Attention-based Fusion for Multi-source Human Image Generation

Stéphane Lathuilière[1], Enver Sangineto[2], Aliaksandr Siarohin[2] and Nicu Sebe[2,3]
[1]LTCI, Télécom Paris, Institut Polytechnique de Paris, France,
[2] DISI, University of Trento, Italy, [3]Huawei Technologies Ireland, Dublin, Ireland

Abstract

We present a generalization of the person-image generation task, in which a human image is generated conditioned on a target pose and a set $\boldsymbol{X}$ *of source appearance images. In this way, we can exploit multiple, possibly complementary images of the same person which are usually available at training and at testing time. The solution we propose is mainly based on a local attention mechanism which selects relevant information from different source image regions, avoiding the necessity to build specific generators for each specific cardinality of* $\boldsymbol{X}$*. The empirical evaluation of our method shows the practical interest of addressing the person-image generation problem in a multi-source setting.*

1. Introduction

The person image generation task, as proposed by Ma et al. [19], consists in generating "person images in arbitrary poses, based on an image of that person and a novel pose". This task has recently attracted a lot of interest in the community because of different potential applications, such as computer-graphics based manipulations [37] or data augmentation for training person re-identification [45, 16] or human pose estimation [5] systems. Previous work on this field [19, 15, 43, 29, 26, 3, 25] assume that the generation task is conditioned on two variables: the appearance image of a person (we call this variable the *source* image) and a *target* pose, automatically extracted from a different image of the same person using a Human Pose Estimator (HPE).

Using person-specific abundant data the quality of the generated images can be potentially improved. For instance, a training dataset specific to each target person can be recorded [6]. Another solution is to build a full-3D model of the target person [17]. However, these approaches lack of flexibility and need an expensive data-collection.

In this work we propose a different direction which relies on a few, variable number of source images (e.g., from 2 to 10). We call the corresponding task *multi-source human image generation*. As far as we know, no previous work has investigated this direction yet. The reason for which we believe this generalization of the person-image generation task is interesting is that multiple source images, when available, can provide richer appearance information. This data redundancy can possibly be exploited by the generator in order to compensate for partial occlusions, self-occlusions or noise in the source images. More formally, we define our multi-source human image generation task as follows. We assume that a set of M ($M \geq 1$) source images $\boldsymbol{X} = \{x^i\}_{i=1..M}$ is given and that these images depict the same person with the same overall appearance (e.g., the same clothes, haircut, etc.). Besides, a unique target body pose p^τ is provided, typically extracted from a target image not contained in $\boldsymbol{X}$. The multi-source human image generation task consists in generating a new image $\hat{x}$ with an appearance similar to the general appearance pattern represented in $\boldsymbol{X}$ but in the pose p^τ (see Fig. 1). Note that M is not a-priori fixed, and we believe this task characteristics are important for practical applications, in which the same dataset can contain multiple-source images of the same person but with unknown and variable cardinalities.

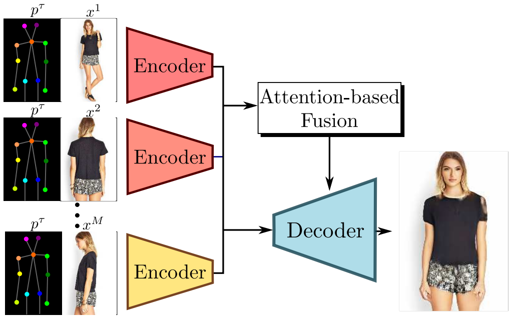


Figure 1: Multi-source Human Image Generation: an image of a person in a novel pose is generated from *a set* of images of the same person.

Most of previous methods on single-source human image generation [26, 29, 15, 19, 37, 43, 9, 25, 16] are based on variants of the U-Net architecture generator proposed by Isola et al. [13]. A common, general idea in these methods is that the conditioning information (e.g., the source image and/or the target pose) is transformed into the desired synthetic image using the U-Net *skip connections*, which shuttle information between those layers in the encoder and in the decoder having a corresponding resolution (see Sec. 3). However, when the cardinality M of the source images is not fixed a priori, as in our proposed task, a "plain" U-Net architecture cannot be used, being the number of input neurons a-priori fixed. For this reason, we propose to modify the U-Net generator introducing an *attention* mechanism. Attention is widely used to represent a variable-length input into a deep network [2, 39, 36, 35, 10, 34] and, without loss of generality, it can be thought of as a mechanism in which multiple-input representations are averaged (i.e., summed) using some saliency criterion emphasizing the importance of specific representations with respect to the others. In this paper we propose to use attention in order to let the generator decide which specific image locations of each source image are the most trustable and informative at different convolutional layer resolutions. Specifically, we keep the standard encoder-decoder general partition typical of the U-Net (see Sec. 3) but we propose three novelties[1]. First, we introduce an *attention-based decoder* (A) which fuses the feature representations of each source. Second, we encode the target pose and each source image with an *encoder* (E) which processes each source image x_i *independently* of the others and E locally deforms each x_i performing a target-pose driven geometric "normalization" of x_i. Once normalized, the source images can be compared to each other in A, assigning location and source-specific saliency weights which are used for fusion. Finally, we use a *multi-source* adversarial loss $\mathcal{L}_{M-GAN}$ that employs a single discriminator to handle any arbitrary number of source images.

2. Related work

Most of the image generation approaches are based either on Variational Autoencoders (VAEs) [14] or on Generative Adversarial Networks (GANs) [11]. GANs have been extended to conditional GANs [31], where the image generation depends on some input variable. For instance, in [13], an input image x is "translated" into a different representation y using a U-Net generator.

The person generation task (Sec. 1) is a specific case of a conditioned generation process, where the conditioning variables are the source and the target images. Most of the previous works use conditional GANs and a U-Net architecture. For instance, Ma et al. [19] propose a two-step training procedure: pose generation and texture refinement, both obtained using a U-Net architecture. Recently, this work has been extended in [20] by learning disentangled representations of the pose, the foreground and the background. Following [19], several methods for pose-guided image generation have been recently proposed [15, 43, 26, 29, 3, 25]. All these approaches are based on the U-Net. However, the original U-Net, having a fixed-number of input images, cannot be directly used for the multi-source image generation as defined in Sec. 1. Siarohin et al. [29] modify the U-Net using *deformable skip connections* which align the input image features with the target pose. In this work we use an *encoder* similar to their proposal in order to align the source images with the target pose, but we introduce a *pose stream* which compares the similarity between the source and the target pose. Moreover, similarly to the aforementioned works, also [29] is single-source and uses a "standard" U-Net *decoder* [13].

Other works on image-generation rely on a strong supervision during training or testing. For instance, [21, 41] use a dense-pose estimator [12] trained using image-to-surface correspondences [12]. Dong et al. [8] use an externally trained model for image segmentation in order to improve the generation process. Zanfir et al. [42] estimate the human 3D-pose using meshes and identify the mesh regions that can be transferred directly from the input image mesh to the target mesh. However, these methods cannot be directly compared with most of the other works, including ours, which rely only on a sparse keypoint detection. Hard data-collection constraints are used also in [6], where a person and a background specific model are learned for video generation. This approach requires that the target person moves for several minutes covering all the possible poses and that a new model is trained specifically for each target person. Similarly, Liu et al. [17] compute the 3D human model by combining several minutes of video. In contrast with these works, our approach is based on fusing only a few source images in random poses and in variable number, which we believe is important because it makes it possible to exploit existing datasets where multiple images are available for the same person. Moreover, our network does not need to be trained for each specific person.

Sun et al. [32] propose a multi-source image generation approach whose goal is to generate a new image according to a target-camera position. Note that this task is different from what we address in this paper (Sec. 1), since a human pose describes an *articulated* object by means of a set of joint locations, while a camera position describes a viewpoint change but does not deal with source-to-target object *deformations*. Specifically, Sun et al. [32] represent the camera pose with either a discrete label (e.g., *left*, *right*,etc.) or a 6DoF vector and then they generate a pixel-flow which estimates the "movement" of each source-image pixel. Multiple images are integrated using a Conv-LSTM [24] and confidence maps. Most of the reported results concern 3D synthetic rigid objects, while few real scenes are

[1] Code available at `https://github.com/Stephlat/Multi-source-Human-Image-Generation`.

also used but only with limited viewpoint changes.

3. Attention-based U-Net

3.1. Overview

We first introduce some notation and provide a general overview of the proposed method. Referring to the multi-source human image generation task defined in Sec. 1, we assume a training set $\mathcal{X} = \{\mathcal{X}_n\}_{n=1..N}$ is given, being each sample $\mathcal{X}_n = (\boldsymbol{X}_n, x_n^\tau)$, where $\boldsymbol{X}_n = \{x_n^i\}_{i=1..M_n}$ is a set of M_n source images of the same person sharing a common appearance and x_n^τ is the target image. Every sample image has the same size $H \times W$. Note that the source-set size M_n is variable and depends on the person identity n. Given an image x depicting a person, we represent the body-pose as a set of 2D keypoints $P(x) = (\mathbf{p}_1, ..., \mathbf{p}_K)$, where each $\mathbf{p}_k$ is the pixel location of a body joint in x. The body pose can be estimated from an image using an external HPE. The target pose is denoted by $p_n^\tau = P(x_n^\tau)$.

Our method is based on a conditional GAN approach, where the generator G follows a general U-Net architecture [13] composed of an encoder and a decoder. A U-Net encoder is a sequence of convolutional and pooling layers, which progressively decrease the spatial resolution of the input representation. As a consequence, a specific activation in a given encoder layer has a receptive field progressively increasing with the layer depth, so gradually encoding "contextual" information. Vice versa, the decoder is composed of up-convolution layers, and, importantly, each decoder layer is connected to the corresponding layer in the encoder by means of *skip connections*, that concatenate the encoder-layer feature maps with the decoder-layer feature maps [13]. Finally, Isola et al. [13] use a conditional discriminator D in order to discriminate between real and fake "image transformations".

We modify the aforementioned framework in three main aspects. First, we use M_n replicas of the same encoder E in order to encode the M_n *geometrically normalized* source images together with the target pose. Second, we propose an *attention-based decoder* A that fuses the feature maps provided by the encoders. Finally, we propose a *multi-source* adversarial loss $\mathcal{L}_{M-GAN}$.

Fig. 2 shows the architecture of G. Given a set $\boldsymbol{X}_n$ of M_n source images, E encodes each source image $x_n^i \in \boldsymbol{X}_n$ together with the target pose. Similarly to the standard U-Net, for a given source image $\boldsymbol{x}_n^i$, each encoder outputs R feature maps $\boldsymbol{\xi}_r^i \in \mathbb{R}^{H_r \times W_r \times C_r^E}$, $r \in [1..R]$ for R different-resolution blocks. Each $\boldsymbol{\xi}_r^i$ is aligned with the target pose (Sec 3.3). This alignment acts as a geometric "normalization" of each $\boldsymbol{\xi}_r^i$ with respect to p_n^τ and makes it possible to compare $\boldsymbol{\xi}_r^i$ with $\boldsymbol{\xi}_r^j$ $(i \neq j)$. Finally, each tensor $\boldsymbol{\xi}_r^i$ jointly represents pose and appearance information at resolution r.

3.2. The Attention-based Decoder

A is composed of R blocks. Similarly to the standard U-Net, the spatial resolution increases symmetrically with respect to the blocks in E. Therefore, to highlight this symmetry, the decoder blocks are indexed from R to 1. In the current r-th block, the image $\hat{x}$ which is going to be generated is represented by a tensor $\boldsymbol{\phi}_r$. This representation is progressively refined in the subsequent blocks using an attention-based fusion of $\{\boldsymbol{\xi}_r^i\}_{i=1,...,M_n}$. We call $\boldsymbol{\phi}_r$ the *latent representation* of $\hat{x}$ at resolution r, and $\boldsymbol{\phi}_r$ is recursively defined starting from $r = R$ till $r = 1$ as follows:

The initial latent representation $\boldsymbol{\phi}_R$ is obtained by averaging the output tensors of the *last* layer of E (Fig. 2):

$$\boldsymbol{\phi}_R = \frac{1}{M_n} \sum_{i=1}^{M_n} \boldsymbol{\xi}_R^i \tag{1}$$

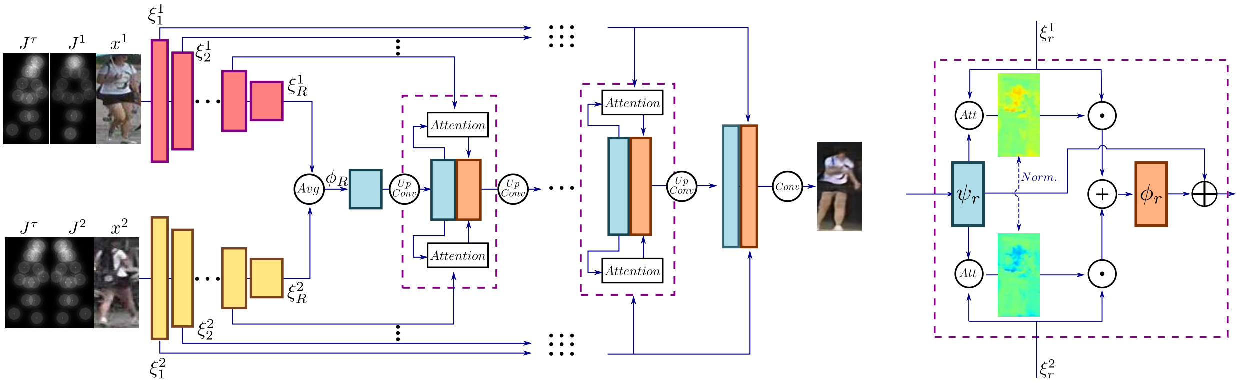
Note that each spatial position in $\boldsymbol{\phi}_R$ corresponds to a large receptive field in the original image resolution which, if R is sufficiently large, may include the whole initial image. As a consequence, we can think of $\boldsymbol{\phi}_R$ as encoding general contextual information on $(\boldsymbol{X}_n, p_n^\tau)$.

For each subsequent block $r \in [R-1, ..., 1]$, $\boldsymbol{\phi}_r$ is computed as follows. Given $\boldsymbol{\phi}_{r+1} \in \mathbb{R}^{H_{r+1} \times W_{r+1} \times C_{r+1}^E}$, we first perform a 2×2 up-sampling on $\boldsymbol{\phi}_{r+1}$ followed by a convolution layer in order to obtain a tensor $\boldsymbol{\psi}_r \in \mathbb{R}^{H_r \times W_r \times C_r^D}$. $\boldsymbol{\psi}_r$ is then fed to an attention mechanism in order to estimate how the different tensors $\boldsymbol{\xi}_r^i$ should be fused into a single final tensor F_r:

$$F_r = \sum_{i=1}^{M_n} Att(\boldsymbol{\psi}_r, \boldsymbol{\xi}_r^i) \odot \boldsymbol{\xi}_r^i, \tag{2}$$

where $\odot$ denotes the element-wise product and $Att(\cdot, \cdot) \in [0,1]^{H_r \times W_r \times C_r^E}$ is the proposed attention module.

In order to reduce the number of weights involved in computing Eq. (2), we factorize $Att(\boldsymbol{\psi}_r, \boldsymbol{\xi}_r^i)$ using a spatial-attention $g(\boldsymbol{\psi}_r, \boldsymbol{\xi}_r^i) \in [0,1]^{H_r \times W_r}$ (which is channel independent) and a channel-attention vector $f(\boldsymbol{\psi}_r, \boldsymbol{\xi}_r^i) \in [0,1]^{C_r^E}$ (which is spatial independent). Specifically, at each spatial coordinate (h, w), $g()$ compares the current latent representation $\boldsymbol{\psi}_r[h,w] \in \mathbb{R}^{C_r^D}$ with $\boldsymbol{\xi}_r^i[h,w] \in \mathbb{R}^{C_r^E}$ and assigns a saliency weight to $\boldsymbol{\xi}_r^i[h,w]$ which represents how significant/trustable is $\boldsymbol{\xi}_r^i[h,w]$ with respect to $\boldsymbol{\psi}_r[h,w]$. The function $g()$ is implemented by taking the concatenation of $\boldsymbol{\psi}_r$ and $\boldsymbol{\xi}_r^i$ as input and then using a $1 \times 1 \times (C_r^D + C_r^E)$ convolution layer. Similarly, $f()$ is implemented by means of global-average-pooling on the concatenation of $\boldsymbol{\psi}_r$ and $\boldsymbol{\xi}_r^i$ followed by two fully-connected



(a) A schematic representation of the proposed attention decoder architecture

(b) Zoom on the attention module

Figure 2: Illustration of the proposed Attention U-Net. For the sake of clarity, in this figure, we consider the case in which we use only two conditioning images ($M_n = 2$). The colored rectangles represent the feature maps. The attention module (dashed purple rectangles) in the figure (a) are detailed in figure (b). The dashed double arrows denote normalization across attention maps, $\odot$ denotes the element-wise product and $\oplus$ denotes the concatenation along the channel axis.

layers. We employ sigmoid activations on both g and f. Combining together $g()$ and $f()$, we obtain:

$$A_r^i[h, w, c] = g(\boldsymbol{\psi}_r, \boldsymbol{\xi}_r^i)[h, w] \cdot f(\boldsymbol{\psi}_r, \boldsymbol{\xi}_r^i)[c]. \tag{3}$$

Importantly, A_r^i is not spatially or channel normalized. This because a normalization would enforce that, overall, each source image is used in the same proportion. Conversely, without normalization, given, for instance, a non-informative source x_n^i (e.g., x_n^i completely black), the attention module can correspondingly produce a null saliency tensor A_r^i. Nevertheless, the final attention tensor $Att()$ in Eq. (2) is normalized in order to assign a *relative* importance to each source:

$$Att(\boldsymbol{\psi}_r, \boldsymbol{\xi}_r^i)[h, w, c] = \frac{A_r^i[h, w, c]}{\sum_{j=1}^{M_n} A_r^j[h, w, c]}. \tag{4}$$

Finally, the new latent representation at resolution r is obtained by concatenating $\boldsymbol{\psi}_r$ with F_r:

$$\boldsymbol{\phi}_r = \boldsymbol{\psi}_r \oplus F_r, \tag{5}$$

where $\oplus$ is the tensor concatenation along the channel axis.

3.3. The Pose-based Encoder

Rather than using a generic convolutional encoder as in [13], we use a task-specific encoder specifically designed to work synergistically with our proposed attention model. Our pose-based encoder E is similar to the encoder proposed in [26, 29] but it also contains a dedicated stream which is used to compare the source and the target pose. In more detail, E is composed of two streams (see Fig. 3). The first stream, referred to as *pose stream*, is used to represent pose information and to compare each other the target pose with the pose of the person in the source image.

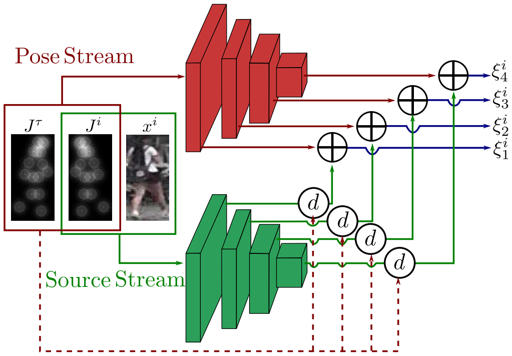


Figure 3: The Pose-based encoder. For simplicity, we show only 4 blocks ($R = 4$). Each parallelepiped represents the feature maps obtained after convolution and max-pooling. The d circles denote deformations.

Specifically, the target pose p^τ is represented using a tensor J^τ composed of K heatmaps $J^{\tau,k} \in [0, 1]^{H \times W}$. For each joint $\mathbf{p}_k^\tau \in p^\tau$, a heatmap $J^{\tau,k}$ is computed using a Gaussian kernel centered in $\mathbf{p}_k$ [26, 29]. Similarly, given $x_n^i \in \boldsymbol{X}_n$, we extract the pose $P(x_n^i)$ using [5] and we describe it using a tensor J_n^i. The tensors J_n^τ and J_n^i are concatenated and input to the pose stream, which is composed of a sequence of convolutional and pooling layers. The purpose of the pose stream is twofold. First, it provides the target pose to the decoder. Second, it encodes the similarity between the i-th source pose and the target pose. This similarity is of a crucial importance for our attention mechanism to work (Sec. 3.2) since a source image with a pose similar to the target pose is likely more trustable in order to transfer appearance information to the final generated image. For instance, a leg in x_n^i with a pose closer to p_n^τ than the corresponding leg in x_n^j, should be most likely preferred for encoding the leg appearance.

The second stream, called *source stream*, takes as input the concatenation of the RGB image x_n^i and its pose representation J_n^i. J_n^i is provided as input to the source stream in order to guide the source-stream convolutional layers in extracting relevant information which may depend on the joint locations. The output of each convolutional layer of the source stream is a tensor (green blocks in Fig. 3). This tensor is then *deformed* according to the difference between $P(x_n^i)$ and p_n^τ (the d circles in Fig. 3). Specifically, we use body part-based affine deformations as in [26, 29] to locally deform the source-stream feature maps at each given layer and then concatenate the obtained tensor with the corresponding-layer pose-stream tensor. In this way we get a final tensor $\boldsymbol{\xi}_r^i$ for each of the R different layers in E ($1 \leq r \leq R$). Each $\boldsymbol{\xi}_r^i$ is a representation of $(P(x_n^i), x_n^i)$ *aligned* with p_n^τ and it is obtained *independently* of $x_n^j \in \boldsymbol{X}_n, j \neq i$.

Given a set $\boldsymbol{X}_n$ of M_n source images, we apply M_n replicas of the E encoder to each $x_n^i \in \boldsymbol{X}_n$ producing the set of output tensors $\mathcal{E}_n = \{\boldsymbol{\xi}_r^i\}_{i=1,\ldots,M_n, r=1,\ldots R}$ that are input to the decoder described in Sec.3.2.

3.4. Training

We train the whole network in an end-to-end fashion combining a reconstruction loss with an adversarial loss. For the reconstruction loss, we use the *nearest-neighbour* loss $\mathcal{L}_{NN}(G)$ introduced in [26, 29] which exploits the convolutional maps of an external network (VGG-19 [30], trained on ImageNet [7]) at the original image resolution in order to compare each location of the generated image $\hat{x}$ with a local neighbourhood of the ground-truth image x^τ. This reconstruction loss is more robust to small spatial misalignments between $\hat{x}$ and x^τ than other common losses as the L_1 loss.

On the other hand, in our multi-source problem, the employed adversarial loss has to handle a varying number of sources. We use a single-source discriminator conditioned on only one source image x_n^i [13]. More precisely, we use M_n discriminators D that share their parameters and independently process each x_n^i. Each D takes as input the concatenation of four tensors: $x, J_n^\tau, x_n^i, J_n^i$, where x is either the ground truth real image x_n^τ or the generated image $\hat{x}$. Differently from other multi-source losses [40, 1, 22], we employ a conditional discriminator in order to exploit the information contained in the source image and the pose heatmaps. The GAN loss for the i^{th} source image is defined as:

$$\mathcal{L}_{GAN}^i(G, D) = \mathbb{E}_{(x_n^i, x_n^\tau) \in \mathcal{X}}[\log D(x_n^\tau, J_n^\tau, x_n^i, J_n^i)] + \mathbb{E}_{(x_n^i, x_n^\tau) \in \mathcal{X}, z \in \mathcal{Z}}[\log(1 - D(\hat{x}, J_n^\tau, x_n^i, J_n^i))], \tag{6}$$

where $\hat{x} = G(z, \boldsymbol{X}_n, p_n^\tau)$ and, with a slight abuse of notation, $\mathbb{E}_{(x_n^i, x_n^\tau) \in \mathcal{X}}[\cdot]$ means the expectation computed over pairs of single-source and target image extracted at random from the training set $\mathcal{X}$. Using Eq. (6), the *multi-source* adversarial loss ($\mathcal{L}_{M-GAN}$) is defined as:

$$\mathcal{L}_{M-GAN}(G, D) = \min_G \max_D \sum_{i=1}^{M_n} \mathcal{L}_{GAN}^i(G, D). \tag{7}$$

Putting all together, the final training loss is given by:

$$G^* = \arg \min_G \max_D \mathcal{L}_{M-GAN}(G, D) + \lambda \mathcal{L}_{NN}(G), \tag{8}$$

where the λ weight is set to 0.01 in all our experiments.

4. Experiments

In this section we evaluate our method both qualitatively and quantitatively adopting the evaluation protocol proposed by Ma et al. [19]. We train G and D for 60k iterations, using the Adam optimizer (learning rate: $2 * 10^{-4}$, $\beta_1 = 0.5$, $\beta_2 = 0.999$). We use instance normalization [33] as recommended in [13]. The networks used for E and D have the same convolutional-layer dimensions and normalization parameters used in [29]. Also the up-convolutional layers of A have the same dimensions of the corresponding decoder used in [29]. Finally, the number of the hidden-layer neurons used to implement $f()$ (Sec. 3.2) is $\frac{C_r^D + C_r^E}{4}$. For a fair comparison with single-source person generation methods [19, 20, 9, 26, 29], we adopt the HPE proposed in [5].

Even if there is no constraint on the cardinality of the source images M_n, in order to simplify the implementation, we train and test our networks using different steps, each step having M_n fixed for all $\mathcal{X}_n$ in $\mathcal{X}$. Specifically, we initially train E, A and D with $M_n = 2$. Then, we fine-tune the model with the desired M_n value, except for single-source experiments where $M_n = 1$ (see Sec. 4.4).

4.1. Datasets

The person re-identification Market-1501 dataset [44] is composed of 32,668 images of 1,501 different persons captured from 6 surveillance cameras. This dataset is challenging because of the high diversity in pose, background, viewpoint and illumination, and because of the low-resolution images (128×64). To train our model, we need tuples of images of the same person in different poses. As this dataset is relatively noisy, we follow the preprocessing described in [29]. The images where no human body is detected using the HPE are removed. Other methods [19, 20, 9, 29] generate all the possible pairs for each identity. However, in

		Market-1501				DeepFashion	
Model	M	*SSIM*	*IS*	*mask-SSIM*	*mask-IS*	*SSIM*	*IS*
Ma et al. [19]	1	0.253	3.460	0.792	3.435	0.762	3.090
Ma et al. [20]	1	0.099	3.483	0.614	3.491	0.614	3.228
Esser et al. [9]	1	**0.353**	3.214	0.787	3.249	**0.786**	3.087
Siarohin et al. [26, 29]	1	0.290	3.185	0.805	3.502	0.756	**3.439**
Ours	1	0.270 ± 0.09	3.251 ± 0.09	0.771 ± 0.07	3.614 ± 0.08	0.757 ± 0.07	3.420 ± 0.06
Ours	2	0.285 ± 0.09	3.474 ± 0.09	0.778 ± 0.06	3.634 ± 0.08	0.769 ± 0.07	**3.421 ± 0.06**
Ours	3	0.291 ± 0.06	3.442 ± 0.09	0.783 ± 0.06	3.739 ± 0.08	0.774 ± 0.07	3.400 ± 0.03
Ours	5	0.306 ± 0.09	3.444 ± 0.05	0.788 ± 0.06	**3.814 ± 0.07**	0.774 ± 0.06	3.416 ± 0.06
Ours	7	0.320 ± 0.09	**3.613 ± 0.05**	0.801 ± 0.06	3.567 ± 0.06	-	-
Ours	10	0.326 ± 0.09	3.442 ± 0.07	**0.806 ± 0.06**	3.514 ± 0.04	-	-

Table 1: Comparison with the state of the art on the Market-1501 and the DeepFashion datasets.

our approach, since we consider tuples of size $M + 1$ (M sources and 1 target image), considering all the possible tuples is computationally infeasible. In addition, Market-1501 suffers from a high person-identity imbalance and computing all the possible tuples, would exponentially increase this imbalance. Hence, we generate tuples randomly in such a way that we obtain the same identity repartition than it is obtained when sampling all the possible pairs. In addition, this solution also allows for a fair comparison with single-source methods which sample based on pairs. Eventually, we get 263K tuples for training. For testing, following [19], we randomly select 12K tuples without person is in common between the training and the test split.

The DeepFashion dataset (*In-shop Clothes Retrieval Benchmark*) [18] consists of 52,712 clothes images with a resolution of 256×256 pixels. For each outfit, we dispose of about 5 images with different viewpoints and poses. Thus, we only perform experiments using up to $M_n = 5$ sources. Following the training/test split adopted in [19], we create tuples of images following the same protocol as for the market-1501 dataset. After removing the images where the HPE does not detect any human body, we finally collect about 89K tuples for training and 12K tuples for testing.

4.2. Metrics

Evaluation metrics in the context of generation tasks is a problem in itself. In our experiments we adopt the evaluation metrics proposed in [19] which is used by most of the single-source methods. Specifically, we use: Structural Similarity (*SSIM*) [38], Inception Score (*IS*) [23] and their corresponding masked versions *mask-SSIM* and *mask-IS* [19]. The masked versions of the metrics are obtained by masking-out the image background. The motivation behind the use of masked metrics is that no background information is given to the network, and therefore, the network cannot guess the correct background of the target image. For a fair comparison, we adopt the masks as defined in [19].

It is worth noting that the SSIM-based metrics compare the generated image with the ground-truth. Thus, they measure how well the model transfers the appearance of the person from the source image. Conversely, IS-based metrics evaluate the distribution of generated images, jointly assessing the degree of realism and diversity of the generated outcomes, but do not take into account any similarity with the conditioning variables. These two metrics are each other complementary [4] and should be interpreted jointly.

4.3. Comparison with previous work

Quantitative comparison. In Tab. 1 we show a quantitative comparison with state-of-the-art single-source methods. Note that, except from [20], none of the compared methods, including ours, is conditioned on background information. On the other hand, the mask-based metrics focus on only the region of interest (i.e., the foreground person) and they are not biased by the randomly generated background. For these reasons, we believe the mask-based metrics are the most informative ones. However, on the DeepFashion dataset, following [20], we do not report the masked values since the background is uniform in most of the images. On both datasets, we observe that the SSIM and masked-SSIM increase when we input more images to our model. This confirms the idea that multi-source image generation is an effective direction to improve the generation quality. Furthermore, it illustrates that the proposed model is able to combine the information provided by the different source images. Interestingly, our method reaches high SSIM scores while keeping high IS values, thus showing that it is able to transfer better the appearance without loosing image quality and diversity.

Concerning the comparison with the state of the art, our method reports the highest performance according to both the mask-SSIM and the mask-IS metrics on the Market-1501 dataset when we use 10 source images. When we employ fewer images, only Siarohin et al [29] obtain better masked-SSIM but at the cost of a significantly lower IS. Similarly, we observe that [9] achieves a really high SSIM

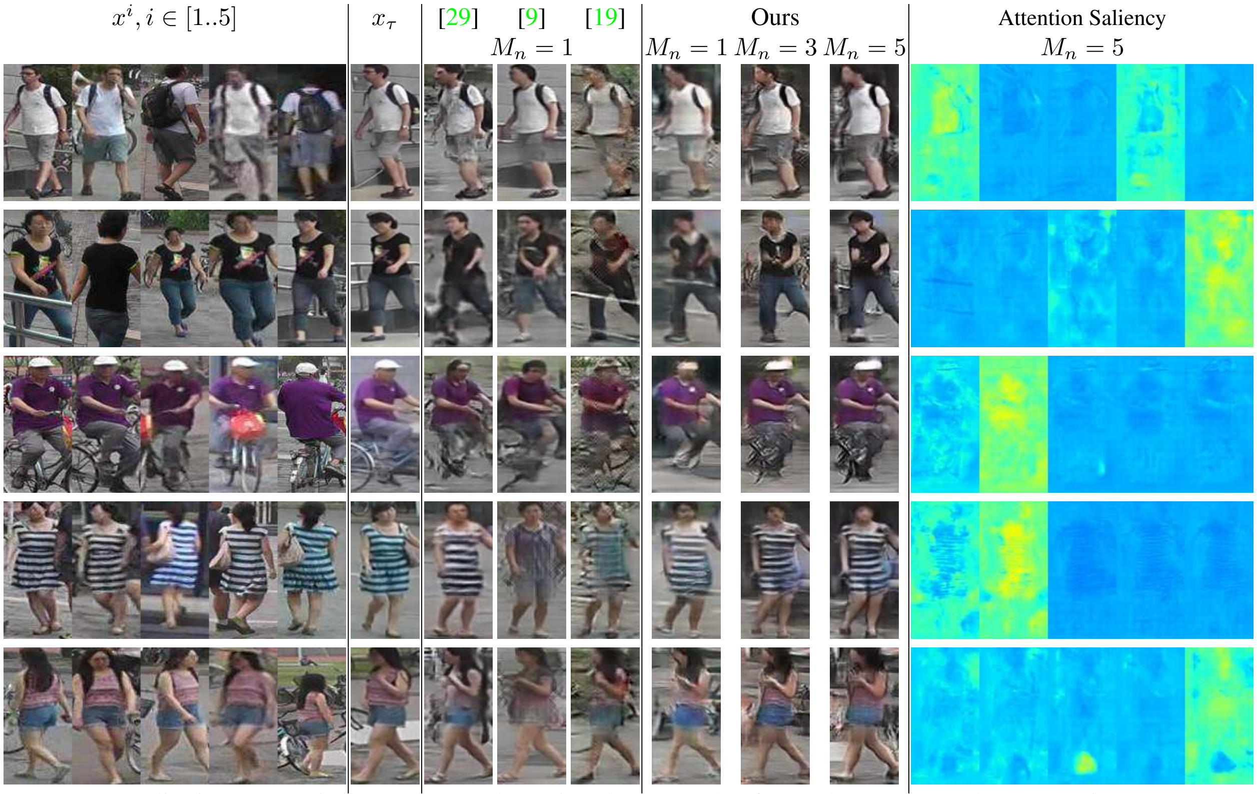


Figure 4: A qualitative comparison on the Market-1501 dataset. The first column shows the source images. Note that [29, 19, 9] use only the leftmost source image. The target poses are given by the ground truth images in column 2. In column 4, we show the results obtain by our model while increasing the number of source images. The source from the first column are added while increasing M_n from left to right. In the last column we show the saliency maps predicted by our model when using all the five source images. These maps are shown in the same order than the source images x^i.

score, but again at the cost of a drastically lower IS, meaning that we can generate more diverse and higher quality images. Moreover, we notice that [9] obtains a lower masked-SSIM. This seems to indicate that their high SSIM score is mostly due to a better background generation. Similar conclusions can be drawn for the DeepFashion dataset. We obtain the best IS and rank second in SSIM. Only [9] outperforms our model in terms of SSIM at the cost of a much lower IS value. The gain in performance seems smaller than on the market-1501 dataset. This is probably due to the lower pose diversity of the DeepFashion dataset.

Qualitative comparison. Fig. 4 shows some images obtained using the Market-1501 dataset. We compare our results with the images generated by three methods for which the code is publicly available [9, 19, 29]. The source images are shown in the first column. Note that the single-source methods use only the leftmost image. The target pose is extracted from the ground-truth target image. We display the generated images varying $M_n \in \{1, 3, 5\}$. We also show the corresponding saliency tensors A_r^i (see Sec. 3.2) at the highest resolution $r = 1$. Specifically, we use $M_n = 5$ and, at each (h, w) location in A_r^i, we average the values over the channel axis (c) using a color scale from dark blue (0 values) to orange (1 values).

The qualitative results confirm the quantitative evaluation since we clearly obtain better images when we increase the number of source images. The images become sharper and with more details and contain less artifacts. By looking at the saliency maps, we observe that our model uses mostly the source images in which the human pose is similar to the target pose. For instance in row 1 and 4, the model has high attention values for the two frontal images but very low values for the back view images. Interestingly, in row 1, among the two source images with a pose similar to the target pose, the saliency values are lower for the more blurry image. This illustrates that, between two images with similar poses, our attention model favours the image with the highest quality. Concerning the comparison with the state of the art, we observe that our model better preserves the details of the source images. In general, we obtain higher-quality details and less artefacts. For instance, in row 3, the three other methods do not generate the white hat nor the small logo of the shirt. In particular, the V-UNet architecture proposed in [9] generates realistic images but with less accurate details. This can be easily observed in the last two rows where the colors of the clothes are wrongly generated.

4.4. Ablation study and qualitative analysis

In this section we present an ablation study to clarify the impact of each part of our proposal on the final performance. We first describe the compared methods, obtained by "amputating" important parts of the full-pipeline presented in Sec. 3. The discriminator architecture is the same for all the methods.

- *Avg No-d*: In this baseline version of our method we use the encoder described in Sec. 3.3 *without* the deformation-based alignment of the features with the target pose. For the decoder, we use a standard U-Net decoder without attention module. More precisely, the tensors provided by the skip connections of each encoder are simply averaged and concatenated with the decoder tensors as in the original U-Net. In other words, Eq. (2) is replaced by the average over each convolution layer of the decoder, similarly to (1).
- *Avg*: We use the encoder described in Sec. 3.3 and the same decoder of *Avg No-d*.
- *Att. 2D*: We use an attention model similar to the full model described in Sec. 3.2. However, in Eq. (3), $f(\cdot, \cdot)[c]$ is not used.
- *Full*: This is the full-pipeline as described in Sec. 3.

		Market-1501				DeepFashion	
Model	M_n	*SSIM*	*IS*	*mask-SSIM*	*mask-IS*	*SSIM*	*IS*
Single source	1	0.27	3.251	0.771	3.614	0.757	3.420
Avg No-d	2	0.258	3.182	0.766	3.658	0.756	3.274
Avg	2	**0.294**	3.468	**0.779**	3.274	0.785	3.321
Att. 2D	2	0.285	3.460	0.777	3.632	**0.769**	3.375
Full	2	0.285	**3.474**	0.778	**3.634**	**0.769**	**3.421**
Avg	5	0.299	3.383	0.782	3.751	0.763	**3.454**
Att. 2D	5	**0.308**	3.159	**0.792**	3.606	0.773	3.411
Full	5	0.306	**3.444**	0.788	**3.814**	**0.774**	3.416

Table 2: Quantitative ablation study on the Market-1501 and the DeepFashion dataset.

Tab. 2 shows a quantitative evaluation. First, we notice that our method without spatial deformation performs poorly on both datasets. This is particularly evident with the *SSIM*-based scores. This confirms the importance of source-target alignment before computing a position-dependent attention. Interestingly, when using only two source images, *Avg*, *Att. 2D* and *Full* perform similarly to each other on the Market-1501 dataset. However, when we dispose of more source images we clearly observe the benefit of using our proposed attention approach. *Avg* performs constantly worst than our *Full* pipeline. The 2D attention model outputs images with higher *SSIM*-based scores but with lower IS values. Concerning the DeepFashion dataset, our attention model performs that the simpler approach with 2 and 5 source images.

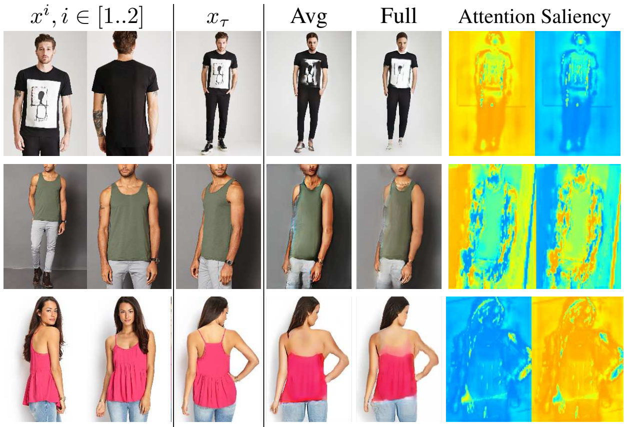


Figure 5: A qualitative ablation study on the Deep-Fashion dataset. We compare *Avg* with *Full* using $M_n = 2$. The attention saliency are displayed in the same order than the source images x^i.

In Fig. 5 we compare *Avg* with *Full* using $M_n = 2$. The advantage of using *Full* is is clearly illustrated by the fact that *Avg* mostly performs an average of the front and back images. In the second row, *Full* reduces the amount of artefacts. Interestingly, in the last row, *Full* fails to generate correctly the new viewpoint but we see that it chooses to focus on the back view in order to generate the collar.

5. Conclusion

In this work we introduced a generalization of the person-image generation problem. Specifically, a human image is generated conditioned on a target pose and *a set* $\boldsymbol{X}$ of source images. This makes it possible to exploit multiple and possibly complementary images. We introduced an attention-based decoder which extends the U-Net architecture to a multiple-input setting. Our attention mechanism selects relevant information from different sources and image regions. We experimentally validate our approach on two different datasets. As future works, we plan to extend this work to video generation and in particular image animation [27, 28].

Acknowledgments

We want to thank the NVIDIA Corporation for the donation of the GPUs used in this project. This project has received funding from the European Research Council (ERC) (Grant agreement No.788793-BACKUP) and the PRIN PREVUE project.

References

[1] S. Azadi, M. Fisher, V. Kim, Z. Wang, E. Shechtman, and T. Darrell. Multi-content gan for few-shot font style transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 5

[2] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv:1409.0473*, 2014. 2

[3] G. Balakrishnan, A. Zhao, A. V. Dalca, F. Durand, and J. Guttag. Synthesizing images of humans in unseen poses. In *CVPR*, 2018. 1, 2

[4] Y. Blau and T. Michaeli. The perception-distortion tradeoff. In *CVPR*, pages 6228–6237, 2018. 6

[5] Z. Cao, T. Simon, S. Wei, and Y. Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. In *CVPR*, 2017. 1, 4, 5

[6] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros. Everybody dance now. *arXiv preprint arXiv:1808.07371*, 2018. 1, 2

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. Ieee, 2009. 5

[8] H. Dong, X. Liang, K. Gong, H. Lai, J. Zhu, and J. Yin. Soft-gated warping-gan for pose-guided person image synthesis. *arXiv preprint arXiv:1810.11610*, 2018. 2

[9] P. Esser, E. Sutter, and B. Ommer. A variational u-net for conditional appearance and shape generation. In *CVPR*, pages 8857–8866, 2018. 1, 5, 6, 7, 8

[10] S. Gidaris and N. Komodakis. Dynamic few-shot visual learning without forgetting. In *CVPR*, 2018. 2

[11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 2

[12] R. A. Guler, N. Neverova, and I. Kokkinos. Densepose: Dense human pose estimation in the wild. 2018. 2

[13] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017. 1, 2, 3, 4, 5

[14] D. P. Kingma and M. Welling. Auto-encoding variational bayes. In *ICLR*, 2014. 2

[15] C. Lassner, G. Pons-Moll, and P. V. Gehler. A generative model of people in clothing. In *ICCV*, 2017. 1, 2

[16] J. Liu, B. Ni, Y. Yan, P. Zhou, S. Cheng, and J. Hu. Pose transferrable person re-identification. In *CVPR*, pages 4099–4108, 2018. 1

[17] L. Liu, W. Xu, M. Zollhoefer, H. Kim, F. Bernard, M. Habermann, W. Wang, and C. Theobalt. Neural animation and reenactment of human actor videos. *arXiv preprint arXiv:1809.03658*, 2018. 1, 2

[18] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, 2016. 6

[19] L. Ma, X. Jia, Q. Sun, B. Schiele, T. Tuytelaars, and L. Van Gool. Pose guided person image generation. In *NIPS*, 2017. 1, 2, 5, 6, 7

[20] L. Ma, Q. Sun, S. Georgoulis, L. Van Gool, B. Schiele, and M. Fritz. Disentangled person image generation. In *CVPR*, pages 99–108, 2018. 2, 5, 6

[21] N. Neverova, R. Alp Guler, and I. Kokkinos. Dense pose transfer. In *ECCV*, 2018. 2

[22] H. Park, Y. Yoo, and N. Kwak. Mc-gan: Multi-conditional generative adversarial network for image synthesis. In *BMVC*, 2018. 5

[23] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training gans. In *NIPS*, 2016. 6

[24] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *NIPS*, 2015. 2

[25] C. Si, W. Wang, L. Wang, and T. Tan. Multistage adversarial losses for pose-based human image synthesis. In *CVPR*, pages 118–126, 2018. 1, 2

[26] A. Siarohin, S. Lathuilière, E. Sangineto, and N. Sebe. Appearance and pose-conditioned human image generation using deformable gans. *IEEE T-PAMI*, 2019. 1, 2, 4, 5, 6

[27] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe. Animating arbitrary objects via deep motion transfer. In *CVPR*, 2019. 8

[28] A. Siarohin, S. Lathuilière, S. Tulyakov, E. Ricci, and N. Sebe. First order motion model for image animation. In *Neurips*. 2019. 8

[29] A. Siarohin, E. Sangineto, S. Lathuilière, and N. Sebe. Deformable gans for pose-based human image generation. In *CVPR*, 2018. 1, 2, 4, 5, 6, 7

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv:1409.1556*, 2014. 5

[31] K. Sricharan, R. Bala, M. Shreve, H. Ding, K. Saketh, and J. Sun. Semi-supervised conditional GANs. *arXiv:1708.05789*, 2017. 2

[32] S.-H. Sun, M. Huh, Y.-H. Liao, N. Zhang, and J. J. Lim. Multi-view to novel view: Synthesizing novel views with self-learned confidence. In *ECCV*, pages 155–171, 2018. 2

[33] D. Ulyanov, A. Vedaldi, and V. S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv:1607.08022*, 2016. 5

[34] O. Vinyals, S. Bengio, and M. Kudlur. Order matters: Sequence to sequence for sets. *arXiv:1511.06391*, 2015. 2

[35] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra. Matching networks for one shot learning. In *NIPS*, 2016. 2

[36] O. Vinyals, M. Fortunato, and N. Jaitly. Pointer networks. In *NIPS*, 2015. 2

[37] J. Walker, K. Marino, A. Gupta, and M. Hebert. The pose knows: Video forecasting by generating pose futures. In *ICCV*, 2017. 1

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE TIP*, 2004. 6

[39] J. Weston, S. Chopra, and A. Bordes. Memory networks. *arXiv:1410.3916*, 2014. 2

[40] G. Yildirim, C. Seward, and U. Bergmann. Disentangling multiple conditional inputs in gans. *ACM SIGKDD W*, 2018. 5

[41] P. Zablotskaia, A. Siarohin, B. Zhao, and L. Sigal. Dwnet: Dense warp-based network for pose-guided human video generation. 2019. 2

[42] M. Zanfir, A.-I. Popa, A. Zanfir, and C. Sminchisescu. Human appearance transfer. In *CVPR*, pages 5391–5399, 2018. 2

[43] B. Zhao, X. Wu, Z. Cheng, H. Liu, and J. Feng. Multi-view image generation from a single-view. *arXiv:1704.04886*, 2017. 1, 2

[44] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. 5

[45] Z. Zheng, L. Zheng, and Y. Yang. Unlabeled samples generated by GAN improve the person re-identification baseline in vitro. In *ICCV*, 2017. 1