the side; being shielded by other things generally refers to being occluded by other objects. All of these will lead to the algorithm cannot extract the effective appearance feature, so it will be hard for us to predict the positions of the joints through the appearance information. Another approach is to predict the positions of the occluded joints based on prior information or statistical knowledge, which greatly increases the accuracy of the algorithm in the prediction.

(3) Illumination problem. All computer vision projects cannot avoid the problem of illumination. Many existing methods and theories default that they are carried out under sufficient illumination. Therefore, the difference between illumination and non-illumination also affects the effectiveness of human pose estimation methods.

(4) The change of scale and angle. In different images, the size of human and the angle of human face to the lens vary widely. Therefore, only when these differences are dealt with, the algorithm will have better generalization ability. Therefore, feature pyramids are used in many methods.

(5) The interference of clothing. Different people have a variety of clothes, and some strange colors and styles will increase the difficulty of accurately estimating the position of joints, such as whether wearing a hat has an impact on the prediction of head position, and whether wearing skirts or pants has a great impact on the prediction of leg position.

Graphic structure model can simultaneously model the appearance of human joints and the relationships between adjacent joints, which provides accurate spatial constraint in-

formation. As shown in the following figure, graph structure model generally represents the appearance model of human body and their spatial constraints as an undirected graph $G = (V, E)$, in which $V = \{v_1, v_2, ..., v_n\}$ is used to represent the main joints of human body, and $E$ denotes the spatial constraints between them. Therefore, the graph structure model can deal with local information: appearance model and global information: the connection between adjacent components.

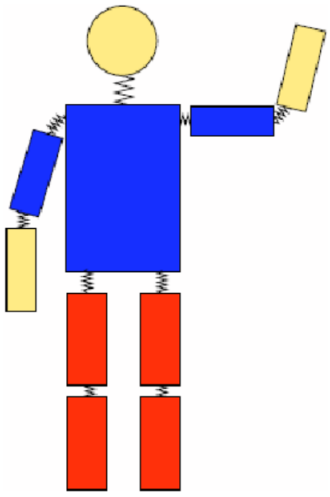

Figure 2.1: Human body with graph structure model

## 2.2 Variational Autoencoder

Auto-Encoder (AE) is a neural network model of unsupervised learning proposed in late 1980. The basic network structure of AE is composed of input layer, hidden layer and output layer, in which the dimension of input layer is the same as that of output layer. With encoderautoencoder first map the input data into a lower dimensional space and then uses the encoded data to recreate the inputs by decoder. In this network, the gradient descent algorithm is used to adjust the weights to reduce errors iteratively. Autoencoder is mainly applied on data matrix dimension reduction, feature extraction, and data denoising.

The number of neurons in the input layer is larger than that in the hidden layer. Suppose the input layer and the output layer have $n$ neurons, the hidden layer has $m$ neurons, then $n > m$, that is to say, the dimension is reduced and the data will be reconstructed. Recently, sparse autoencoder (SAE) and denoising autoencoder (DAE) are commonly used.

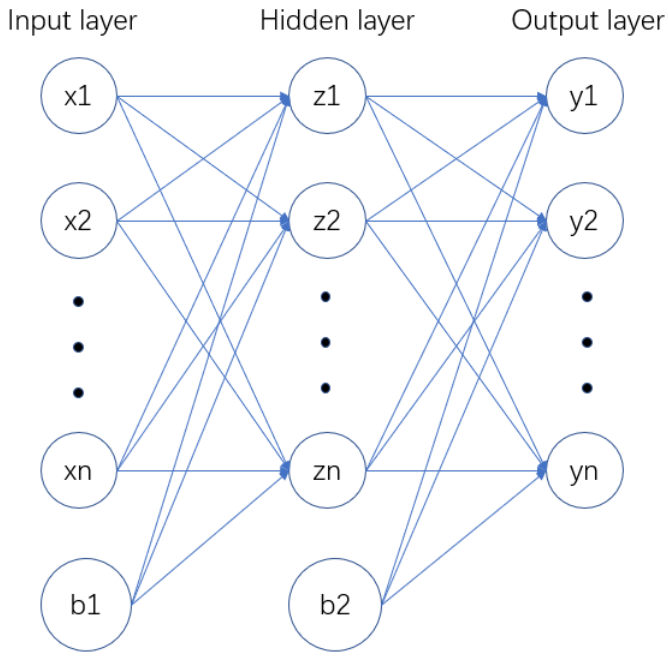


Figure 2.2: Network model of autoencoder

In AE neural network, the relationship between the edges of adjacent nodes is expressed by weight matrix, and the weights between the nodes from input layer to hidden layer are expressed by $w$. At the same time, the bias vectors of hidden layer and output layer are expressed by $q$ and $w$, where $b$ is the bias value of the $j$th node in the coding process. The network model of autoencoder is shown in Fig. 2.2.

The output value of the encoder needs to be represented by the non-linear activation function sigmoid $f(z)$, and the output layer $y$ has the following results:

$$y = \sum_{i-1}^{m} w_{i,j} a_j + b_i \tag{2.1}$$

The variational autoencoder(VAE) is an unsupervised learning generative model proposed by Diederik P. Kingma and Max Welling which has been widely used on image classification, dimension reduction, and image reconstruction. VAE is very popular because it is based on standard approximation function (neural network) and can be trained with stochastic gradient descent. Now, the commonly used depth generative models are GAN, DBN and VAE which can be used to deal with complicated data in machine learning. These complex data include handwritten digits, human face, CIFAR images, and scene physical models. The visual effect of generating digit images using a variational autoencoder is illustrated in the following figure.

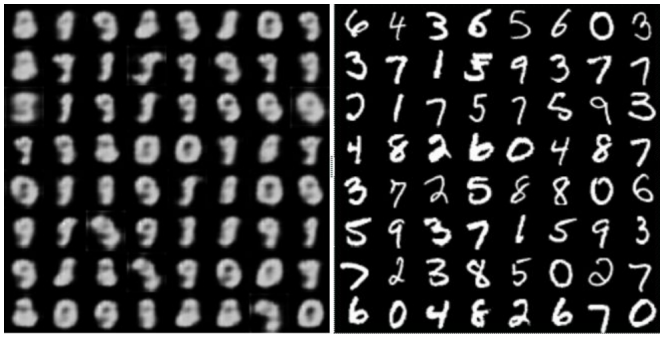

Figure 2.3: Digits image reconstruction by VAE

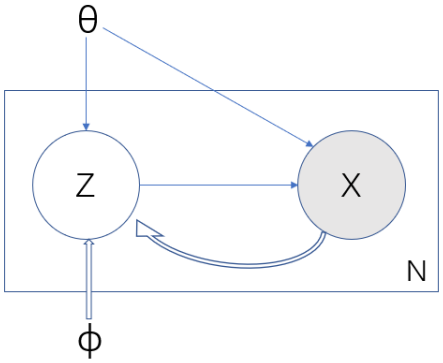


Figure 2.4: VAE graph model

Variational autoencoder has a basic model structure similar to autoencoder which includes encoder, decoder and loss function. As shown in the figure above, the observed data

is $X$, while $\widetilde{X}$ is generated by the latent variable $Z$, so $Z \to \widetilde{X}$ is a generation model, and the probability distribution is represented by $p_\theta(x|z)$. On the other hand, the recognition model is $X \to Z$, and the probability distribution is denoted by $q_\phi(z|x)$ which is similar to the encoder of autoencoder, however, they are different, because the variational encoder generates two vectors of mean $\mu$ and standard deviation $\sigma$ each time, and the autoencoder only generates an latent vector $Z$, as shown in the following figure.

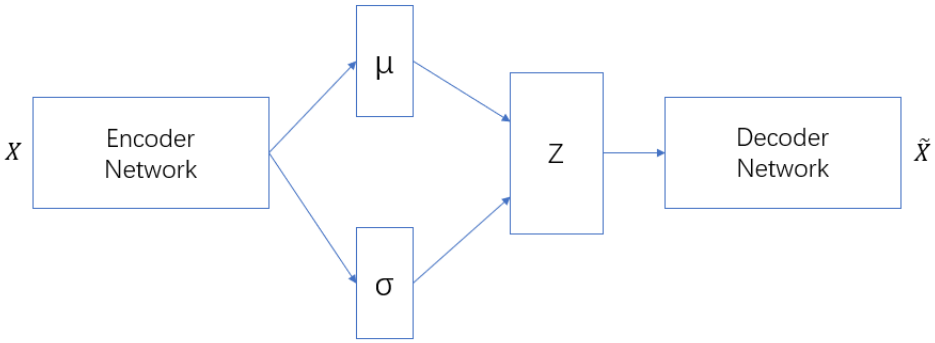


Figure 2.5: VAE model structure

The loss function of the variational autoencoder is the reconstruction loss of generated images, that is the difference between the posterior probability density in the encoder and the posterior probability density in the decoder. Therefore, in VAE, the whole network uses gradient descent to continuously adjust the $\theta$, $\phi$ in order to optimize the model. The reconstruction loss can be calculated by cross-entropy. Whether the latent variables fit the standard normal distribution is calculated by Kullback-Leibler divergence. KL divergence can describe the distance between two probability distributions. The closer the two distributions are, the smaller the KL divergence is, and vice versa.

Let's say we have a dataset $X = \{x^{(i)}\}$ and $i = 1, ..., N$, from the VAE graph structure in figure above, we know that $x^{(i)}$ is generated from $z^{(i)} \sim p_\theta(z)$and $x^{(i)} \sim p_\theta(x|z)$ ,edge similarities consists of the sum of the edge similarity of a single data point,$logp_\theta(x^{(1)}, ..., x^{(N)}) =$

$\sum_{i=1}^{N} p_\theta(x^{(i)})$ .Each data can be written as:

$$log(p^{(i)}) = D_{KL}(q_\phi(z|x^{(i)})||p_\theta(z|x^{(i)})) + L(\theta, \phi; x^{(i)}) \tag{2.2}$$

By choosing the best $p_\phi(z)$ to analyze and integrate the formulas. For this reason, firstly, by introducing the differential transformation $g_\phi(\epsilon, x)$ with stochastic noise $\epsilon$ to reparameterize Z.

$$Z^{(i,l)} = g_\phi(\epsilon, x^{(i)}), \epsilon \sim p(\epsilon) \tag{2.3}$$

More concretely, VAE uses:

$$Z^{(i,l)} = g_\phi(\epsilon, x^{(i)}) = \mu^{(i)} + \sigma^{(i)} \odot \epsilon^{(i)}, \epsilon^{(i)} \sim N(0, 1) \tag{2.4}$$

In which $l$ represents the sample of the $l$th noise epsilon, $i$ represents the $i$th data point, and $\odot$ represents the product of elements, $\mu$,$\sigma$ are the outputs of nonlinear mapping (encoder). Based on the formula above, we have:

$$q_\phi(z|x^{(i)}) = N(z; \mu^{(i)}, \sigma^{2(i)} I) \tag{2.5}$$

At the same time, it is assumed that the latent variable Z fits multivariant Gaussian distribution$q_\theta = N(z; 0, 1)$, then we get:

$$D_{KL}(q_\phi(z|x^{(i)})||p_\theta(z)) = \frac{1}{2} \sum_{j=1}^{J} (1 + log((\sigma_j^{(i)})^2) - (\mu_j^{(i)})^2 - (\sigma_j^{(i)})^2)) \tag{2.6}$$

## 2.3 U-Net

U-net was proposed by Olaf Ronneberger, Philipp Fischer and Thomas Brox in 2015, it was one of the earliest semantic segmentation algorithms using fully convolutional network, and was used for cell image segmentation under electron microscope in ISBI competition. U-Net achieved very low error rate through only 30 pictures and supplemented by data augmentation strategy, and won the championship with great advantage.

The network consists of two parts: a contracting path to obtain context information and a symmetrical expanding path to locate accurately. The symmetrical U-shaped structure it used was very innovative at that time, and to some extent affected the design of the following several partitioning networks. The name of the network was also taken from its U-shaped shape.

Compared with other common segmentation networks, U-net has a very different point: U-net uses a completely different way of feature fusion: concatenation with skip connection. Generally speaking, there are two methods for feature fusion in semantic segmentation networks:

1. The sum of corresponding points of FCN formula which corresponds to the tf. add () function in TensorFlow.

2. The U-net concatenating features from different channels which correspond to the tf. concat () function of TensorFlow.

In addition to the novel feature fusion methods mentioned above, U-net has the following advantages:

Five pooling layers can achieve multi-scale feature recognition. Also, the up-sampling part fuses the output of feature extraction part. In fact, multi-scale features can be fused together. Take the last up-sampling as an example, its features come from both the output of the first convolutional block (the same scale feature) and the output of the up-sampling

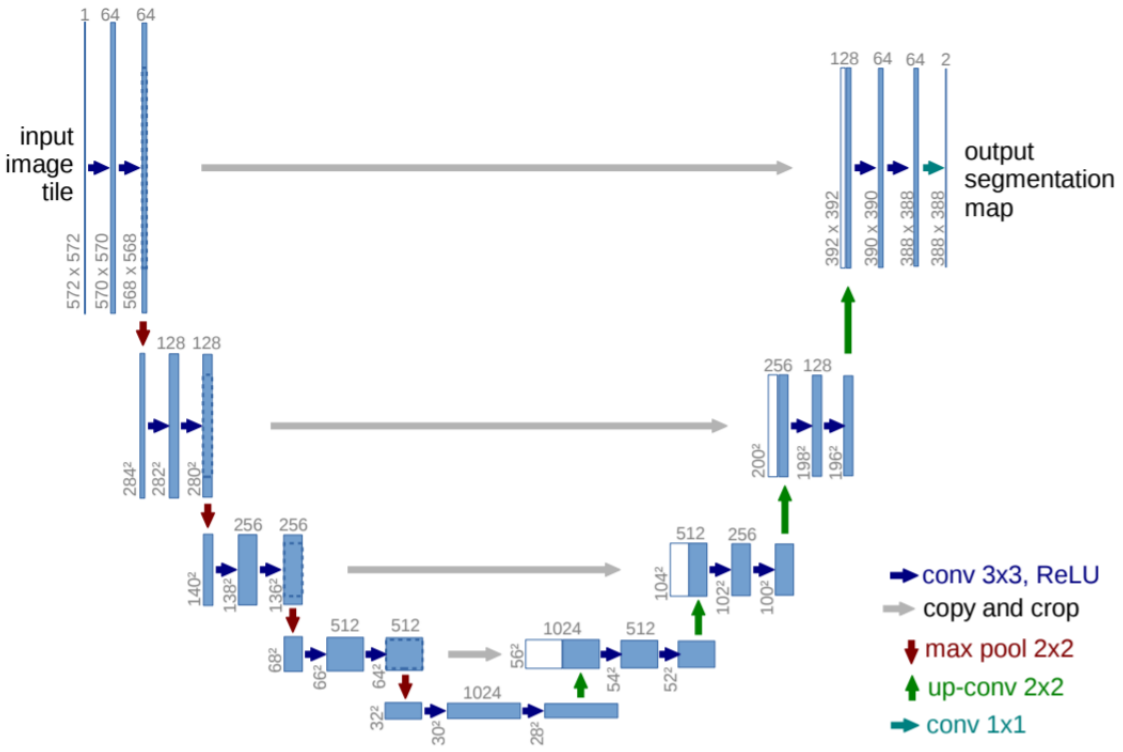


Figure 2.6: U-Net structure (Goodfellow et al. [2014]).

part (the large-scale feature). This connection runs through the whole network. As shown in the figure above, there are four skip connections in the U-net. However, the corresponding FCN only merge once in the last layer.

## 2.4 Keypoints Estimation

Given an image or a video, human pose recognition is the process of restoring the position of human joints. According to the format of input image, human pose recognition algorithms can be divided into two categories: depth map-based algorithm and RGB image-based algorithm. Compared with the limitation of image acquisition equipment due to the high requirement of depth maps, RGB-based human pose estimation algorithm has a wider

application prospect and has achieved academic results.

At present, human pose estimation algorithm based on RGB image can achieve good performance even in complex scenes. For human pose recognition, we can consider it not only as a regression problem but also as a detection problem. The difference between the two is that for the former, we expect to get accurate coordinate values (x, y); for the latter, we expect to get the corresponding heatmap. The response of different parts should be different. In practical application, human pose recognition also faces several major problems, mainly in three aspects: (1). Human body movement is flexible; (2). Changes in the background; (3). Changes in clothing. These three aspects lead to great changes in the visual information of various parts of the human body, thus bringing great challenges to human posture recognition.

There are many applications of pose estimation, such as intelligent surveillance, which is a new technology developed by using pose estimation technology and has the function of content analysis. The monitoring system with moving object recognition can omit a lot of useless information, extract the key information of human motion, and make judgments automatically according to rules, thus saving many workforce costs.

The CPM method (Wei et al. [2016]) has strong robustness, and many of the following methods are based on this method. CPM has the advantage of using sequential convolution architecture to express spatial information and texture information. The network is divided into several stages, and each stage has a part of supervisory training. In the former stage, the original image is the input, while in the latter stage, the feature map generated from the previous stage is used as input. The main purpose is to fuse spatial information, texture information and center constraints. In addition, for the same convolution architecture, multiple scales are used to process the input characteristics and responses simultaneously, which not only ensures the accuracy but also considers the distance relationship between the components.

The stacked hourglass (Newell et al. [2016]) published in the same year also achieved excellent results. For a given single RGB image, we can get the precise position of the key points of the human body, and the spatial position information of each joint point of the human body is captured by using multiscale features. The structure of the network is hourglass-like, and the hourglass module is repeatedly used to infer the position of the human bodys nodes. The structured pose (Chu et al. [2016]) proposed by Wang in 2017 is also fine-tuned on the basis of CNN. Its innovation lies in the use of geometric transformation kernels in convolution layer, which can model the dependence relationship between joint points. In addition, a bidirectional tree model is proposed, so that the feature channel of each joint can receive information from other joints. For information transmission, this tree structure can also estimate the poses of multiple people. However, the accuracy of multi-person pose estimation is not desirable.

Deepcut (Pishchulin et al. [2016]) in 2016, using the top-down method, is the first one using CNN to find all the candidate nodes, then composes a graph of these nodes, clustering the nodes in the graph to determine whom each node belongs to. In 2017, ArtTrack (Insafutdinov et al. [2017]) used the pose estimation in video tracking. The contribution of this paper is that the existing single-frame attitude estimation model is used as the basic framework, and the speed is obviously accelerated. The model still adopts the top-down method, that is, to detect body part proposals with Resnet, and then classify them into different people according to the association and spatial information.

## 2.5 Generative Model

Most deep generative models for human image generation can be categorized as either unsupervised learning method such as Variational Autoencoders((VAEs) proposed by Esser et al. [2018] or Generative Adversarial Networks (GANs)(Goodfellow et al. [2014]), VAEs are clas-

sical generative models, which have been widely used in many tasks such as text generation, image style migration. The encoder of VAEs maps the distribution of datapoints to latent vectors, and the decoder takes it as input, then outputs the parameters to the probability distribution of the data. GANs can also generate realistic-looking images based on two networks, discriminator and generator. The discriminator is used to classify "real" and "fake" images while the generator tries to construct realistic images that the discriminator cannot distinguish.

Siarohin et al. [2018] proposed Deformable GANs for the pixel-to-pixel misalignments problem by introducing deformable skip connections in the generator of GANs. Moreover, a nearest-neighbor loss is proposed in order to match the details of the generated image with the target image. PG2 network (Ma et al. [2017]) consists of pose integration network and image refinement network, an initial but coarse image of the person with the target pose can be generated from the first stage, the U-Net-like generator in the second stage can refine the initial and blurry result to generate the realistic looking human images with convincing details. Neverova et al. [2018] also adopted a two-stage approach, surface-based pose estimation and deep generative models. With the dense pose estimation system that maps pixels from the images of the object and the pose donor to a common surface-based coordinate system, both images can be brought in correspondence with each other, and then the convolutional predictive module can warp them onto the target pose.

In order to cope with the high variance in human pose, shape ,appearance, a controllable surface-based model of the human body is proposed in Lassner et al. [2017], which is trained on the semantic segmentation of the body and clothing. This model can generate the image of entirely new people with realistic clothing. However, the pose is determined by controlling a surface-based model, which can be limiting when we process videos. To solve image-to-image translation problems, conditional adversarial networks are used in Lassner et al. [2017], which is effective at synthesizing photos from label maps, reconstructing objects from edge maps,

and colorizing images, but their method is limited to the synthesis of a single, uncontrollable appearance. A different approach is taken in the pose transfer work of Esser et al. [2018], where appearance and pose are used to guide the image generation process separately, similar to Esser et al. [2018], Reed et al. [2016] utilizes the GAN framework and Reed et al. [2017] uses the autoregressive framework to provide control over shape and appearance. However, both methods cannot produce the ideal shape consistently.

# Chapter 3

# Pose Guided Image Generation

## 3.1 2D Pose Estimation

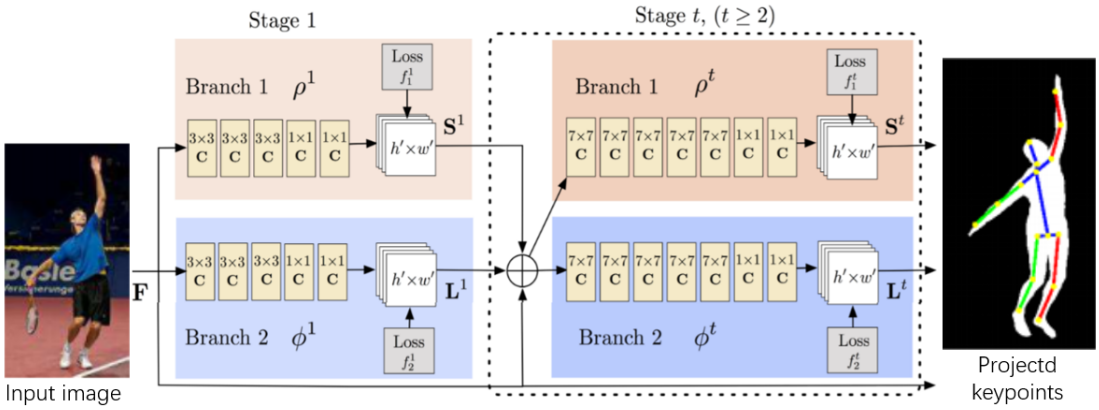


Figure 3.1: Openpose model (Cao et al. [2016]), the first branch is confidence maps of keypoints, the second branch is part affinity heatmaps for each keypoint pair.

The Fig. 3.1 illustrates the model of Openpose method (Cao et al. [2016]). We take a resized image as input, extract features through convolution network, get a set of feature maps, and then send them into two branches, extracting Parts Confidence Maps and Part Affinity Fields respectively based on CNN network. Next, we can use Bipartite Matching to combine the joint points information and get the coordinates of human keypoints.

The network structure of the model is: Firstly, the feature map is generated from 10 layers VGG. Feature map serves as the input of the two branches, merging the results of the two branches and the feature maps from the previous stage would be used for the next stage. Correspondingly, there are two loss functions in each stage, both are L2 loss. Also, there is a target mask in the loss function to prevent punishment to the true positive prediction in the training process. The network structure is divided into two branches, predicting the confidence map of human body position and part-to-part association simultaneously. The whole network is an iterative structure. The intermediate supervision is generated in each state process and will be used for the final pose estimation.

Specific to each stage: S-part detection, the foundation of building the connection between parts: When we get the confidence map of each body part, as shown in Fig. 3.2, the map has a peak for a single person; when you have more than one person, a peak represents a person. Peak points are used to estimate the exact location of body parts.

Figure 3.2: Confidence map of keypoints.

L-part association which denotes the degree of association between parts: The problem to be solved is how to get the complete post from the detected human body parts, espe-

cially when the number of people is unknown. The traditional method is to construct the relationship between body parts by confidence measure and calculate the midpoint between each pair of detected parts. The new method proposed in this model is part affinity fields - obtaining 2D affinity maps from the existing positions of components, as shown in Fig. 3.3. Then, the candidate part is used to calculate the corresponding PAF line integral to detect the components, and then the human skeleton is built with the help of PAF.

Figure 3.3: Part affinity heatmap for keypoint pairs.

To improve the efficiency of PAFs method, we also introduce Mobile-Net (Howard et al. [2017]) to achieve model miniaturization. MobileNet is based on deepwise separable convolution. The work of deepwise separable convolution is to decompose the standard convolution into depthwise convolution and pointwise convolution. The advantage of this method is that the number of parameters and calculation can be reduced greatly. The decomposition process is illustrated as follows:

The computational cost of original convolution filter is $Dk * Dk * M * N * Df * D$ in which $Df$ is the size of imput feature, depthwise convolution works as filter, the size is $(Dk, Dk, 1, M)$ and the computational cost is: $Dk * Dk * M * Df * Df$.

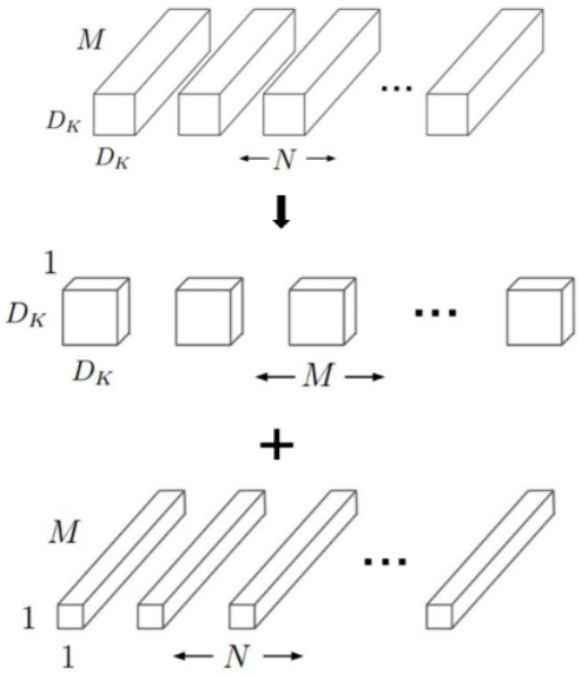


Figure 3.4: The standard convolutional filters are replaced by two layers: depthwise convolution and pointwise convolution.

Pointwise convolution is responsible for switching channels, the size is $(1, 1, M, N)$ and the computational cost is: $M * N * Df * Df$. So, we can get:

$$overall\,cost = Dk * Dk * M * Df * Df + M * N * Df * Df \tag{3.1}$$

Compared with the original convolution filter, the computational complexity can be put down a lot.

We reconstruct openpose model with mobilenet, evaluate the new model on one benchmark for multiperson pose estimation: the COCO 2016 keypoints challenge dataset (Lin et al. [2014]). We test two models on a machine without (i5- 8250U), then compare their Heatmap Loss (last layer), PAFmap Loss (last layer) and inference time.

Table 3.1: The results after modification.

| Model | Original | MobileNet |
|---|---|---|
| Heatmap Loss @25k iterations | 21.3 | 24.9 |
| PAFmap Loss @25k iterations | 43.6 | 50.2 |
| Inference Time @i5-8250U, No GPU | 4-5s | 0.5-0.8s |

With the modification, the Heatmap Loss and PAFmap Loss rise more than 15 percent, but the inference time drops from about 5s to about 0.7s. So, the modified model can greatly improve the efficiency of our algorithm.

## 3.2 Three-Layer Pose Space

In this section, we describe the architecture of our generator and how to generate 2.5D poses with the Three-Layer pose space, in order to build an end-to-end model. Our pose space can automatically transfer the 3D coordinates of human keypoints into the corresponding 2.5D skeleton map which depicts the essential characteristics of the pose of the person. For the image generation process, we use VAE to extract the invariant appearance features and then fill it into the new image.

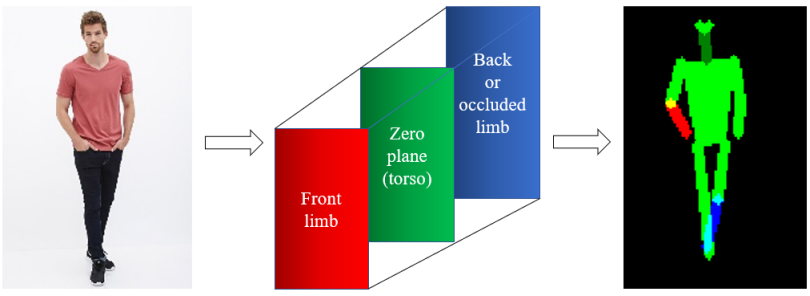


Figure 3.5: Three-Layer Pose Space.

Integrating depth info with the 2D pose map is a challenging task. Point cloud can accurately describe the pose of an object, however, it is computationally expensive especially as a label and hard to achieve without complete data. In this paper, we propose a simplified method called Three-Layer pose space that can roughly describe the spatial front-back relations of limbs. Firstly, in order to build human skeleton map which consists of a sequence of 2D human keypoints, we apply a state-of-the-art art 2D pose estimation method (Cao et al. [2016]) on Deepfashion dataset to obtain the approximate keypoints positions, and depth info are required to be labeled manually. With Human 3.6M dataset which offers 3D coordinates, pose space can automatically accomplish the pose projection without manual annotation. The detailed process regarding this transformation is shown in Fig. 3.5.

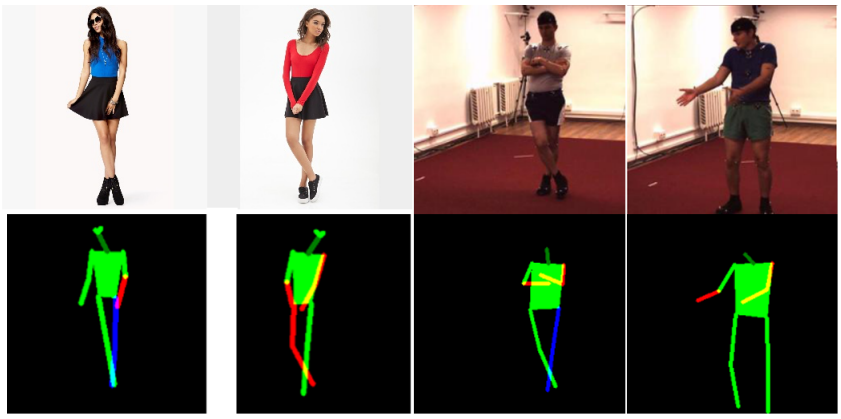

Figure 3.6: Pose extraction example.

In our method, the human body is separated into torso and other 8 subregions that include front arm, rear arm, calf, thigh on the left and right. The depth info of each limb is denoted by three different colors, blue (-1) indicating the limb is being occluded, red (1) implying the limb is in the front of torso or other limbs, green (0) denoting the limb and torso are on the same plane, plane where torso is located is considered as zero plane.

Specifically, our algorithm can easily transfer 3D coordinates to 2.5D skeleton map with Human 3.6M dataset. Firstly, we can find where occlusion happens by calculating and

comparing $(x_i, y_i)$ of each limb, and then $z_{i,j}$ can be used to tell us which limb is being occluded and which one is in the front of the torso. On the other hand, the thickness of human body must be considered, it plays a important role when deciding spatial relations. This value depends on the specific dataset and the corresponding motion capture system. Based on these strategies, our model can assign the depth label (the color) to the 2D skeleton map as shown in Fig. 3.7.

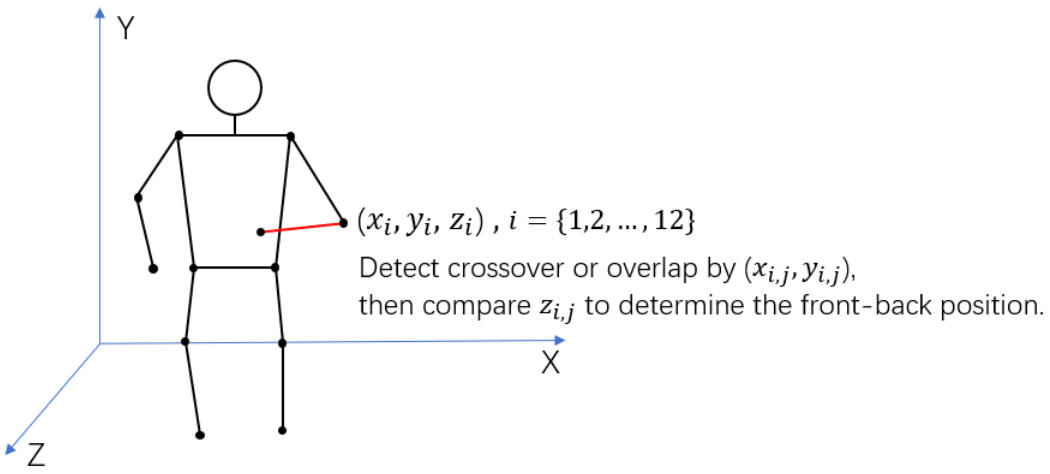


Figure 3.7: Transformation for 3D coordinates to 2.5D skeleton map.

With these rules, our method can incorporate spatial front-back information into the 2D skeleton map, and then construct our RGB stickman image that can be used as a label to control the human image generation process. Some examples are shown in Fig. 3.6 and Fig. 3.8.

## 3.3 Conditional Variational U-Net

In the previous section, we have mentioned that Variational U-net model is suitable for pose transfer, and it can disentangle human appearance information $z$ and spatial information $y$ from an image. Let $x$ be the input image, we can see the overall network below.

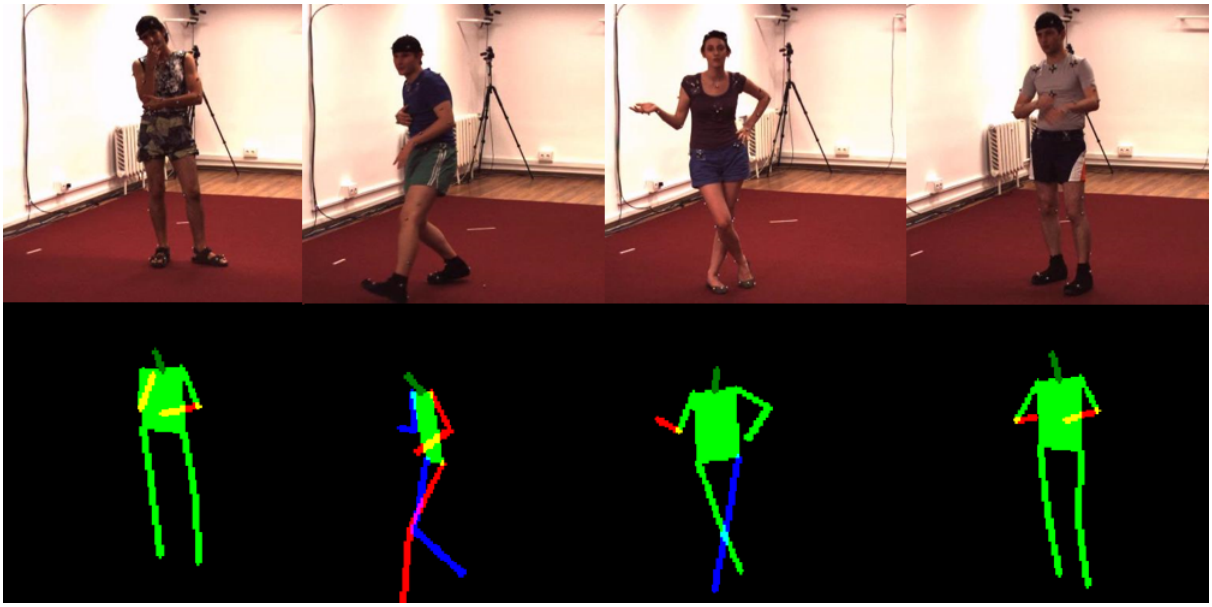

Figure 3.8: Another pose extraction example.

The generator is fed with the input image $x$ and the corresponding 2.5D pose map $y$, drawn from three-layer pose space we introduced before, $z$ is latent variable that represents appearance info. To preserve spatial information given by 2.5D pose map, we adopt U-net architecture, as the skip connections can propagate information from encoder to decoder directly without any loss.

As to appearance info $z$, we model $q(z|x, y)$ as a parametric Gaussian distribution according to (Rosca et al. [2017]) whose the parameters are estimated by an encoder network of VAE, as shown in Fig. 3.9. Appearance $Z$ is required to extract enough appearance information from the image without any spatial information, which means the encoder of VAE must be invariant to pose information. In this case, we can minimize the Kullback-Leibler divergence between $q(z|x, y)$ and $p(\hat{z}|y)$ which represents the distribution of latent variable of spatial information.

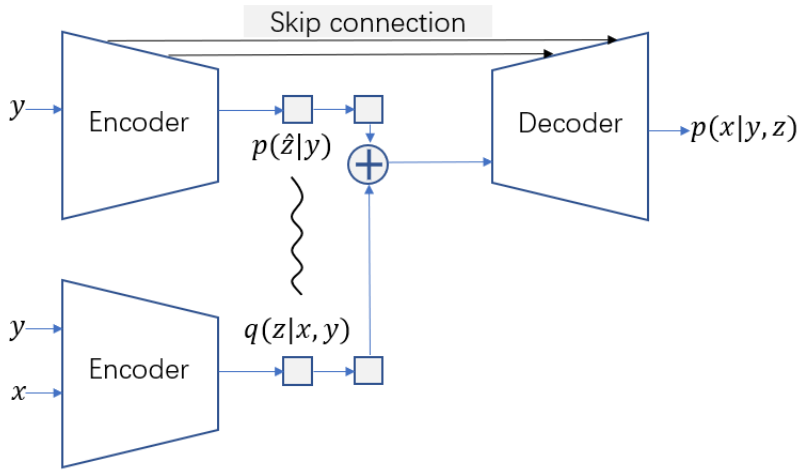


Figure 3.9: Model training.

$$loss\ function = -KL(q(z|x,y)||p(\hat{z}|y)) + perceptual\ loss \tag{3.2}$$

To compare the generated image with the original image, we adopt the perceptual loss from Chen and Koltun [2017].

Generating images from semantic information is a problem with incomplete constraints. There are many kinds of images corresponding to the same semantic feature, so even in the training process, the researchers regard the images corresponding to the layout objects as reference images.

For training problems with incomplete constraints, we hope to find the most suitable loss function. If the model directly compares the reference image with the generated image from pixel-to-pixel, it will result in a great penalty for the sufficiently real content, such as when the color of the generated car is different which is actually unnecessary. So, we choose the method of content representation, or perceptual loss and feature matching, which corresponds to the feature matching activation in the visual network, thus maintaining enough distance from the low-level features of the reference image. The network architecture of our model is

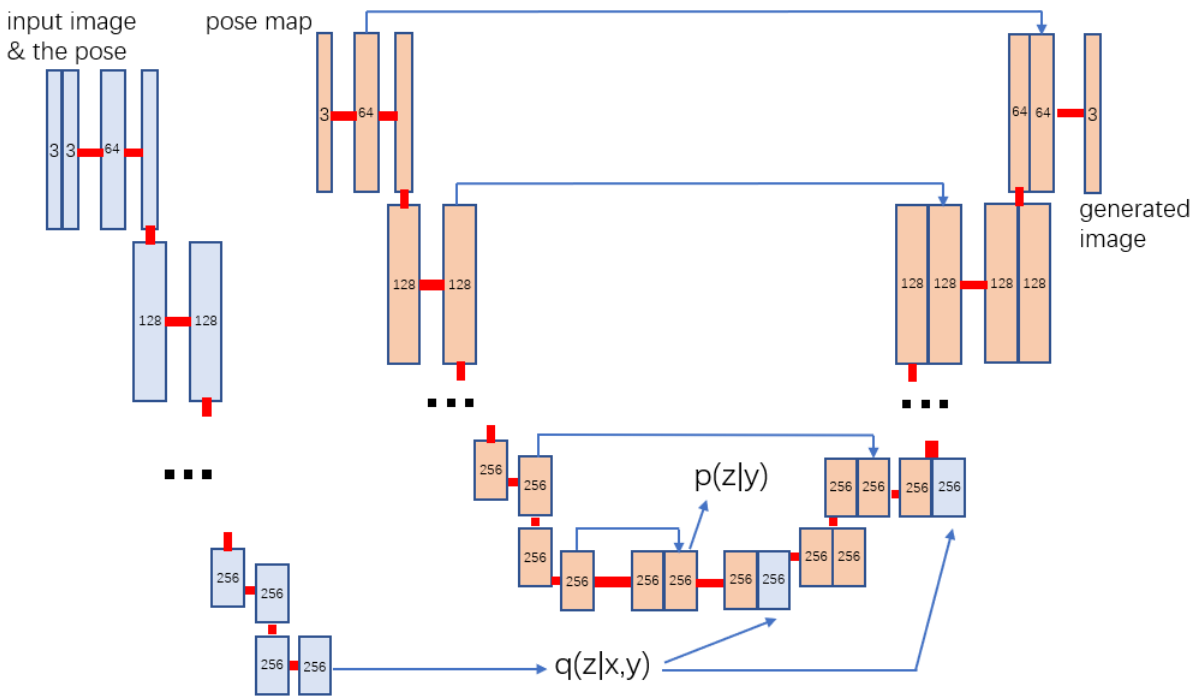


Figure 3.10: Network architecture of our model.

shown in Fig. 3.10.

Specifically, Chen and Koltun [2017] proposes a new way, using a VGG-19 image perception model, to extract image features from different layers which can be used to calculate training loss, thus the model contains both low-level fine-grained features such as edges and colors in image features and high-level overall layout features such as objects and categories, so as to construct a comprehensive and powerful loss function.

Let $\Phi$ be a trained visual perception network (we use VGG-19, Simonyan and Zisserman [2015]), layers in the network represent an image at increasing levels of abstraction: from edges and colors to objects and categories. By learning both fine-grained details and more

global part arrangement, this network is a good choice for measuring perceptual similarity.

$$perceptual\ similarity = \sum_{k} \lambda_k ||\Phi_k(x) - \Phi_k(G(\hat{y}, z))|| \tag{3.3}$$

By combining KL divergence equation (3.1) and reconstruction loss equation (3.2), we can formulate the final loss function as:

$$loss\ function = -KL(q(z|x, y)||p(\hat{z}|y)) + \sum_{k} \lambda_k ||\Phi_k(x) - \Phi_k(G(\hat{y}, z))|| \tag{3.4}$$

where $\lambda_k$, $k$ are hyper-parameters that control the contribution of the different layers of $\Phi$ to the total loss, and $G$ represents the decoder of our model that can generate new images.

## 3.4 Evaluation Metrics

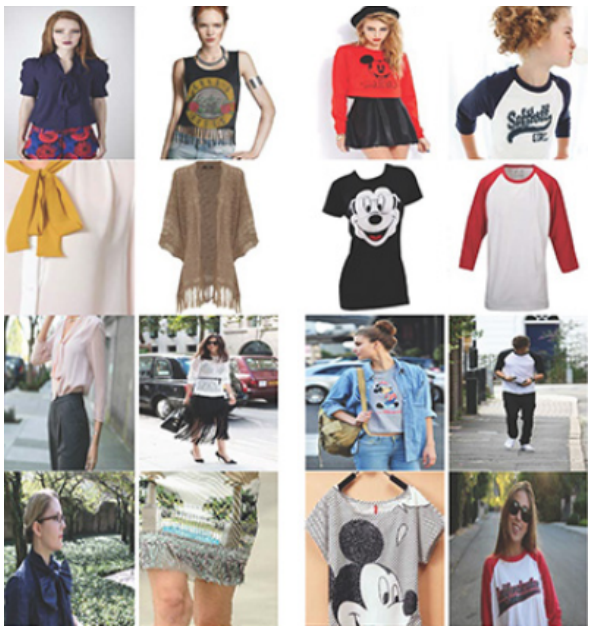

Figure 3.11: Deepfashion dataset samples.

We test our model on two human datasets (DeepFashion and Human 3.6M) as shown

in Fig.3.11 and Fig.3.12, which contain person images with diverse poses. Since there is no depth labels in the DeepFashion dataset, we label more than 1,000 images to do the test. The images have a resolution of 256x256 pixels. Human 3.6M dataset uses 4 high-resolution motion capture cameras which are located in different corners to collect video at 50 Hz. With this motion capture system which records 3D coordinates of subjects, pose data is given with respect to a skeleton of 32 joints. We extract the images from the videos at 10 Hz, then match the images with the skeleton data in order to form about 10,000 pairs of input images with corresponding joints coordinates. Our model can achieve end-to-end training with Human 3.6M dataset.

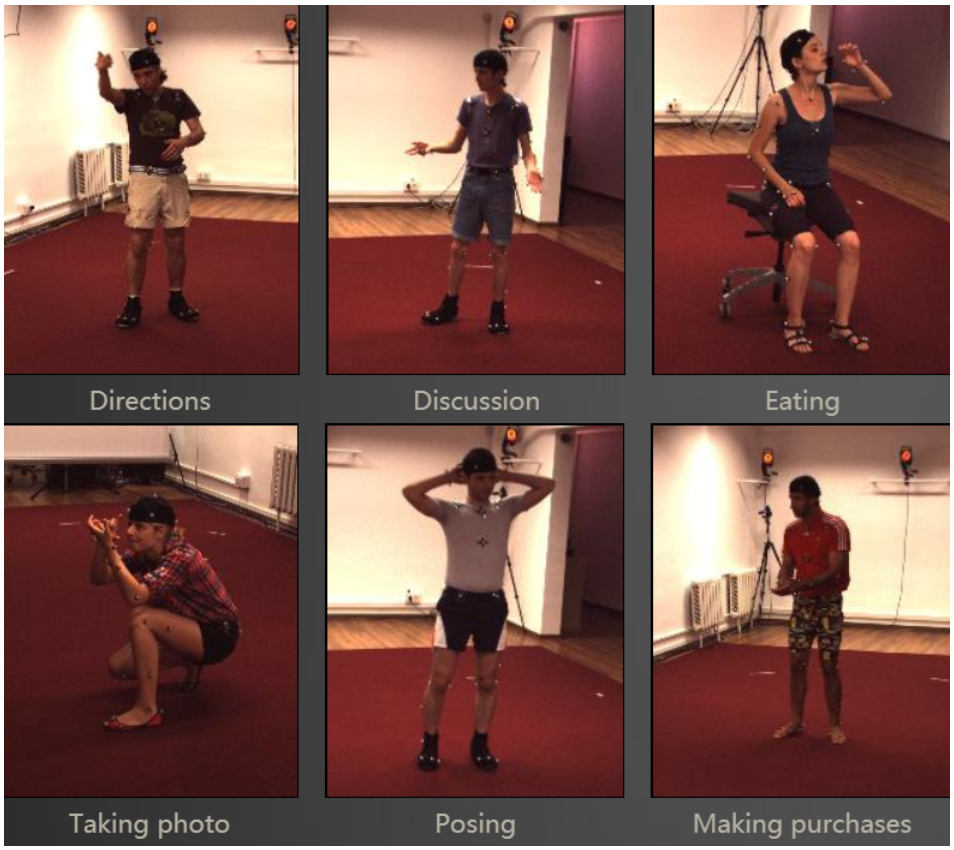


Figure 3.12: Human3.6M dataset samples.

In general, there is no standard criterion that allows us to evaluate the generated images from the perspective of both structural fidelity and photorealism. But some perceptual

metrics are proposed recently and have widely used in human pose estimation fields and human image generation programs. The geometry of the generated images can be evaluated by perception correlated Structural Similarity metric (SSIM)(Wang et al. [2004]).

SSIM, which is called structural similarity index, is a measure of similarity between two images. The index was first proposed by Laboratory for Image and Video Engineering of the University of Texas at Austin. If the two images are original and reconstructed respectively, then SSIM algorithm can be used to evaluate the quality of the reconstructed image.

How SSIM characterizes the similarity:

$$\begin{aligned} L(X,Y) &= \frac{2\mu_X\mu_Y + C_1}{\mu_X^2\mu_Y^2 + C_1} \\ C(X,Y) &= \frac{2\sigma_X\sigma_Y + C_2}{\sigma_X^2\sigma_Y^2 + C_2} \\ S(X,Y) &= \frac{\sigma_{XY} + C_3}{\sigma_X\sigma_Y + C_3} \end{aligned} \tag{3.5}$$

$\mu_X$ and $\mu_Y$ represent the mean of the intensity values in image X and Y respectively, $\sigma_X$ and $\sigma_Y$ denote the corresponding standard deviations, $\sigma_X^2$ and $\sigma_Y^2$ represent the covariances,$C1$, $C2$ and $C3$ are constants to keep the denominator stable. Usually $C1 = (K1 * L)^2$, $C2 = (K2 * L)^2$, $C3 = C2/2$, $K1 = 0.01$, $K2 = 0.03$, $L = 255$ (the dynamic range of pixel values). Finally, we have:

$$SSIM(X,Y) = L(X,Y) * C(X,Y) * S(X,Y) \tag{3.6}$$

So the structural similarity index defines the structural information as the attribute of the object structure which is independent of brightness and contrast in the scene, and the similarity loss is measured by the combination of brightness, contrast, and structure. The mean is used to estimate brightness, the standard deviation is used to estimate contrast, and the covariance is used to measure structural similarity.

Table 3.2: Evaluation metrics.

| method | DeepFashion | | | |
|---|---|---|---|---|
| | IS | | SSIM | |
| | mean | std | mean | std |
| real data | 3.415 | 0.399 | 1.000 | 0.000 |
| PG2 G1+D | **3.091** | - | 0.761 | - |
| PG2 G1+G2+D | 3.090 | - | 0.762 | - |
| ours | 3.082 | 0.248 | **0.785** | 0.067 |

According to previous works, we also offer the results of Inception scores (IS)(Salimans et al. [2016]). Inception Score evaluates generated images from two aspects: 1. Whether the generated pictures are clear; 2. Whether the generated pictures are diverse. Even if the generated images are clear enough, we still need to see whether the model can generate enough diverse images. Some generative models can only generate a limited range of clear pictures, and fall into the so-called mode collapse. In Tab.3.2, we compare our method with Ma et al. [2017]. As a baseline for quantitative comparison, we use PG2(Ma et al. [2017]). PG2 does not report mask-based metrics on Deepfashion dataset. So, we use the available codes and network weights published by the authors to generate new images. On the DeepFashion dataset, our approach reports the highest performance over SSIM. Specifically, our SSIM mean value is higher than that obtained through PG2. But our method returns a slightly lower IS average value. However, as our method differs from others by trying to generate images that contain more spatial relationship information, these metrics cannot tell the difference from this aspect.

Figure 3.13: The generated images sampled from other 2D pose guided methods (Siarohin et al. [2018], Ma et al. [2017]), as indicated by the arrow, the self-occluded limbs look blurry and mixed, and we cannot distinguish the front back positions of the limbs because of the lack of depth information.

## 3.5 Generated Image Samples

In this section, we show some generated image samples to validate the impact of depth information on the qualitative results. Firstly, we test our model on DeepFashion dataset, to prove our 2.5D pose can deal with self-occlusion problems and help us easily distinguish the spatial relationships of the limbs. We draw two pairs of different 2.5D poses with the same 2D projections as the labels to guide the image generation process. With any other 2D pose guided methods, the generated images will be the same for sure. However, our model generates different new images with clear front-back relations of the limbs. In Fig. 3.12 and Fig. 3.13, we swap the front and back positions of the left and right legs in the target pose on purpose. In general, the methods proposed before cannot cope with this kind of limb crossover problems because of the lack of depth information. The generated images always look blurry or mixed in the self-occluded parts, as you can see in Fig. 3.11. However, our model that takes into account the depth information apparently obtains better performance.

Fig.3.14 and Fig.3.15 show how our model solves leg occlusion problems in generated

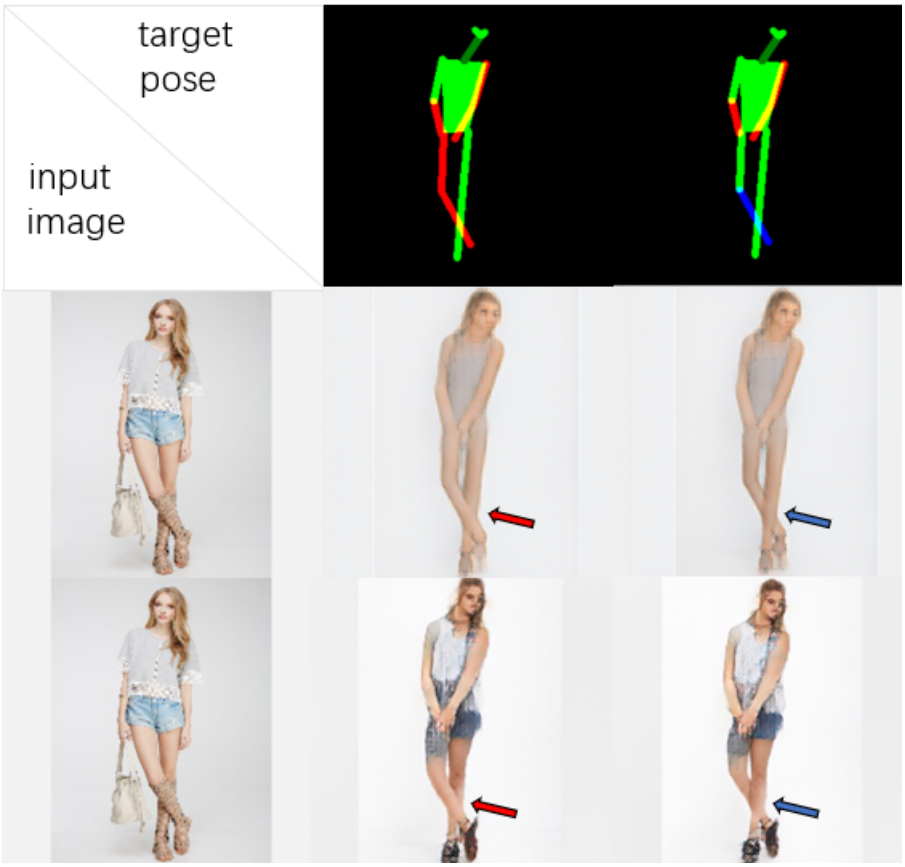


Figure 3.14: Our target poses consist of the same 2D pose maps and different depth information, Column 1 represents the input image, Row 2,3 represent the generated images after 2,000 iterations and 28,000 iterations, as you can see, we can easily tell the difference about the front back positions of the human leg even from the preliminary results, the self-occluded part is not blurry anymore, spatial relations can be easily distinguished.

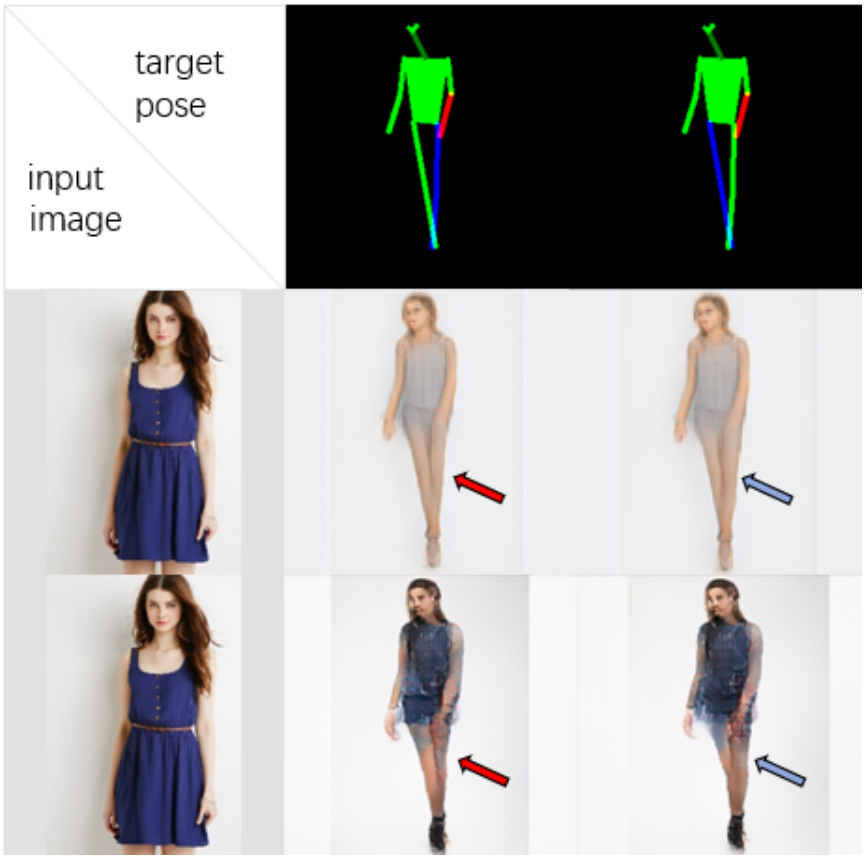


Figure 3.15: Another example of 2.5D pose guided image generation, we set the right calf and thigh of the first pose as being occluded (-1) as opposite as the second pose, the generated images reflect the corresponding changes.

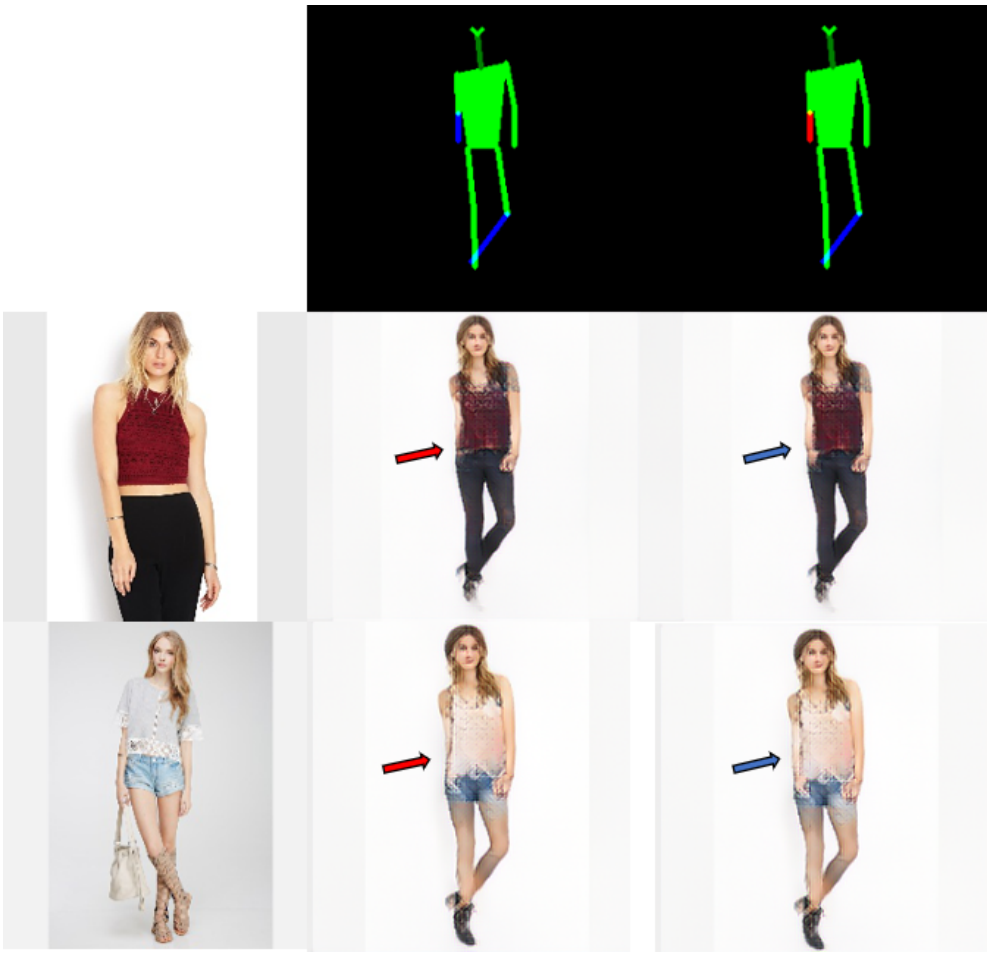

Figure 3.16: In this case, we swap the position of the right arms in the pose map – one is in the front of torso, and the other is occluded by the torso. The pictures we generated clearly illustrate this difference.

images. Furthermore, arms are often occluded by the torso in many human images, in Fig. 3.16, and we can see that our model can easily distinguish the spatial relations between arms and the torso, then generate corresponding target images.

With Human 3.6M dataset, our model can achieve better qualitative results, cause the 3D joints coordinates are collected by motion capture system instead of estimated by PAFs method. Also, we sample more than 10,000 images of 11 persons captured from 6 different surveillance cameras which are ideal for our training. The results are shown in Fig. 3.18

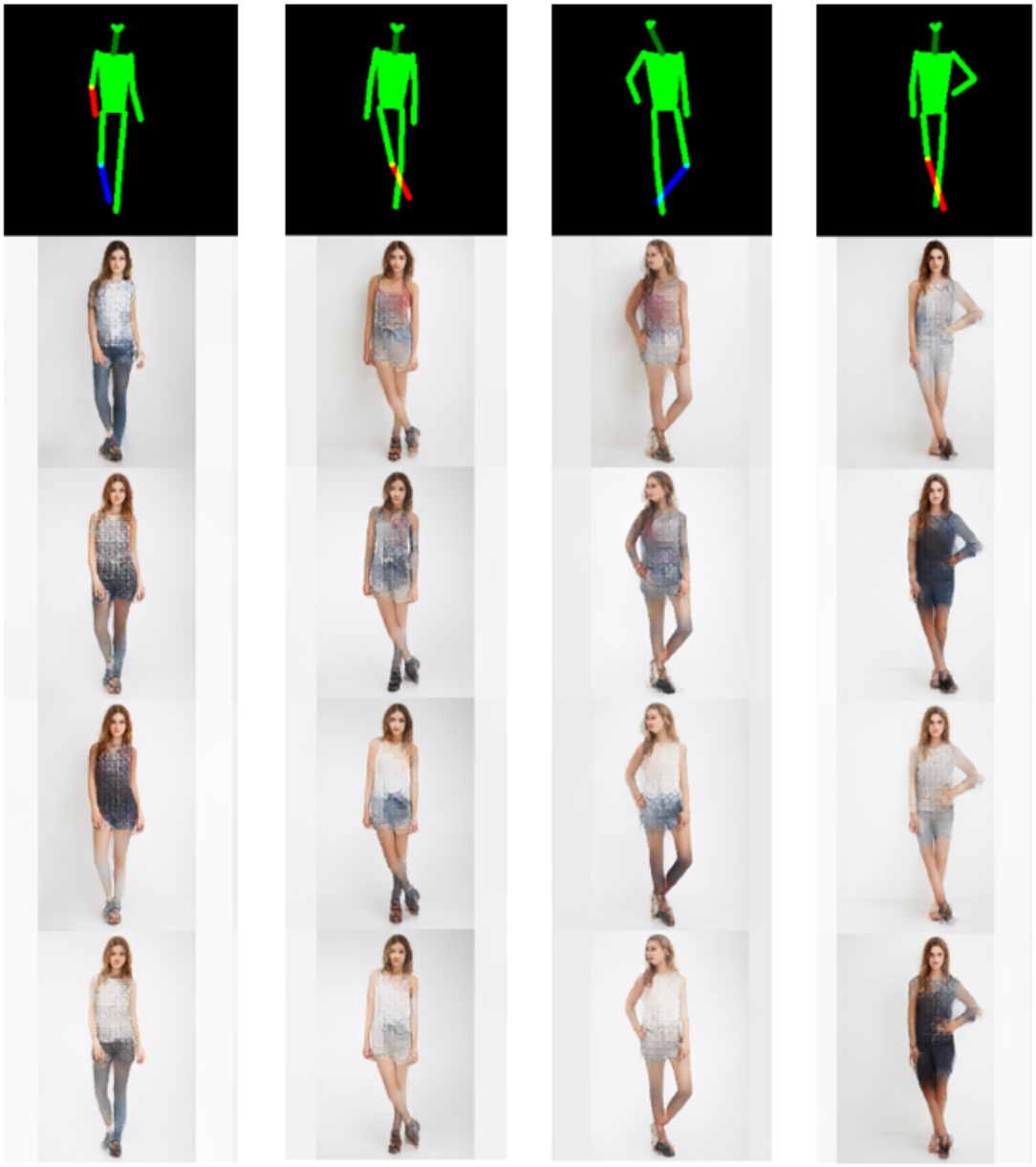

Figure 3.17: Other examples with DeepFashion dataset.

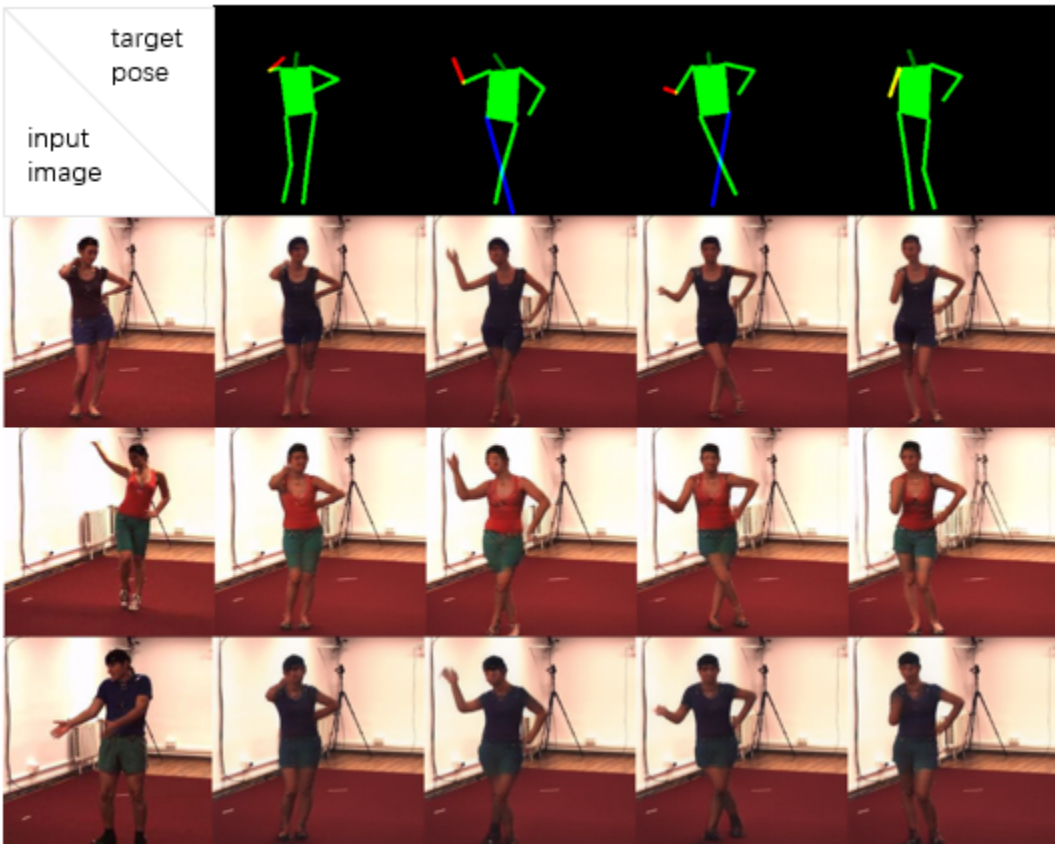


Figure 3.18: Pose transfer on Human 3.6M dataset.

and Fig. 3.19, obviously, our model can accurately capture the appearance information and pose information, then achieve pose transfer. Spatial relations are very clear in all generated images, which justifies our hypothesis, that is depth information playing an important role in human image generation. We can generate more realistic looking images by using depth info reasonably.

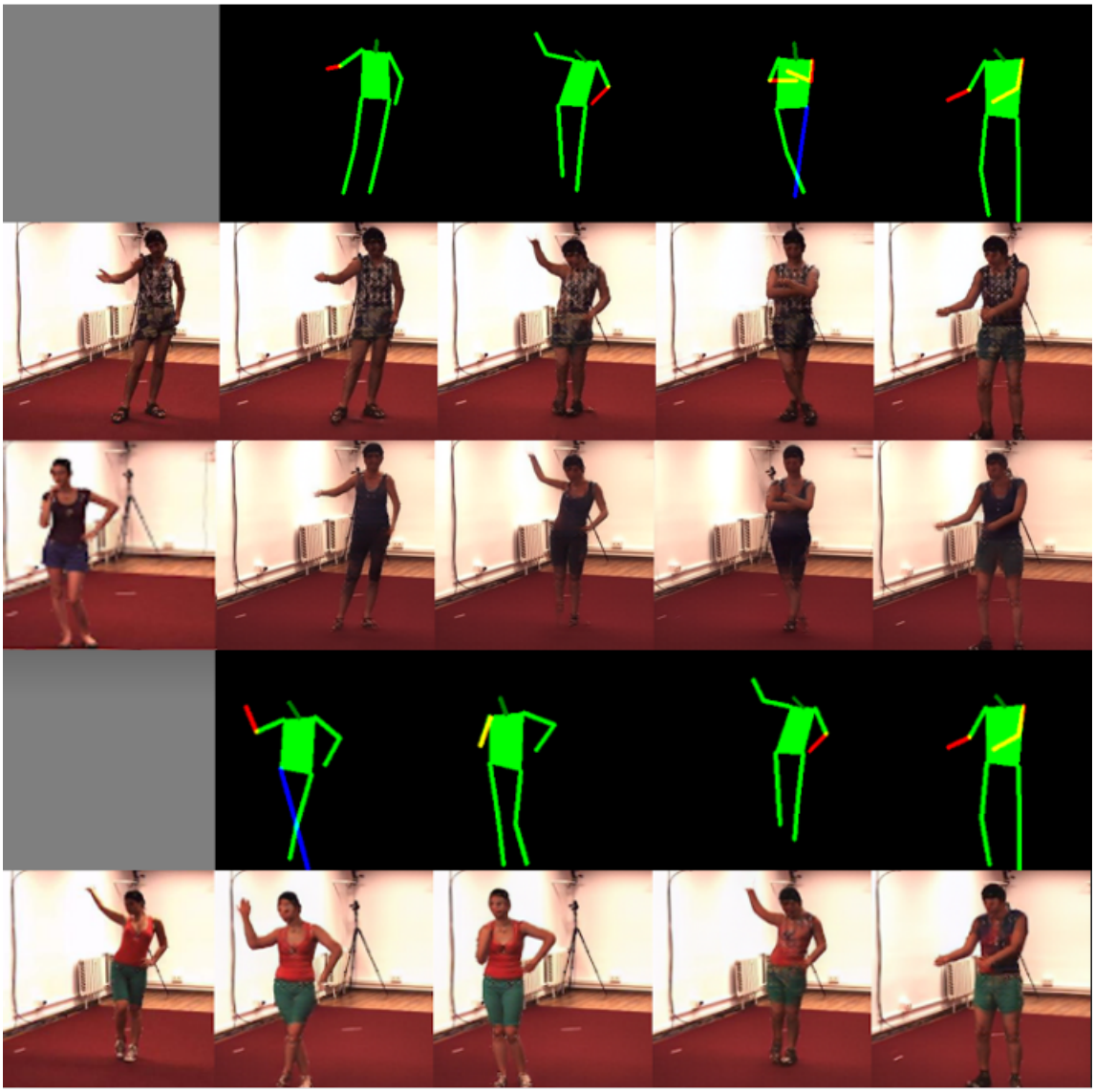

Figure 3.19: Other examples on Human 3.6M dataset.

# Chapter 4

# Conclusion and future work

In this thesis, we propose a 2.5D pose guided human image generation method that integrates depth information with 2D poses. With Three-Layer Pose Space we present, our model can transfer 3D coordinates of human joints into a 2.5D skeleton map. Experiments show that our model can solve self-occlusion problems commonly happened in human image generation field, and generate new images containing more spatial information. Also, our model can achieve end-to-end training on datasets with 3D coordinates, such as Human 3.6M dataset.

The depth generative model has shown excellent performance in the field of image generation. However, because these models directly generate the target image without modeling the complex interaction between its intrinsic shape and appearance information, there will be performance degradation in space conversion. But, the variational U-net model we use in this paper can independently encode and sample the appearance information and pose information, so it can achieve accurate pose transfer. With the Three-Layer Pose Space we propose, the depth information can be added to help the model generate images from different viewpoints and under different self-occlusion conditions.

This paper proves that depth information can play an important role in human image generation. Even simplified depth information still has a great impact on the results. We be-

lieve this contribution can inspire image generation projects in other fields. The disadvantage is that the change and angle of human face are relatively separated from the human body structure. If we want to generate more realistic-looking images, the human face must be independently modeled and stitched, which is also one of the future improvement directions of our project.

# Bibliography

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *CoRR*, abs/1611.08050, 2016. URL `http://arxiv.org/abs/1611.08050`.

Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. *CoRR*, abs/1707.09405, 2017. URL `http://arxiv.org/abs/1707.09405`.

Xiao Chu, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Structured feature learning for pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4715–4723, 2016.

Patrick Esser, Ekaterina Sutter, and Björn Ommer. A variational u-net for conditional appearance and shape generation. *CoRR*, abs/1804.04694, 2018. URL `http://arxiv.org/abs/1804.04694`.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. *arXiv e-prints*, art. arXiv:1406.2661, Jun 2014.

Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

Eldar Insafutdinov, Mykhaylo Andriluka, Leonid Pishchulin, Siyu Tang, Evgeny Levinkov, Bjoern Andres, and Bernt Schiele. Arttrack: Articulated multi-person tracking in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6457–6465, 2017.

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.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. *CoRR*, abs/1506.02025, 2015. URL `http://arxiv.org/abs/1506.02025`.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual Losses for Real-Time Style Transfer and Super-Resolution. *arXiv e-prints*, art. arXiv:1603.08155, Mar 2016.

Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv e-prints*, art. arXiv:1312.6114, Dec 2013.

Christoph Lassner, Gerard Pons-Moll, and Peter V. Gehler. A generative model of people in clothing. *CoRR*, abs/1705.04098, 2017. URL `http://arxiv.org/abs/1705.04098`.

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 *European conference on computer vision*, pages 740–755. Springer, 2014.

Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. *CoRR*, abs/1705.09368, 2017. URL `http://arxiv.org/abs/1705.09368`.

Natalia Neverova, Riza Alp Güler, and Iasonas Kokkinos. Dense pose transfer. *CoRR*, abs/1809.01995, 2018. URL `http://arxiv.org/abs/1809.01995`.

Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European Conference on Computer Vision*, pages 483–499. Springer, 2016.

Leonid Pishchulin, Eldar Insafutdinov, Siyu Tang, Bjoern Andres, Mykhaylo Andriluka, Peter V Gehler, and Bernt Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4929–4937, 2016.

Scott E. Reed, Zeynep Akata, Santosh Mohan, Samuel Tenka, Bernt Schiele, and Honglak Lee. Learning what and where to draw. *CoRR*, abs/1610.02454, 2016. URL `http://arxiv.org/abs/1610.02454`.

Scott E. Reed, Aäron van den Oord, Nal Kalchbrenner, Sergio Gomez Colmenarejo, Ziyu Wang, Dan Belov, and Nando de Freitas. Parallel multiscale autoregressive density estimation. *CoRR*, abs/1703.03664, 2017. URL `http://arxiv.org/abs/1703.03664`.

Mihaela Rosca, Balaji Lakshminarayanan, David Warde-Farley, and Shakir Mohamed. Variational Approaches for Auto-Encoding Generative Adversarial Networks. *arXiv e-prints*, art. arXiv:1706.04987, Jun 2017.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pages 2234–2242, 2016.

Aliaksandr Siarohin, Enver Sangineto, Stéphane Lathuilière, and Nicu Sebe. Deformable gans for pose-based human image generation. *CoRR*, abs/1801.00055, 2018. URL `http://arxiv.org/abs/1801.00055`.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. URL `http://arxiv.org/abs/1409.1556`.

Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. *CoRR*, abs/1711.11585, 2017. URL `http://arxiv.org/abs/1711.11585`.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016.