Pose Guided Human Image Synthesis by View Disentanglement and Enhanced Weighting Loss

Mohamed Ilyes Lakhal[1](✉), Oswald Lanz[2], and Andrea Cavallaro[1]

[1] CIS, Queen Mary University of London, London, UK
{m.i.lakhal,a.cavallaro}@qmul.ac.uk
[2] TeV, Fondazione Bruno Kessler, Trento, Italy
lanz@fbk.eu

Abstract. View synthesis aims at generating a novel, unseen view of an object. This is a challenging task in the presence of occlusions and asymmetries. In this paper, we present View-Disentangled Generator (VDG), a two-stage deep network for pose-guided human-image generation that performs coarse view prediction followed by a refinement stage. In the first stage, the network predicts the output from a target human pose, the source-image and the corresponding human pose, which are processed in different branches separately. This enables the network to learn a disentangled representation from the source and target view. In the second stage, the coarse output from the first stage is refined by adversarial training. Specifically, we introduce a masked version of the structural similarity loss that facilitates the network to focus on generating a higher quality view. Experiments on Market-1501 and DeepFashion demonstrate the effectiveness of the proposed generator.

Keywords: Pose-guided view synthesis · Generative models · Structural similarity

1 Introduction

View synthesis is of considerable interest for data augmentation, animation, augmented and virtual reality. Generating a novel view of a human [1,2] is more challenging that generating a view for a rigid 3D object [3,4], especially when scene parameters are unavailable.

The appearance of an object from an unseen view can be generated with geometry or transformation-based methods [5]. Geometry-based methods generate novel views by scaling, rotation, translation, and non-rigid deformations of specific 3D objects [6,7]. A limitation of geometry-based methods is the structure of the rendered 3D objects, which are characterized by shape invariance and symmetry [5]. Transformation-based methods encode directly the correspondence between input and output images to synthesize the view [3,4,8,9].

L. Leal-Taixé and S. Roth (Eds.): ECCV 2018 Workshops, LNCS 11130, pp. 380–394, 2019.
https://doi.org/10.1007/978-3-030-11012-3_30

Synthesizing an image of a person from an arbitrary pose is challenging due to occlusions and the potentially complex human pose changes from the source to target view. Unlike 3D-object-based synthesis [8,9], synthesizing the image of a human from another pose and view cannot always make use of extrinsic camera parameters or information about changes in illumination. Moreover, there might be considerable differences in *image quality* in the dataset; the *scale* difference between the input pair and output can be large; some body parts can be *occluded* and some poses can appear infrequently (*e.g.,* crossed arm). These factors have an impact on the quality of the synthesized image.

Human image synthesis methods can be classified as view specific or pose guided. *View-specific* methods synthesize human images into a pre-defined set of views (*e.g.* front, back, side [10]). *Pose-guided* methods impose constraints over the input view using a target 2D pose (defined as a set of 2D locations of the body joints) as a guidance in the generation [1,2,5,11,12]. Recent approaches [2,10] force the input image of the human body and its target pose to be encoded into a joint feature space. This solution is undesirable as the input image and the target pose are fed to the same encoder and their mixing in early layers can lead to misalignment in the decoder. This problem becomes even more critical if the input and the target have different scales and spatial locations. In fact, because of this variation, the receptive field of the convolutional layers may not capture the change of body appearance between the input and the target pose [1].

To address this problem, in this paper we propose a two-stage deep encoder-decoder pipeline that explicitly separates the processing of input and target into two branches, namely the image and the pose branch. The *image branch* learns the mapping of the input image and pose into a compact discriminative space. The *pose branch* independently encodes the target pose into the same space as the compact feature of the image branch. The two feature vectors are then combined and fed to the decoder network, which learns the pixel correspondence of the target pose to generate a new image. Our network is presented as a U-Net [13] architecture with residual blocks and skip connections. To encourage the generator to produce visually appealing images, instead of optimizing the generator using pixel-wise penalty (*e.g.* $\mathcal{L}_1$) we use a masked version of the structural similarity loss.

The rest of the paper is structured as follows: Sect. 2 reviews recent advances in human view synthesis. Section 3 presents a general formulation of the problem. In Sect. 4, we present our proposed generator as well as a new weighted loss function based on a masked structural similarity loss (mask-SSIM). Experimental results are discussed in Sect. 5. Finally, in Sect. 6 we draw conclusions.

2 Background

Deep generative models for computer vision can be classified are Variational Autoencoders (VAEs) [14] or Generative Adversarial Networks (GANs) [15]. GANs are widely used for image inpainting [16], image-to-image translation [17], super-resolution [18], and cross-view image synthesis [19]. These solutions share the same *encoder-decoder* structure [20]. The idea is to map the high dimension input to much lower dimensional discriminative feature using down-sampling mappings (a series of convolutions and pooling operations). The resulting feature is then processed by a series of up-sampling (convolutions with interpolations, *e.g.* Nearest-neighbour) to get to a target space (*e.g.* the reconstruction of another image). U-Net is an encoder-decoder architecture that uses mid-level features from the encoder in the decoding module by means of skip-connections [13].

Dosovitskiy *et al.* [21] presented a generative model by learning a lower dimensional feature vector from the 3D object identity, a target view, and a transformation vector. The combined feature vector is then transformed to the desired 3D object and its segmented mask through a series of up-sampling and convolutions. Appearance Flow Network [4] learns to map the input pixel to the desired viewpoint by means of a learnable module called *Spatial Transformer* [22] which allows explicit manipulation of the feature maps. The Transformation-grounded view synthesis network (TVSN) [3] is a two-step model. The first module called Disocclusion-aware Appearance Flow Network (DOAFN) extends the Appearance Flow Network by only keeping the target pixels of the input that are presented in the output transformation using a mask that uses ground truth object coordinates and surface normals. The second network takes the output of DOAFN and refines the results by a hallucination of the missing pixels using adversarial training together with a pixel-wise reconstruction loss and perceptual loss [23]. More recently, a novel 3D synthesis model was proposed in which optical flow is first estimated, then the target view image and the target mask are synthesized on different networks. These two networks are linked using a geometry module called *perspective projection layer* [9].

Zhao *et al.* [10] formulated the problem of multi-view person synthesis using a pre-defined view as guidance in the generation. They proposed VariGANs which are a combination of variational inference and adversarial learning. The synthesized view is generated in a coarse-to-fine manner, *i.e.*, a first stage is used to produce a coarse result and a refinement step is applied to generate the high-level details. Ma *et al.* [2] proposed the PG^2 network which employs U-Net with residual blocks in the generator, enabling to generate human images as a coarse-to-fine manner from pose information. Siarohin *et al.* [1] addressed the problem of pixel-to-pixel misalignments between the input and the target image by introducing deformable skip connections in the generator and using a nearest-neighbor loss. Ma *et al.* [12] proposed to generate each of the factors (foreground, background, and pose) separately to synthesize the person image. By learning each factor in a separate branch, the model is able to sample the foreground, background, and pose separately and reconstruct a new image based on these features. An unsupervised GAN [11] for human pose generation has been proposed to avoid the need of supervision. The model introduces a loss function that only depends on the input image and the generated one. The generator is built in two-steps: firstly, a new image is generated based on the target pose, then, the rendered image is fed to a second generator that reconstructs the input image back. Si *et al.* [5] presented a multi-stage solution. First, a network learns to map the input pose to the target view pose based on a transformation vector. Then, a generator synthesizes the person to the target view conditioned on the generated pose along with the input image and pose. Finally, the background is generated in a separated network. Recently, Yang *et al.* [24] proposed a pose-guided sequence generation, from a single image, of people performing an action. A sequence of poses of the human is predicted given the action as input. The video generation is then performed as image synthesis conditioned on the predicted poses.

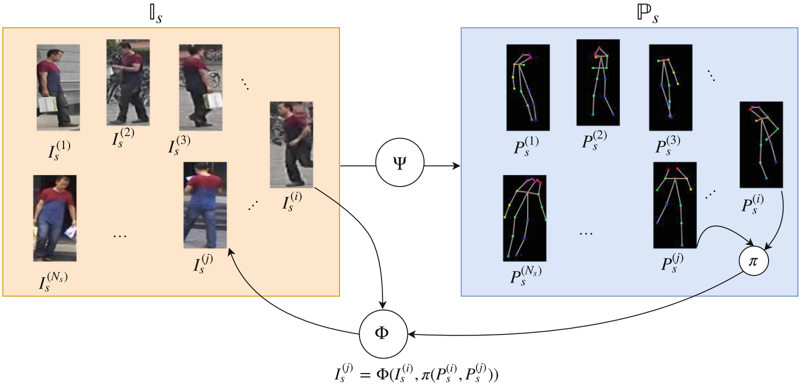

Fig. 1. The goal of pose guided human image synthesis is to generate the image of the same person given the input image $I_s^{(i)}$, its corresponding pose $P_s^{(i)}$ along with the target pose $P_s^{(j)}$ using a generator $\varPhi$.

3 Problem Formulation

We are given a set of N images taken from different view-point and poses $\mathbb{I} = \bigcup_{s=1}^{S} \mathbb{I}_s$ consisting of S persons, where $|\mathbb{I}_s| = N_s$ and $N = \sum_{s=1}^{S} N_s$. Each image $I_s^{(i)} \in \mathbb{R}^{w \times h \times c}$ is a bounding box around the person s, where w, h, and $c = 3$ are the image width, height, and channel respectively.

Let $\varPsi \colon \mathbb{I} \rightarrow \mathbb{P}$ be a mapping such that given an input image $I_s^{(i)} \in \mathbb{I}$, it estimates K-2D joints representing the body parts, *i.e.*, $P_s^{(i)} = (P_s^{(i)}[1], \ldots, P_s^{(i)}[K])$. Therefore, $\varPsi$ maps $\mathbb{I}$ to the pose set $\mathbb{P} = \bigcup_{s=1}^{S} \mathbb{P}_s \subset \mathbb{R}^{2 \times K}$. The pose guided human image synthesis is defined as follows: Given an input image $I_s^{(i)}$ and its corresponding pose $P_s^{(i)}$ along with the target pose $P_s^{(j)}$, the goal is to generate the target image $I_s^{(j)}$ using a mapping $\varPhi$ that we call generator. The generator $\varPhi$ takes the input image, and is conditioned on an operator $\pi \colon \mathbb{P} \rightarrow \mathbb{P}$ that is able to learn transform the input to the output. Finally, the target image is obtained as: $I_s^{(j)} = \varPhi(I_s^{(i)}, \pi(P_s^{(i)}, P_s^{(j)}))$ (see Fig. 1).

4 View-Disentangled Generator Model

In this section, we present our View-Disentangled Generator (VDG) model. VDG is a two-stage pipeline that first produces a coarse result from an input pair (I_a, P_a) and a target pose P_b, we refer to this stage as *Reconstruction stage*. The *Refinement stage* takes as input the generated image from the Reconstruction stage and the original input I_a, along with the target pose P_b. To train our Refinement stage model, we propose a new loss function that we extend from Structural similarity (SSIM).

4.1 Reconstruction Stage

The Reconstruction stage synthesizes a coarse representation of the target image. The encoder-decoder network is conditioned on the input image, input pose, and the target pose. We explicitly disentangle the learning of the feature between the input and the target view. The coarse image results from this stage (see Fig. 4) are obtained by training the network via a $\mathcal{L}_1$ optimization.

Given an input view image of a person I_a^i and a target view I_b^i of the same person, we build a dataset $\mathcal{D}_1$ of N pairs $\{(I_a^i, I_b^i)\}_{i=1}^N$. We define the body pose $P(I)$ of an image I to be the set of K body joints locations $P(I) = (p[1], \ldots, p[K])$. Following [1], for a given image I, we compute a heat map H consisting of the concatenation of K Gaussian heat maps centered around the j^{th} joint of the estimated pose. Therefore, for a location $p \in \mathbb{R}^2$ and the concatenation operator $\oplus$, we compute $H(I) = \bigoplus_{j=1}^{K} H_j$, where:

$$H_j(p) = \exp\Big(-\frac{\left\|p - P(I)_j\right\|_2^2}{2\sigma^2}\Big), \tag{1}$$

where $j \in \{1, \ldots, K\}$. We give to K and σ the same values as in [1]: $K = 18$ and $\sigma = 6$. We then compute the corresponding heat maps of the input pair as: $H_a = H(P(I_a))$ and $H_b = H(P(I_b))$ using Eq. 1. Finally, the supervision dataset is: $\mathcal{D}_1 = (\mathcal{X}_1, \mathcal{Y})$, where $\mathcal{X}_1 = \{(I_a^{(i)}, H_a^{(i)}, H_b^{(i)})\}_{i=1}^N$, and $\mathcal{Y} = \{I_b^{(i)}\}_{i=1}^N$. The poses are obtained using a pose estimator (*e.g.* [25]), thus, the resulting estimations are prone to errors (Fig. 2).

Processing the input image and the target pose together in the encoder network can be challenging for the network to make the correspondence if the variation between the input and the output is high. Thus, we propose to disentangle the processing of the input and the output into separate encoders. We build a dedicated encoder for the target pose denoted as Enc_{heat}, this branch will learn a compact representation of the pose, such that $\beta_b = Enc_{heat}(H_b)$. The other encoder that we note as Enc_{img} merges together the input image I_a with the input heat map H_a, it allows the encoder to learn discriminative spatial feature present in the image. The encoder Enc_{img} also learns how to combine the image space with the heat map space, this is a desirable property that will be needed in order to join the Enc_{heat} feature output, from this we

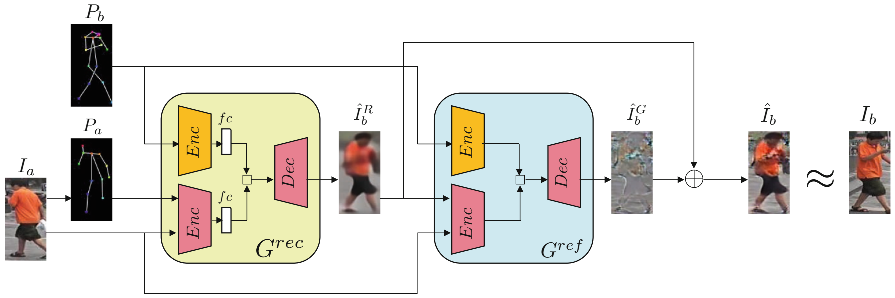

Fig. 2. Proposed VDG model.

have $\epsilon_a = Enc_{img}(I_a, H_a)$. We then combine the two feature $f = \epsilon_a \oplus \beta_b$, the target view image is reconstructed using f *i.e.*, $\hat{I}_b = Dec(f)$.

Similar to [2], the encoder-decoder network presented above (denoted as G^{rec}) follows a U-Net [13] like architecture. The encoder is built using N stacked residual blocks followed by a convolution. Each residual block is composed of two Conv-ReLU operations. In the decoder, skip connections are added between the decoder and the image branch feature maps.

Because the goal of this step is to reconstruct a coarse result, we believe that all the pixels (foreground and background) in the input image have the same importance. Therefore, to train the network we use a $\mathcal{L}_1$ loss function between the prediction $\hat{I}_b^R$ and I_b as follows:

$$\mathcal{L}_{G^{rec}} = \left\| \hat{I}_b^R - I_b \right\|_1 \tag{2}$$

4.2 Refinement Stage

We present the Refinement stage model, where the goal is to use adversarial training to reconstruct the high fidelity of the resulting images. In terms of architecture, the network has a similar disentangling encoder part except that instead, we encode the input and reconstructed image together. SSIM is also presented as loss function for the generator.

Let $\hat{I}_b^{R(i)} = G^{rec}(I_a^{(i)}, H_a^{(i)}, H_b^{(i)})$ be the output from the trained G^{rec} model for the i^{th} data sample. The dataset of the Refinement stage generator G^{ref} is built as follows: $\mathcal{D}_2 = (\mathcal{X}_2, \mathcal{Y})$, where $\mathcal{X}_2 = \{(I_a^{(i)}, \hat{I}_a^{R(i)}, H_b^{(i)})\}_{i=1}^N$, and $\mathcal{Y} = \{I_b^{(i)}\}_{i=1}^N$.

Because with the fully connected layer we lose more of the spatial information in the encoder part, we did not add this layer in the G^{ref} encoder (see Fig. 2) as suggested in [2]. Other than that, the generator G^{ref} is similar to the G^{rec} network. The *image branch* in this step is used to restore back the high-level frequency via adversarial training. The network will use the input image I_a as a reference to map the missing details from the Reconstruction stage.

The training of the model is done using a more general class of functions that was proposed in [16,26]:

$$\mathcal{L} = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{rec}\mathcal{L}_{rec}, \tag{3}$$

where $\mathcal{L}_{adv}$ and $\mathcal{L}_{rec}$ are the adversarial loss and the reconstruction loss respectively. The weighting terms are there to balance between the coarse (low-frequency) results obtained from the reconstruction loss and the sharpness (high-frequency) of the results. For the adversarial learning, instead of trying to optimize $\hat{I}_b^G$ to fit the target, we learn the residue in order for the Reconstruction stage image $\hat{I}_b^R$ to fit the target using Eq. 4:

$$\hat{I}_b = \hat{I}_b^R + \hat{I}_b^G \tag{4}$$

The network is trained using adversarial learning between our Refinement stage generator G^{ref} and the discriminator D by alternating the optimization between Eqs. 5 and 6. We use conditional discriminator [27] on the pair (I_a, I_b) for the positive samples and $(I_a, \hat{I}_b)$ for the generated images.

$$\mathcal{L}_D = \mathbb{E}_{I_a, I_b}\big[\log D(I_a, I_b)\big] + \mathbb{E}_{I_a}\big[\log(1 - D(I_a, \hat{I}_b))\big], \tag{5}$$

$$\mathcal{L}_{G^{ref}} = \mathbb{E}_{I_a}\big[\log D(I_a, \hat{I}_b)\big] + \lambda\mathcal{L}_{img}. \tag{6}$$

$\mathbb{E}$ denotes expectation, λ is a parameter that controls the influence of the reconstruction loss. Structural similarity (SSIM) [28] is a metric that assesses the quality of images. Since the goal of Refinement stage is to enhance the images from the Reconstruction stage, we propose mask-SSIM to let the model focus more on the generated person by making use of the target mask $\mathbb{M}_b$:

$$\mathcal{L}_{reconst} = \begin{cases} \left\|(\hat{I}_b^G - I_b) \odot (1 + \mathbb{M}_b)\right\|_1, & \text{for loss=}\mathcal{L}_1^{mask} \\ \mathcal{L}_{SSIM}(\hat{I}_b \odot (1 + \mathbb{M}_b), I_b \odot (1 + \mathbb{M}_b)), & \text{for loss=}\mathcal{L}_{SSIM}^{mask} \end{cases} \tag{7}$$

where $\mathcal{L}_{SSIM}$ is the SSIM loss.

Because the generator creates some visible artifacts during the adversarial training, we try to reduce them by another $\mathcal{L}_1$ term with a small weight. Inspired by the weighted reconstruction loss [29], we propose an adapted version of our model. The final loss function becomes:

$$\mathcal{L}_{img} = \alpha\mathcal{L}_{SSIM}^{mask} + (1 - \alpha)\mathcal{L}_1^{mask}. \tag{8}$$

We choose $\alpha \in \{0.9, 0.8, 0.7, 0.6, 0.5\}$ for an ablation study presented in Sect. 5 *i.e.,* the $\mathcal{L}_1$ term in $\mathcal{L}_{img}$ can go up to half of the influence. After that, the results will be blurred.

5 Experiments

This section presents our evaluation protocol as well as a quantitative and qualitative study. We provide an ablation study of our architecture and highlight some of the key factors that challenge the generation. To evaluate our model we start with an ablation study of each component of VDG (as described in Sect. 4), trained using $\mathcal{L}_1$ loss for the reconstruction in both stages. We also highlight the importance of the careful choice of the loss function and how it affects the results. The VDG model is trained using the $\mathcal{L}_1$ loss in the reconstruction term in the Refinement stage. VDG$^{mask-L_1}$ uses the target mask $\mathbb{M}_b$ in the reconstruction $\mathcal{L}_1$ loss as defined in Eq. 7. For VDG, we instead train the generator using the mask-SSIM loss. Finally, VDG$_w$ trains the Refinement stage generator using Eq. 8.

Datasets: We use Market-1501 [30] and DeepFashion [31]. Market-1501 [30] dataset contains $32,668$ images of $1,501$ identities collected from six cameras. The datasets have images of different poses, illuminations, viewpoints and backgrounds, all images are of size 128×64 and we split them into train/test sets of $12,936/19,732$. We pre-process each split by removing the images that do not contain any pose in the estimation, we then create pairs in which we have the image of the same person but with different pose. After this step we end up having $263,631$ training pairs and we randomly select $12,000$ pairs for testing.

We use the In-shop Clothes Retrieval Benchmark of DeepFashion [31] dataset, it has $52,712$ clothes images of 256×256 pixels. In total, there are $200,000$ pairs of identical clothes with different poses and/or scales. Following the procedure described for Market-1501 dataset, we build our train/test sets and we get $101,268$ training pairs and we select $8,670$ pairs for testing. To construct the train/test set on each dataset, we follow the protocol defined by Ma *et al.* [2].

Implementation Details: We train the generators G^{rec}, and G^{ref} and the discriminator D using Adam [32] optimizer with $\beta_1 = 0.5$, $\beta_2 = 0.999$, and learning rate $\epsilon = 2.10^{-5}$.

On Market-1501 (resp. DeepFashion), we set the number of residual blocks in the generators G^{rec} and G^{ref} to $N = 5$ (resp. $N = 6$). We train the model with a minibatch of size 16 (resp. 6) for 22k (resp. 40k) iterations at the Reconstruction stage and 14k (resp. 30k) iterations at the Refinement stage.

Model Evaluation: To assess the models we use the SSIM score and the Inception Score (IS) [33] which is one of the widely used metric to evaluate a generative model. IS measures the performance of the generator by evaluating the quality and the diversity of the generated images. Because of the high variation in the background of the Market-1501 dataset, Ma *et al.* [2] proposed a variant of SSIM and IS scores, which is to only apply the mask to both the original and the reconstructed image to get the scores, we report these as well.

Table 1 compares results obtained from the Reconstruction stage with the PG^2 model and our proposed VDG. To study the influence of the mask in this phase, we train PG^2 using only $\mathcal{L}_1$ without the mask which we refer to as PG^2

Table 1. Reconstruction stage results comparison with the PG^2 [2] model.

Method	Market-1501				DeepFashion	
	SSIM	IS	Mask-SSIM	Mask-IS	SSIM	IS
PG^2 [2]	.285	3.363	.801	2.798	.693	2.882
PG^2 [2] w/o mask	.290	3.356	.804	2.797	.689	2.833
VDG	.274	3.407	.799	2.733	.691	2.773

Table 2. Artifact removal evaluation by varying α over $\mathcal{L}_{img}$.

Method	Market-1501				DeepFashion	
	SSIM	IS	Mask-SSIM	Mask-IS	SSIM	IS
VDG$_w^{\alpha=0.9}$	**.266**	3.453	**.783**	3.227	.700	**3.428**
VDG$_w^{\alpha=0.8}$	.258	3.315	.779	3.201	.706	3.073
VDG$_w^{\alpha=0.7}$	.240	**3.882**	.773	**3.469**	.710	2.906
VDG$_w^{\alpha=0.6}$	.261	3.195	.773	3.258	**.711**	2.887
VDG$_w^{\alpha=0.5}$	.265	3.463	.777	3.210	.709	3.056

Table 3. Results comparison of our proposed model with other state-of-the-art solutions. ($\star$) Results were reported on different test set.

Method	Market-1501				DeepFashion	
	SSIM	IS	Mask-SSIM	Mask-IS	SSIM	IS
PG^2 [2]	.252	4.015	.771	**3.555**	.641	3.187
Def-GAN [1]	**.290**	2.990	**.798**	3.544	.665	3.420
PDIG [12] ($\star$)	.099	3.483	.614	3.491	.614	3.228
VDG^{L_1}	.224	3.733	.767	3.503	.700	3.428
VDG$^{mask-L_1}$	.238	3.933	.768	3.542	.690	3.429
VDG	.238	4.007	.775	3.354	**.708**	3.003
VDG$_w$	.266	3.453	.783	3.227	.702	**3.491**

w/o mask. We notice a slight improvement on the SSIM for PG^2 model compared to our, as for the mask, from Table 1 we do not notice any clear evidence of using the mask during this stage.

We further conducted an empirical evaluation on how varying the weighting term α affects the artifact removal due to the adversarial training and the perceptual quality as well. Table 2 reports the scores over Market-1501 and DeepFashion, we can notice that the overall best performing model is with $\alpha = 0.9$ which suggest that the $\mathcal{L}_1$ term helps to remove the noticeable artifacts without altering the perceptual quality of the results. We show some qualitative results as well in Fig. 3, for example, in the second row of DeepFashion, we can clearly

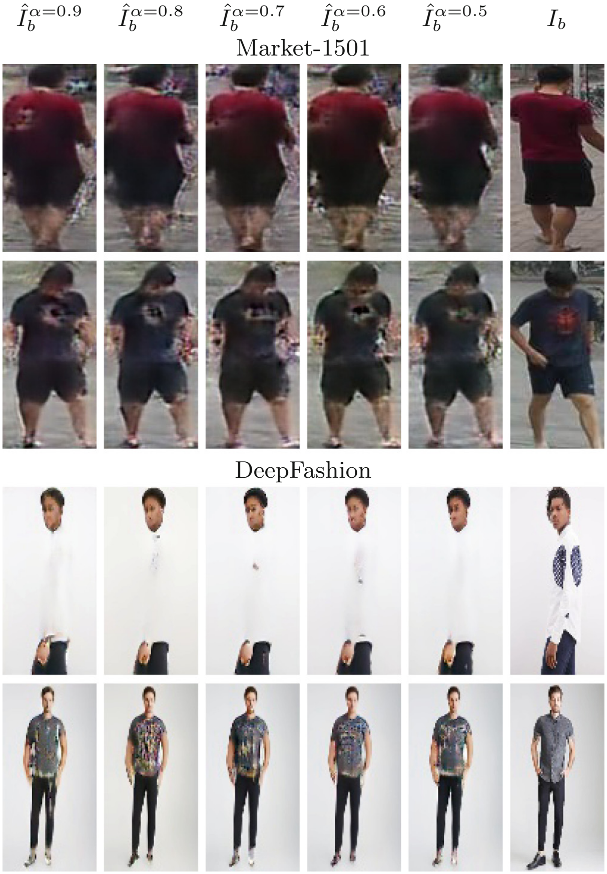

Fig. 3. Qualitative results on a various weighting scheme with the proposed loss.

see the advantage of $\text{VDG}_w^{\alpha=0.9}$ where for the other models the generation of the shirt comes with additional artifacts.

For the Refinement stage (Table 3), we can observe that the mask loss (VDG^{mask-L_1}) improves the results over only a $\mathcal{L}_1$ term (VDG^{L_1}), this is because we let the network focus more on the generation of human image. Additionally, using SSIM as a loss function helps the generator (VDG and VDG_w). We explain the equality in the SSIM scores for VDG^{mask-L_1} and VDG due to the background influence which is still challenging for both models. We further compare our model against other state-of-the-art methods. From these results, we can see the effectiveness of branching solution (VDG and Def-GAN [1]) compared to PG^2.

An important remark to make regarding the results is the inconstancy on how the evaluation measures behave. In our study, we observe that when the SSIM score improves the inception scores decrease. Similar behavior has been observed

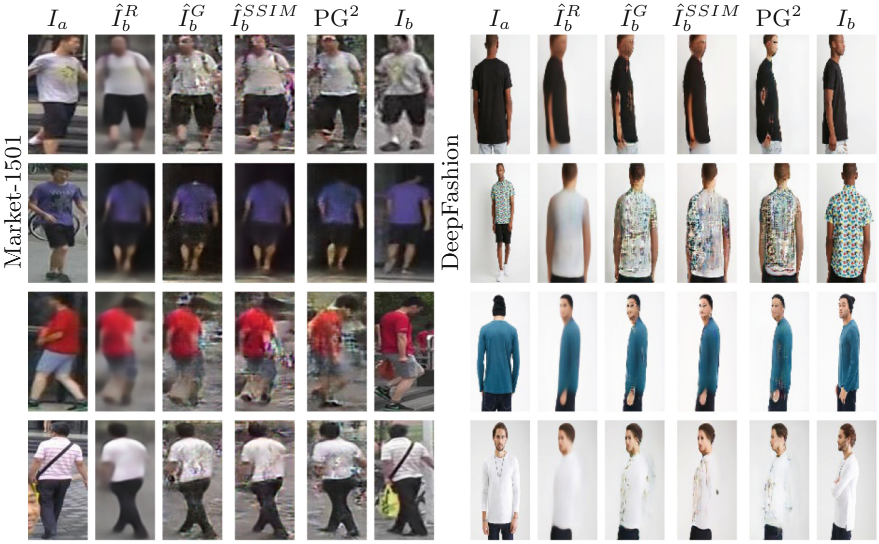

Fig. 4. Sample results obtained from Market-1501 and DeepFashion datasets.

also in [12], where the authors reported the opposite phenomena. We believe that this is still a challenging open problem on how to benchmark properly over GANs performance, we refer the reader to the work presented in [34] for more in-depth study.

Qualitatively, Fig. 4 shows the generated images from the Reconstruction stage and the final results w/o mask. Results from Reconstruction stage (second column on both datasets) reconstruct the target image based on appearance and the target pose but high-level details are absent. Refinement stage adds details by a hallucination of some missing human parts from the input (*e.g.,* faces). We notice some artifacts that are present in the final results produced by the GAN generator, which affect the general scores. Figure 5 compares our method against other state-of-the-art, our improved loss VDG$_w$ can generate images with a clear distinction between the body and the arms compared to the VDG model (see the fourth row). We also note that the $\mathcal{L}_1$ term in the proposed loss helps to remove some of the visible artifacts due to the residual term in the adversarial process (2^{nd} and 5^{th} row). We observe that PG^2 can not preserve well the color as can be seen from the third and seventh row.

Quality Assessment: In general, the generator is able to reconstruct the full body limbs. Figure 6 shows some results of our model with regards to the factors defined in Sect. 1: quality, scale, occlusion, and complexity. Interestingly, when some parts of the body are missing in the input but needed in the output our generator can hallucinate about the face and the full arm with the appropriate pants colors even with partial initial information (3^{rd} row right part). The model can handle well the scale difference (1^{st} and 2^{nd} rows). On the other hand, the

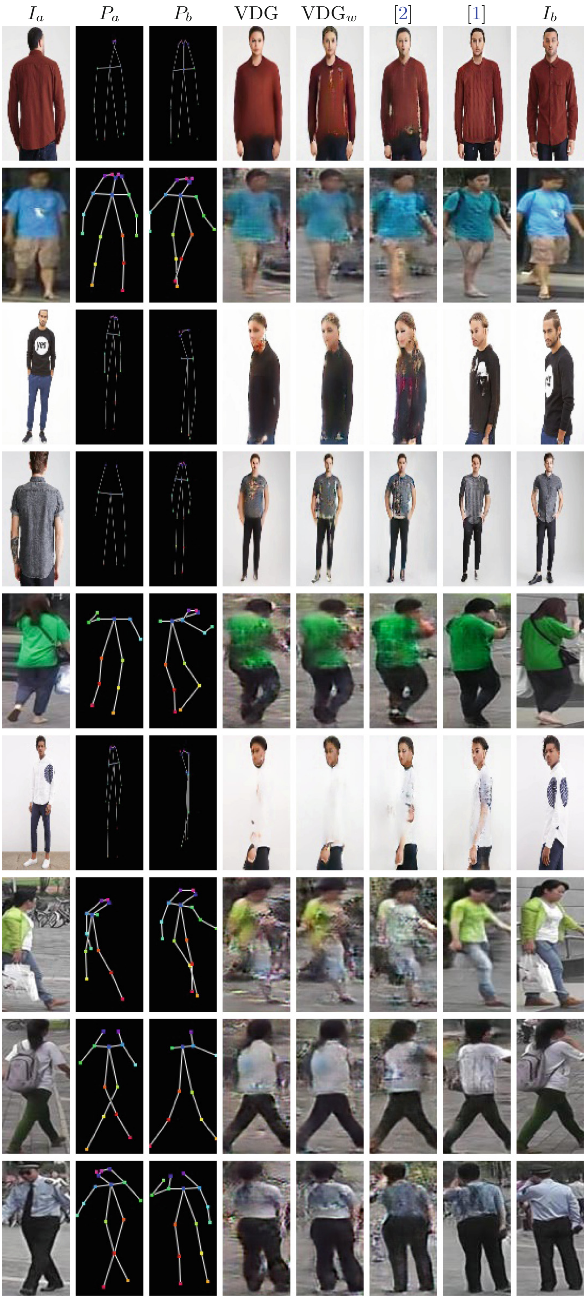

Fig. 5. Generated results using different methods on Market-1501 and DeepFashion. (Color figure online)

model produces crippled outputs on challenging cases like occlusion and the quality of the images. Also, the high variation between the input and the target background affects the produced samples.

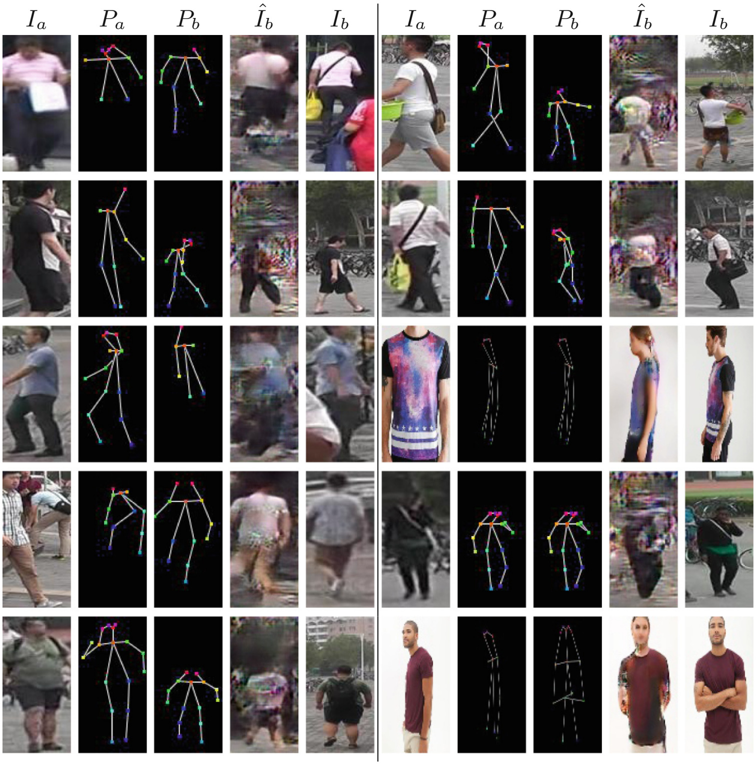

Fig. 6. Results of our model on challenging cases. (Color figure online)

6 Conclusions

We presented a two-stage deep encoder-decoder network for pose guided human image generation. We proposed a disentangled generator that explicitly separates the source image and the target pose into different branches. We further introduced mask-SSIM as a reconstruction loss function during the adversarial training, which facilitates the generator to focus on perceptually appealing outputs. The proposed model has shown competitive performance compared to the state of the art.

We observed that the background should be taken into account as it affects the quality of the generated view. Moreover, further improvements could be achieved by using sub-modules, each focusing on a specific part [5,12,35].

References

1. Siarohin, A., Sangineto, E., Lathuilière, S., Sebe, N.: Deformable GANs for pose-based human image generation. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2018
2. Ma, L., Jia, X., Sun, Q., Schiele, B., Tuytelaars, T., Van Gool, L.: Pose guided person image generation. In: Advances in Neural Information Processing Systems, NIPS, December 2017
3. Eunbyung, P., Jimei, Y., Ersin, Y., Duygu, C., Alexander, C.B.: Transformation-grounded image generation network for novel 3D view synthesis. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, July 2017
4. Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.A.: View synthesis by appearance flow. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 286–301. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_18
5. Chenyang, S., Wei, W., Liang, W., Tieniu, T.: Multistage adversarial losses for pose-based human image synthesis. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2018
6. Kholgade, N., Simon, T., Efros, A., Sheikh, Y.: 3D object manipulation in a single photograph using stock 3D models. ACM Trans. Comput. Graph. **33**, 127 (2014)
7. Zheng, Y., Chen, X., Cheng, M.M., Zhou, K., Hu, S.M., Mitra, N.J.: Interactive images: cuboid proxies for smart image manipulation. ACM Trans. Graph. **31**, 99:1–99:11 (2012)
8. Yan, X., Yang, J., Yumer, E., Guo, Y., Lee, H.: Perspective transformer nets: learning single-view 3D object reconstruction without 3D supervision. In: Advances in Neural Information Processing Systems, NIPS, December 2016
9. Zhu, H., Su, H., Wang, P., Cao, X., Yang, R.: View extrapolation of human body from a single image. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2018
10. Zhao, B., Wu, X., Cheng, Z., Liu, H., Feng, J.: Multi-view image generation from a single-view. Volume abs/1704.04886 (2017)
11. Pumarola, A., Agudo, A., Sanfeliu, A., Moreno-Noguer, F.: Unsupervised person image synthesis in arbitrary poses. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2018
12. Ma, L., Sun, Q., Georgoulis, S., Van Gool, L., Schiele, B., Fritz, M.: Disentangled person image generation. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2018
13. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
14. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: The International Conference on Learning Representations, ICLR, April 2014
15. Goodfellow, I., et al.: Generative adversarial nets. In: Advances in Neural Information Processing Systems, NIPS, December 2014
16. Pathak, D., Krähenbühl, P., Donahue, J., Darrell, T., Efros, A.: Context encoders: feature learning by inpainting. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2016
17. Isola, P., Zhu, J., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, July 2017

18. Ledig, C., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE International Conference on Computer Vision, ICCV, October 2017
19. Krishna, R., Ali, B.: Cross-view image synthesis using conditional GANs. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2018
20. Bengio, Y., Courville, A., Vincent, P.: Representation learning: a review and new perspectives. IEEE Trans. Pattern Anal. Mach. Intell. (PAMI) **35**, 1798–1828 (2013)
21. Dosovitskiy, A., Springenberg, J.T., Tatarchenko, M., Brox, T.: Learning to generate chairs, tables and cars with convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. (PAMI) **39**, 692–705 (2017)
22. Jaderberg, M., Simonyan, K., Zisserman, A., Kavukcuoglu, K.: Spatial transformer networks. In: Advances in Neural Information Processing Systems, NIPS, December 2015
23. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 694–711. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_43
24. Yang, C., Wang, Z., Zhu, X., Huang, C., Shi, J., Lin, D.: Pose guided human video generation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11214, pp. 204–219. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01249-6_13
25. Cao, Z., Simon, T., Wei, S.E., Sheikh, Y.: Realtime multi-person 2D pose estimation using part affinity fields. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, July 2017
26. Dosovitskiy, A., Brox, T.: Generating images with perceptual similarity metrics based on deep networks. In: Advances in Neural Information Processing Systems, NIPS, December 2016
27. Mirza, M., Osindero, S.: Conditional generative adversarial nets. Volume abs/1411.1784 (2014)
28. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. (TIP) **13**, 600–612 (2004)
29. Zhao, H., Gallo, O., Frosio, I., Kautz, J.: Loss functions for image restoration with neural networks. IEEE Trans. Comput. Imag. **3**, 47–57 (2017)
30. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: a benchmark. In: IEEE International Conference on Computer Vision, ICCV, December 2015
31. Liu, Z., Luo, P., Qiu, S., Wang, X., Tang, X.: DeepFashion: powering robust clothes recognition and retrieval with rich annotations. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2016
32. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. In: International Conference on Learning Representations, ICLR, May 2015
33. Salimans, T., et al.: Improved techniques for training GANs. In: Advances in Neural Information Processing Systems, NIPS, December 2016
34. Borji, A.: Pros and cons of GAN evaluation measures. Volume abs/1802.03446 (2018)
35. Guha, B., Amy, Z., Adrian, V.D., Fredo, D., John, G.: Synthesizing images of humans in unseen poses. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR, June 2018