A Generative Model of People in Clothing

Christoph Lassner[1, 2] classner@tue.mpg.de
Gerard Pons-Moll[2] gpons@tue.mpg.de
Peter V. Gehler[3,*] peter.gehler@uni-wuerzburg.de

[1]BCCN, Tübingen [2]MPI for Intelligent Systems, Tübingen [3]University of Würzburg

Figure 1: Random examples of people generated with our model. For each row, sampling is conditioned on the silhouette displayed on the left. Our proposed framework also supports unconditioned sampling as well as conditioning on local appearance cues, such as color.

Abstract

We present the first image-based generative model of people in clothing for the full body. We sidestep the commonly used complex graphics rendering pipeline and the need for high-quality 3D scans of dressed people. Instead, we learn generative models from a large image database. The main challenge is to cope with the high variance in human pose, shape and appearance. For this reason, pure image-based approaches have not been considered so far. We show that this challenge can be overcome by splitting the generating process in two parts. First, we learn to generate a semantic segmentation of the body and clothing. Second, we learn a conditional model on the resulting segments that creates realistic images. The full model is differentiable and can be conditioned on pose, shape or color. The result are samples of people in different clothing items and styles. The proposed model can generate entirely new people with realistic clothing. In several experiments we present encouraging results that suggest an entirely data-driven approach to people generation is possible.

[*] This work was performed while P. V. Gehler was with the BCCN[1] and MPI-IS[2].

1. Introduction

Perceiving people in images is a long standing goal in computer vision. Most work focuses on detection, pose and shape estimation of people from images. In this paper, we address the inverse problem of automatically generating images of people in clothing. A traditional approach to this task is to use computer graphics. A pipeline including 3D avatar generation, 2D pattern design, physical simulation to drape the cloth, and texture mapping is necessary to render an image from a 3D scene.

Graphics pipelines provide precise control of the outcome. Unfortunately, the rendering process poses various challenges, all of which are active research topics and mostly require human input. Especially clothing models require expert knowledge and are laborious to construct: the physical parameters of the cloth must be known in order to achieve a realistic result. In addition, modeling the complex interactions between the body and clothing and between different layers of clothing presents challenges for many current systems. The overall cost and complexity limits the applications of realistic cloth simulation. Data driven models of cloth can make the problem easier, but available data of clothed people in 3D is scarce.

Figure 2: Sample results for virtual humans from existing approaches (ltr., ttb.): [35], [34], local warping. [16], [47], animated 3D scans in real environments. [48], 3D avatars in virtual environments. [5], 3D avatars in real environments.

Here, we investigate a different approach and aim to sidestep this pipeline. We propose ClothNet, a generative model of people learned directly from images. ClothNet uses task specific information in the form of a 3D body model, but is mostly data-driven. A basic version (*ClothNet-full*) allows to randomly generate images of people from a learned latent space. To provide more control we also introduce a conditional model (*ClothNet-body*). Given a synthetic image silhouette of a projected 3D body model, ClothNet-body produces random people with similar pose and shape in different clothing styles (see Fig. 1).

Learning a direct image based model has several advantages: firstly, we can leverage large photo-collections of people in clothing to learn the statistics of how clothing maps to the body; secondly, the model allows to dress people fully automatically, producing plausible results. Finally, the model learns to add realistic clothing accessories such as bags, sunglasses or scarfs based on image statistics.

We run multiple experiments to assess the performance of the proposed models. Since it is inherently hard to evaluate metrics on generative models, we show representative results throughout the paper and explore the encoded space in a principled way in several experiments. To provide an estimate on the perceived quality of the generated images, we conducted a user study. With a rate of 24.7%or more, depending on the ClothNet variant, humans take the generated images for real.

2. Related Work

2.1. 3D Models of People in Clothing

There exists a large and diverse literature on the topic of creating realistic looking images of people. They can be grouped into rendering systems and systems that attempt to modify existing real photographs (warping pixels).

Warping pixels. Xu et al. [50] pose the problem as one of video retrieval and warping. Rather than synthesizing meshes with wrinkles, they look up video frames with the right motions. Similarly, in [18, 56] an unclothed body model is fit to multi-camera and monocular image data. The body is warped and the image reshaped accordingly.

Two prominent works that aim to reshape people in photos are [34, 35] (*c.f*. Fig. 2). A number of different synthetic sources has been used in [35] for improvement of pedestrian detection. The best performing source is obtained morphing images of people, but requires data from a multi-view camera setup; consequently only 11 subjects were used. Subsequent work [34] reshaped images but required significant user interaction. All aforementioned approaches require manual input and can only modify existing photographs.

Rendering systems. Early works synthesizing people from a body model are [37, 42, 45]. Here, renderings were limited to depth images with the goal of improving human pose estimation. The work of [5] combines real photographs of clothing with a SCAPE [2] body model to generate synthetic people whose pose can be changed (*c.f*. Fig. 2). The work of [38] proposes a pose-aware blending of 2D images, limiting the ability to generalize.

A different line of works use rendering engines with different sources of inputs. In [16], a mixed reality scenario is created by rendering 3D rigged animation models into videos, but it is clearly visible that results are synthetic (*c.f*. Fig. 2). The work of [47] combines a physical rendering engine together with real captured textures to generate novel views of people. All these works use 3D body models without clothing geometry, hence the quality of the results is limited. One exception is [12] where only the 2D contour of the projected cloth is modeled.

3D clothing models. Much of the work in the field of clothing modeling is focused on how to make simulation faster [9, 30], particularly by adding realistic wrinkles to low-resolution simulations [20, 22]. Other approaches have focused on taking off-line simulations and learning data driven models from them [7, 13, 22, 44, 49]. The authors of [39] simulate clothing in 3D and project back to the image for augmentation. All these approaches require predesigned garment models. Furthermore, current 3D models are not fully automatic, restricted to a set of garments or not photorealistic. The approach of [36] automatically captures real clothing, estimates body shape and pose [55] and retargets to new body shapes. The approach does not require predefined 3D garments but requires a 3D scanner.

2.2. Generative Models

Variational models and GANs. Variational methods are a well-principled approach to build generative models. Kingma and Welling developed the Variational Autoencoder [25], which is a key component of our method.

In their original work, they experimented with a multi-layer perceptron on low resolution data. Since then, multiple projects have designed VAEs for higher resolutions, *e.g.*, [51]. They use a CVAE [24, 43] to condition generated images on vector embeddings.

Recurrent formulations [11, 32, 46] enable to model complex structures, but again only at low resolution. With [8], Denton et al. address the resolution issue explicitly and propose a general architecture that can be used to improve network output. This strategy could be used to enhance ClothNet. Generative Adversarial Networks [10] use a second network during the training to distinguish between training data and generated data to enhance the loss of the trained network. We use this strategy in our training to increase the level of detail of the generated images. Most of the discussed works use resolutions up to 64x64 while we aim to generate 256x256 images. For our model design we took inspiration from encoder-decoder architectures such as the U-Net [40], Context Encoders [33] and the image-to-image translation networks [17].

Inpainting methods. Recent inpainting methods achieve a considerable level of detail [33, 52] in resolution and texture. To present a comparison with a state-of-the art encoder-decoder architecture, we provide a comparison with [33] in our experiments. [41] works directly on a texture map of a 3D model. Future work could explore to combine it with our approach from 2D image databases.

Deep networks for learning about 3D objects. There are several approaches to reason about 3D object configuration with deep neural networks. The work of Kulkarni [26] use VAEs to model 3D objects with very limited resolution and assume that a graphics engine and object model are available at learning time. In [6] an encoder-decoder CNN in voxel space is used for 3D shape completion, which requires 3D ground truth. The authors of [31] develop a generative model to create depth training data for articulated hands. This avoids the problem of generating realistic appearance.

3. Chictopia made SMPL

To train a supervised model connecting body parameters to fashion, we need a dataset providing information about both. Datasets for training pose estimation systems [1, 16, 19] capture complex appearance, shape and pose variation, but are not labeled with clothing information. The Chictopia10K dataset [27] contains fine-grained fashion labels but no human pose and shape annotations. In the following sections, we explain how we augmented Chictopia10K in an automatic manner so that it can be used as a resource for generative model training.

Fitting SMPL to Chictopia10K. The Chictopia10K dataset consists of 17,706 images collected from the chictopia fashion blog[1]. For all images, a fine-grained segmentation into 18 different classes (*c.f.* [27]) is provided: 12 clothing categories, background and 5 features such as hair and skin, see Fig. 4. For shoes, arms and legs, person centric left and right information is available. We augment the Chictopia10K dataset with pose and shape information by fitting the 3D SMPL body model [28] to the images using the SMPLify pipeline [3]. SMPLify requires a set of 2D keypoint locations which are computed using the DeeperCut [14] pose estimator.

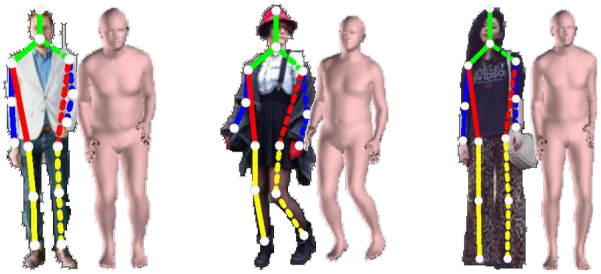


Figure 3: Example images from the Chictopia10K dataset [27], detected joints from DeeperCut [14] and the final SMPLify fits [3]. Typical failure cases are foot and head orientation (**center**). The pose estimator works reliably even with wide clothing and accessories. (**right**).

Figure 4: Example annotations from the Chictopia10K dataset [27] before and after processing (for each pair **left** and **right** respectively). Holes are inpainted and a face shape matcher is used to add facial features. The rightmost example shows a failure case of the face shape matcher.

Qualitative results of the fitting procedure are shown in Fig. 3. The pose estimator has a high performance across the dataset and the 3D fitting produces few mistakes. The most frequent failures are results with wrong head and foot orientation. To leverage as much data as possible to train our supervised models, we refrain from manually curating the results. Since we are interested in overall body shape and pose, we are using a six-part segmented projection of the SMPL fits for conditioning of ClothNet-body. Due to the rough segmentation, segmented areas are still representative even if orientation details do not match.

Face Shape Matching and Mask Improvement. We further enhance the annotation information of Chictopia10K and include facial landmarks to add additional guidance to the generative process. With only a single label for the entire face, we found that all models generate an almost blank skin area in the face.

[1] http://www.chictopia.com/

We use the dlib [23] implementation of the fast facial shape matcher [21] to enhance the annotations with face information. We reduce the detection threshold to oversample and use the face with the highest intersection over union (IoU) score of the detection bounding box with ground truth face pixels. We only keep images where either no face pixels are present or the IoU score is above a certain threshold. A threshold score of 0.3 was sufficient to sort out most unusable fits and still retain a dataset of 14,411 images (81,39%).

Furthermore, we found spurious "holes" in the segmentation masks to be problematic for the training of generative models. Therefore, we apply the morphological "close" and "blackhat" operations to fill the erroneously placed background regions. We carefully selected the kernel size and found that a size of 7 pixels fixes most mistakes while retaining small structures. You can find examples of original and processed annotations in Fig. 4.

4. ClothNet

Currently, the most visually appealing technique to create fine-grained textures at 256×256 resolution are image-to-image translation networks [17]. An encoder-decoder structure with skip connections between their respective layers allows the model to retain sharp edges.

However, applying image-to-image translation networks directly to the task of predicting dressed people from SMPL sketches as displayed in Fig. 1 does not produce good results (we provide example results from such a model in Fig. 10b). The reason is that there are many possible completions for a single 3D pose. The image-to-image translation model can not handle such multi-modality. Furthermore, its sampling capabilities are poor. Variational Autoencoders, on the other hand, excel at encoding high variance for similar inputs and provide a principled way of sampling.

We combine the strengths of both, Variational Autoencoders and the image-to-image translation models, by stacking them in a two-part model: the *sketch* part is variational and deals with the high variation in clothing shape. Its output is a semantic segmentation map (sketch) of a dressed person. The second *portray* part uses the created sketch to generate an image of the person and can make of use of skip connections to produce a detailed output. In the following sections, we will introduce the modules we experimented with.

4.1. The Latent Sketch Module

The latent sketch module is a variational auto-encoder which allows to sample random sketches of people.

The Variational Auto-Encoder [25] consists of two parts, an *encoder* to a latent space, and a *decoder* from the latent space to the original representation. As for any latent variable model, the aim is to reconstruct the training set $\mathbf{x}$ from a latent representation $\mathbf{z}$. Mathematically, this means maximizing the data likelihood $p(\mathbf{x}) = \int p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})d\mathbf{z}$. In high dimensional spaces, finding the decoder parameters θ that maximize the likelihood is intractable. However, for many values of $\mathbf{z}$ the probability $p_\theta(\mathbf{x}|\mathbf{z})$ will be almost zero. This can be exploited by finding a function $q_\phi(\mathbf{z}|\mathbf{x})$, the *encoder*, parameterized by ϕ. It encodes a sample $\mathbf{x}^i$ and produces a distribution over z values that are likely to reproduce $\mathbf{x}^i$. To make the problem tractable and differentiable, this distribution is assumed to be Gaussian, $q_\phi(\mathbf{z}|\mathbf{x}^i) = \mathcal{N}(\boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i)$. The parameters $\boldsymbol{\mu}^i, \boldsymbol{\Sigma}^i$ are predicted by the ϕ-parameterized encoding neural network Enc_ϕ. The decoder is the θ parameterized neural network Dec_θ.

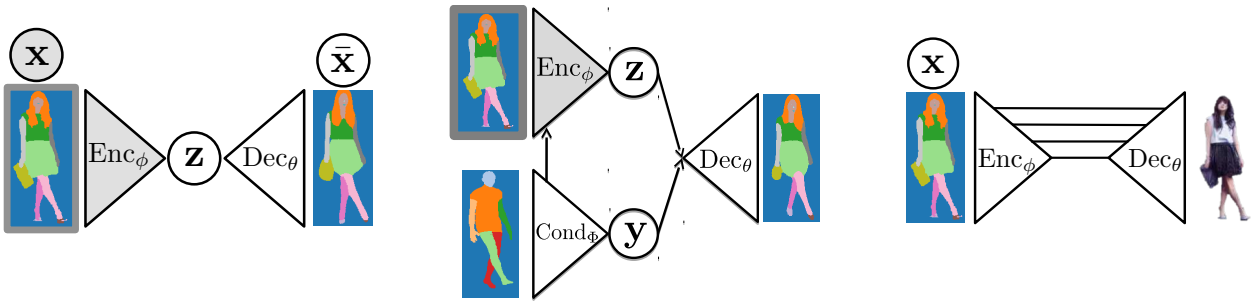
Another key assumption for VAEs is that the marginal distribution on the latent space is Gaussian distributed with zero mean and identity covariance, $p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$. Under these assumptions, the VAE objective (see [25] for derivations) to be maximized is

$$\sum_i E_{z\sim q}[\log p_\theta(\mathbf{x}^i|\mathbf{z})] - D_{KL}(q_\phi(\mathbf{z}|\mathbf{x}^i)||p(\mathbf{z})), \quad (1)$$

where $E_{z\sim q}$ indicates expectation over distribution q and D_{KL} denotes Kullback-Leibler (KL) divergence. The first term measures the decoder accuracy for the distribution produced by the encoder $q_\phi(\mathbf{z}|\mathbf{x})$ and the second term penalizes deviations of $q_\phi(\mathbf{z}|\mathbf{x}^i)$ from the desired marginal distribution $p(\mathbf{z})$. Intuitively, the second term prevents the encoding from carrying too much information about the input $\mathbf{x}^i$. Since both $q_\phi(\mathbf{z}|\mathbf{x}^i)$ and $p(\mathbf{z})$ are Gaussian, the KL divergence can be computed in closed form [25]. Eq. (1) is maximized using stochastic gradient ascent.

Computing Eq. (1) involves sampling; constructing a sampling layer in the network would result in a non differentiable operation. This can be circumvented using the reparameterization trick [25]. With this adaptation, the model is deterministic and differentiable with respect to the network parameters θ, ϕ. The latent space distribution is forced to follow a Gaussian distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$ during training. This implies that at test time one can easily generate samples $\bar{\mathbf{x}}^i$ by generating a latent sample $\mathbf{z}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and pushing it through the decoder $\bar{\mathbf{x}}^i = \text{Dec}_\theta(\mathbf{z}^i)$. This effectively ignores the encoder at test time.

Sketch encoding: we want to encode images $\mathbf{x} \in \mathbb{R}^{256\times256}$ of sketches of people into a 512-D latent space, $\mathbf{z} \in \mathbb{R}^{512}$. This resolution requires a sophisticated encoder and decoder layout. Hence, we combine the recently proposed encoder and decoder architecture for image-to-image translation networks [17] with the formulation of the VAE. We use a generalized Bernoulli distribution to model $p_\theta(\mathbf{x}^i|\mathbf{z})$. The architecture is illustrated in Fig. 5(a).



(a) (b) (c)

Figure 5: ClothNet modules: **(a)** the latent sketch module consists of a variational auto-encoder, **(b)** the conditional sketch module consists of a conditional variational auto-encoder and **(c)** the portray module is an image-to-image translation network that fills a sketch with texture. The modules in (a) and (c) are concatenated to form ClothNet-full and modules (b) and (c) are concatenated to form ClothNet-body. The learned latent representation $\mathbf{z}$ in (a) and (b) is a 512-D random variable that follows a multivariate Gaussian distribution. The variable $\mathbf{y}$ is a deterministic latent encoding of the body model silhouette that we use to condition on pose and shape. At test time in (a) and (b) one can generate a sample from the multivariate Gaussian $\mathbf{z}^i \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ and push them through the decoder network to produce random sketches of people in different clothing. We show in (a) and (b) the input to the encoder in gray color, indicating that they are not available at test time.

4.2. The Conditional Sketch Module

For some applications it may be desirable to generate different people in different clothing in a pose and shape specified by the user. To that end, we propose a module that we call *conditional sketch module*.

The conditional sketch module gives control of pose and shape by conditioning on a 3D body model sketch as illustrated in Fig. 5(b). We use a conditional variational autoencoder for this model (for a full derivation and description of the idea, we refer to [24]). To condition on an image $\mathbf{Y} \in \mathbb{R}^{256\times256}$ (a six part body model silhouette), the model is extended with a new encoding network Cond_Φ, with similar structure as Enc_ϕ. Since the conditioning variable is deterministic, the encoding is $\mathbf{y} = \text{Cond}_\Phi(\mathbf{Y})$. To provide the conditioning input to the encoder, we concatenate the output of the first layer of Cond_Φ to the output of the first layer of Enc_ϕ. To train the model, we use the same objective as in Eq. (1). Here, the decoder reconstructs a sample using both, $\mathbf{z}$ and $\mathbf{y}$, with $\bar{\mathbf{x}}^i = \text{Dec}_\theta(\mathbf{y}^i, \mathbf{z}^i)$ and the minimization of the KL-divergence term is only applied to $\mathbf{z}$.

4.3. The portray Module

For applications requiring a textured image of a person, the sketch modules can be chained to a portray module. We use an image-to-image translation model [17] to color the results from the sketch modules. With additional face information, we found this model to produce appealing results.

4.4. ClothNet-full and ClothNet-body

Once the sketch part and the portray part are trained, they can be concatenated to obtain a full generative model of images of dressed people. We refer to the concatenation of the latent sketch module with the portray module as ClothNet-full. The concatenation of the conditional sketch module with the portray module is named ClothNet-body. Several results produced by ClothNet-body are illustrated in Fig. 1. All stages of ClothNet-full and ClothNet-body are differentiable and implemented in the same framework. We trained the sketch and portray modules separately, simply because it is technically easier; propagating gradients through the full model is possible[2].

4.5. Network Architectures

Adhering to the image-to-image translation network architecture for designing encoders and decoders, we make use of LReLUs [29], batch normalization [15] and use fractionally strided convolutions [54]. We introduce weight parameters for the two loss components in Eq. (1) and balance the losses by weighing the KL component with factor 6.55. Then, the KL objective is optimized sufficiently well to create realistic samples $\boldsymbol{z}^i$ from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ after training. Full network descriptions are part of the supplementary material[3].

5. Experiments

5.1. The Latent Sketch Module

Variational Autoencoders are usually evaluated on the likelihood bounds on test data. Since we introduced weights into our loss function as described in Sec. 4.5, these would not be meaningful. However, for our purpose, the reconstruction ability of the sketch modules is just as important.

[2]To propagate gradients through the full model, it must represent sketches as $256 \times 256 \times 22$ probability maps instead of $256 \times 256 \times 3$ color maps since applying the color map function is not differentiable in general. The *portray* module results presented in the following sections have been created with color maps as inputs. The published code contains *portray* models for both, color map and probability map inputs.

[3]http://files.is.tue.mpg.de/classner/gp

Figure 6: A walk in latent space along the dimension with the highest variance. We built a PCA space on the 512 dimensional latent vector predictions of the test set and walk -1STD to 1STD in equidistant steps.

Model	Part	Accuracy	Precision	Recall	F1
LSM	Train	0.958	0.589	0.584	0.576
	Test	0.952	0.540	0.559	0.510
CSM	Train	0.962	0.593	0.591	0.587
	Test	0.950	0.501	0.502	0.488

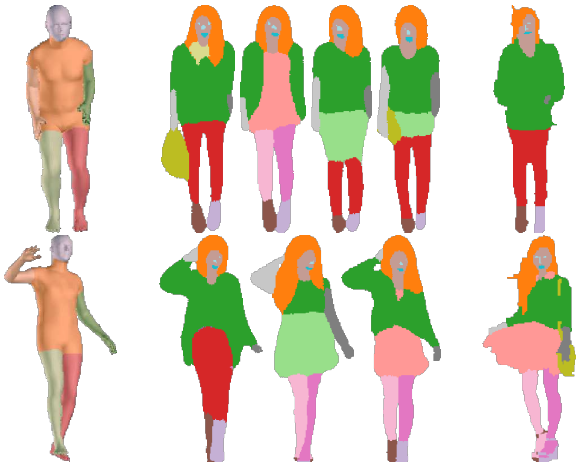
Table 1: Reconstruction metrics for the *Latent Sketch Module* (CSM) and *Conditional Sketch Module* (CSM). The overall reconstruction accuracy is high. The other metrics are dominated by classes with few labels. The CSM overfits faster.

We provide numbers on the quality of reconstruction in Tab. 1. The values are averages of the respective metrics over all classes. The overall reconstruction accuracy is high with a score of more than 0.95 in all settings. The other metrics are influenced by the small parts, in particular facial features. The CSM overfits faster than the LSM due to the additional information from the conditioning.

For a generative model, qualitative assessment is important as well. For this, we provide a visualization of a high variance dimension in latent space in Fig. 6. To create it, we produced the latent encodings $\mathbf{z}^i$ of all test set images. To normalize their distribution, we use the cumulative distribution function (CDF) values at their positions instead of the plain values. We then used a principal component analysis (PCA) to identify the direction with the most variance. In the PCA space, we take evenly spaced steps from minus one to plus one standard deviations; the PCA mean image is in the center of Fig. 6. Even though the direction encoding the most variance in PCA space only encodes roughly 1% of the full variance, the complexity of the task becomes obvious: this dimension encodes variations in pose, shape, position, scale and clothing types. The model learns to adjust the face direction in plausible ways.

5.2. The Conditional Sketch Module

As described in Sec. 4.2, we use a CVAE architecture to condition the generated clothing segmentations. We use the SMPL body model to represent the conditioning. However, instead of using the internal SMPL vector representation of shape and pose, we render the SMPL body in the desired



(a) (b) (c)

Figure 7: Per row: **(a)** SMPL conditioning for pose and shape, **(b)** sampled dressed sketches conditioned on the same sample in (a), **(c)** the nearest neighbor of the rightmost sample in (b) from the training set. The model learns to add various hair types, style and accessories.

configuration. We use six body parts: head, central body, left and right arms, left and right legs, to give the model local cues about the body parts.

We found the six part representation to be a good trade-off: using only a foreground-background encoding may convey too little information, especially about left and right parts. A too detailed segmentation introduces too much noise, since the data for training our supervised models has been acquired by automatic fits solely to keypoints. These fits may not represent detailed matches in all cases. You can find qualitative examples of conditional sampling in Fig. 1 and Fig. 7.

At test time, we encode the model sketch (Fig. 7(a)) to obtain $\mathbf{y}^i = \text{Cond}_\Phi(\mathbf{Y})$, sample from the latent space $\mathbf{z}^i \sim p(\mathbf{z}) = \mathcal{N}(\mathbf{0}, \mathbf{I})$ and obtain a clothed sketch with $\bar{\mathbf{x}}^i = \text{Dec}_\theta(\mathbf{y}^i, \mathbf{z}^i)$. For every sample $\mathbf{z}^i$ a new sketch $\bar{\mathbf{x}}^i$ is generated with different clothing but roughly the same pose and shape. Notice how different samples produce different hair and cloth styles as well as different configurations of accessories such as bags.

Figure 8: Conditioning on color: (**left**) sketch input to the network. (**right**) Four different outputs for four different color combinations. Color conditioning for the regions are shown in the boxes below the samples (**boxes below, ltr**): lower dress, upper dress, jacket, hat.

5.3. Conditioning on Color

As another example, we describe how to condition our model on color. During training, we compute the median color in the original image for every segment of a sketch. We create a new image by coloring the sketch parts with the respective median color. The concatenation of the colored image and the sketch are the new input to our portray module which is retrained on this input. Conditioning can then be achieved by selecting a color for a sketch segment. An example result is shown in Fig. 8. The network learns to follow the color cues, but still does not only generate plain color clothing, but places patterns, texture and wrinkles.

5.4. ClothNet

With the following two experiments, we want to provide an insight in how realistic the images are that are generated from the ClothNet-full pipeline.

5.4.1 Generating an Artificial Dataset

In the first experiment, we generate an artificial dataset and train a semantic segmentation network on the generated data. By comparing the performance of a discriminative model trained on real or synthetic images we can asses how realistic the generated images are.

For this purpose, we generate an equally sized dataset to our enhanced subset of Chictopia10K. We store the semantic segmentation masks generated from the latent sketch module as artificial 'ground truth' and the outputs from the full ClothNet-full pipeline as images. To make the images comparable to the Chictopia10K images, we add artificial background. Similar to [47], we sample images from the *dining room*, *bedroom*, *bathroom* and *kitchen* categories of the LSUN dataset [53]. Example images are shown in Fig. 9.

Even though the generated segmentations from our VAE model look realistic at first glance, some weaknesses become apparent when completed by the portray module: bulky arms and legs and overly smooth outlines of fine structures such as hair. Furthermore, the different statistics of facial landmark size to ground truth sketches lead to less realistic faces.

Train \ Test	Full Synth.	Synth. Text.	Real
Full Synth.	0.566 0.978	0.437 0.964	0.335 0.898
Synth. Text.	0.503 0.968	0.535 0.976	0.411 0.915
Real	0.448 0.955	0.417 0.957	0.522 0.951

Table 2: Segmentation results (per line: intersection over union (IoU), accuracy) for a variety of training and testing datasets. **Full Synth.** results are from the ClothNet-full model, **Synth. Text.** from the portray module on ground truth sketches.

Model	Real image rated gen.	Gen. image rated real
ClothNet-full	0.154	0.247
portray mod.	0.221	0.413

Table 3: User study results from 12 participants. The first row shows results for the full ClothNet-full model, the second for the portray module used on ground truth sketches.

We train a DeepLab ResNet-101 [4] segmentation model on real and synthetic data and evaluate on test images from all data sources. Evaluation results for this model can be found in Tab. 2. As expected, the models trained and tested from the same data source perform best. The model trained on the real dataset reaches the highest performance and can be trained longest without overfitting. The fully synthetic datasets lose at most 5.3 accuracy points compared to the model trained on real data. The IoU scores, however, suffer from fewer fine structures present in the generated data such as sunglasses and belts.

5.4.2 User Study

We performed a user study to quantify the realism of images. We set up an experiment to evaluate both stages of our model: one for images generated from the portray module on ground truth sketches and once for the full ClothNet-full model. For each of the experiments, we asked users to label 150 images for being a photo or generated from our model. 75 images were real Chictopia images, 75 generated with our model. Every image was presented for 1 second akin to the user study in Isola et al. [17]. We blanked out the faces of all images since those would be dominating the decision of the participants: this body part still provides the most reliable cues for artificially generated images. The first 10 rated images are ignored to let users calibrate on image quality.

Figure 9: Results from ClothNet with added random backgrounds. **First row**: results from ClothNet-full (*i.e.*, sketch and texture generation), **second row**: results from the portray module on ground truth sketches.

(a) (b)

Figure 10: Example results from **(a)** the context encoder architecture [33] from a ground truth sketch. Without skip connections, the level of predicted detail remains low. **(b)** Results from an image-to-image network trained to predict dressed people from six part SMPL sketches directly. Without the proposed two-stage architecture, the model is not able to determine shape and cloth boundaries.

With this setup we follow the setup of Isola et al. [17]. They use a forced choice between two images, one ground truth, one sketched by their model on ground truth segmentation. Since we do not have ground truth comparison images, we display one image at a time and ask for a choice. This setting is slightly harder for our model, since the user can focus on one image. The results for the 12 participants of our study are presented in Tab. 3. Even by the fully generative pipeline, users are fooled 24.7% of the time, by the portray module on ground truth sketches even 41.3% of the time. We observe a bias of users to assume that 50% of images are generated, resulting in a higher rate of misclassified real images for the stronger model. We used 50% fake and real images but did not mention this in the task description. For comparison: Isola et al. [17] report fooling rates of 18.9% and 6.1%, however on other modalities.

6. Conclusion

In this paper, we developed and analyzed a new approach to generate people with accurate appearance. We find that modern machine learning approaches may sidestep traditional graphics pipeline design and 3D data acquisition. This study is a first step and we anticipate that the results will become better once more data is available for training.

We enhanced the existing Chictopia10K dataset with face annotations and 3D body model fits. With a two-stage model for semantic segmentation prediction in the first, and texture prediction in the second stage, we presented a novel, modular take on generative models of structured, high-resolution images.

In our experiments, we analyzed the realism of the generated data in two ways: by evaluating a segmentation model trained on real data, on our artificial data and by conducting a user study. The segmentation model achieved 85% of its segmentation performance of the real data on the artificial, indicating that it 'recognized' most parts of the generated images equally well. In the user study, we could in 24.7% trick participants into mistaking generated images for real.

With this possibility to generate large amounts of training data at a very low computational and infrastructural cost together with the possibility to condition generated images on pose, shape or color, we see many potential applications for the presented method. We will make data and code available for academic purposes.

Acknowledgements This study is part of the research program of the Bernstein Center for Computational Neuroscience, Tübingen, funded by the German Federal Ministry of Education and Research (BMBF; FKZ: 01GQ1002). We gratefully acknowledge the support of the NVIDIA Corporation with the donation of a K40 GPU.

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2D human pose estimation: New benchmark and state of the art analysis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 3

[2] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. SCAPE: shape completion and animation of people. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 24(3):408–416, 2005. 2

[3] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *Proc. European Conf. on Computer Vision (ECCV)*, Oct. 2016. 3

[4] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3640–3649, 2016. 7

[5] W. Chen, H. Wang, Y. Li, H. Su, Z. Wang, C. Tu, D. Lischinski, D. Cohen-Or, and B. Chen. Synthesizing training images for boosting human 3d pose estimation. In *3D Vision (3DV)*, 2016. 2

[6] A. Dai, C. R. Qi, and M. Nießner. Shape completion using 3D-encoder-predictor CNNs and shape synthesis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[7] E. de Aguiar, L. Sigal, A. Treuille, and J. K. Hodgins. Stable spaces for real-time clothing. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 29(4):106:1–106:9, July 2010. 2

[8] E. L. Denton, S. Chintala, R. Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1486–1494, 2015. 3

[9] R. Goldenthal, D. Harmon, R. Fattal, M. Bercovier, and E. Grinspun. Efficient simulation of inextensible cloth. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 26(3):to appear, 2007. 2

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680, 2014. 3

[11] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015. 3

[12] P. Guan, O. Freifeld, and M. J. Black. A 2d human body model dressed in eigen clothing. In *Proc. European Conf. on Computer Vision*, pages 285–298. Springer, 2010. 2

[13] P. Guan, L. Reiss, D. Hirshberg, A. Weiss, and M. J. Black. DRAPE: DRessing Any PErson. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 31(4):35:1–35:10, July 2012. 2

[14] E. Insafutdinov, L. Pishchulin, B. Andres, M. Andriluka, and B. Schiele. Deepercut: A deeper, stronger, and faster multi-person pose estimation model. In *Proc. European Conf. on Computer Vision (ECCV)*, pages 34–50. Springer, 2016. 3

[15] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. 5

[16] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Trans. Pattern Analysis and Machine Intelligence (TPAMI)*, 2014. 2, 3

[17] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3, 4, 5, 7, 8

[18] A. Jain, T. Thormählen, H.-P. Seidel, and C. Theobalt. Moviereshape: Tracking and reshaping of humans in videos. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 29(6):148:1–148:10, Dec. 2010. 2

[19] S. Johnson and M. Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proc. British Machine Vision Conf. (BMVC)*, 2010. doi:10.5244/C.24.12. 3

[20] L. Kavan, D. Gerszewski, A. W. Bargteil, and P.-P. Sloan. Physics-inspired upsampling for cloth simulation in games. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 30(4):93:1–93:10, July 2011. 2

[21] V. Kazemi and J. Sullivan. One millisecond face alignment with an ensemble of regression trees. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1867–1874, 2014. 4

[22] D. Kim, W. Koh, R. Narain, K. Fatahalian, A. Treuille, and J. F. O'Brien. Near-exhaustive precomputation of secondary cloth effects. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 32(4):87:1–7, July 2013. 2

[23] D. E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research (JMLR)*, 10:1755–1758, 2009. 4

[24] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3581–3589, 2014. 3, 5

[25] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 2, 4

[26] T. D. Kulkarni, W. F. Whitney, P. Kohli, and J. Tenenbaum. Deep convolutional inverse graphics network. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2539–2547, 2015. 3

[27] X. Liang, C. Xu, X. Shen, J. Yang, S. Liu, J. Tang, L. Lin, and S. Yan. Human parsing with contextualized convolutional neural network. In *Proc. IEEE International Conf. on Computer Vision (ICCV)*, pages 1386–1394, 2015. 3

[28] M. Loper, N. Mahmood, J. Romero, G. Pons-Moll, and M. J. Black. Smpl: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, Oct. 2015. 3

[29] A. L. Maas, A. Y. Hannun, and A. Y. Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. International Conf. on Machine Learning (ICML)*, volume 30, 2013. 5

[30] R. Narain, A. Samii, and J. F. O'Brien. Adaptive anisotropic remeshing for cloth simulation. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 31(6):147:1–10, Nov. 2012. 2

[31] M. Oberweger, P. Wohlhart, and V. Lepetit. Training a feedback loop for hand pose estimation. In *Proc. IEEE International Conf. on Computer Vision (ICCV)*, pages 3316–3324, 2015. 3

[32] A. v. d. Oord, N. Kalchbrenner, and K. Kavukcuoglu. Pixel recurrent neural networks. In *Proc. International Conf. on Machine Learning (ICML)*, 2016. 3

[33] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros. Context encoders: Feature learning by inpainting. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2536–2544, 2016. 3, 8

[34] L. Pishchulin, A. Jain, M. Andriluka, T. Thormählen, and B. Schiele. Articulated people detection and pose estimation: Reshaping the future. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3178–3185. IEEE Press, 2012. 2

[35] L. Pishchulin, A. Jain, C. Wojek, M. Andriluka, T. Thormählen, and B. Schiele. Learning people detection models from few training samples. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011. 2

[36] G. Pons-Moll, S. Pujades, S. Hu, and M. Black. ClothCap: Seamless 4D clothing capture and retargeting. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 36(4), 2017. 2

[37] G. Pons-Moll, J. Taylor, J. Shotton, A. Hertzmann, and A. Fitzgibbon. Metric regression forests for human pose estimation. In *Proc. British Machine Vision Conf. (BMVC)*. BMVA Press, Sept. 2013. 2

[38] G. Rogez and C. Schmid. Mocap-guided data augmentation for 3d pose estimation in the wild. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3108–3116, 2016. 2

[39] L. Rogge, F. Klose, M. Stengel, M. Eisemann, and M. Magnor. Garment replacement in monocular video sequences. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 34(1):6:1–6:10, Nov. 2014. 2

[40] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proc. International Conf. on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015. 3

[41] S. Saito, L. Wei, L. Hu, K. Nagano, and H. Li. Photorealistic facial texture inference using deep neural networks. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[42] J. Shotton, T. Sharp, A. Kipman, A. Fitzgibbon, M. Finocchio, A. Blake, M. Cook, and R. Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013. 2

[43] K. Sohn, H. Lee, and X. Yan. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3483–3491, 2015. 3

[44] C. Stoll, J. Gall, E. de Aguiar, S. Thrun, and C. Theobalt. Video-based reconstruction of animatable human characters. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 29(6):139:1–139:10, Dec. 2010. 2

[45] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon. The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 103–110. IEEE, 2012. 2

[46] A. van den Oord, N. Kalchbrenner, L. Espeholt, O. Vinyals, A. Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems (NIPS)*, pages 4790–4798, 2016. 3

[47] G. Varol, J. Romero, X. Martin, N. Mahmood, M. Black, I. Laptev, and C. Schmid. Learning from synthetic humans. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2, 7

[48] D. Vázquez, A. M. Lopez, J. Marin, D. Ponsa, and D. Geronimo. Virtual and real world adaptation for pedestrian detection. *IEEE Trans. Pattern Analysis and Machine Intelligence (TPAMI)*, 36(4):797–809, 2014. 2

[49] H. Wang, J. F. O'Brien, and R. Ramamoorthi. Data-driven elastic models for cloth: Modeling and measurement. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 30(4):71:1–11, July 2011. 2

[50] F. Xu, Y. Liu, C. Stoll, J. Tompkin, G. Bharaj, Q. Dai, H.-P. Seidel, J. Kautz, and C. Theobalt. Video-based characters: Creating new human performances from a multi-view video database. *ACM Trans. Graphics (Proc. SIGGRAPH*, 30(4):32:1–32:10, July 2011. 2

[51] X. Yan, J. Yang, K. Sohn, and H. Lee. Attribute2image: Conditional image generation from visual attributes. In *Proc. European Conf. on Computer Vision (ECCV)*, pages 776–791. Springer, 2016. 3

[52] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li. High-resolution image inpainting using multi-scale neural patch synthesis. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[53] F. Yu, Y. Zhang, S. Song, A. Seff, and J. Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 7

[54] M. D. Zeiler, G. W. Taylor, and R. Fergus. Adaptive deconvolutional networks for mid and high level feature learning. In *Proc. IEEE International Conf. on Computer Vision (ICCV)*, pages 2018–2025. IEEE, 2011. 5

[55] C. Zhang, S. Pujades, M. Black, and G. Pons-Moll. Detailed, accurate, human shape estimation from clothed 3D scan sequences. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[56] S. Zhou, H. Fu, L. Liu, D. Cohen-Or, and X. Han. Parametric reshaping of human bodies in images. *ACM Trans. Graphics (Proc. SIGGRAPH)*, 29(4):126, 2010. 2