Multi-View Image Generation from a Single-View

Bo Zhao[1,4], Xiao Wu[1#], Zhi-Qi Cheng[1], Hao Liu[2], Zequn Jie[3] and Jiashi Feng[4]
[1]Southwest Jiaotong University, Chengdu, China [2]Tencent YouTu Lab, Hefei, China
[3]Tencent AI Lab, Shenzhen, China [4]National University of Singapore, Singapore
{zhaobo.cs,zhiqicheng,hfut.haoliu,zequn.nus}@gmail.com,wuxiaohk@home.swjtu.edu.cn,elefjia@nus.edu.sg

ABSTRACT

How to generate multi-view images with realistic-looking appearance from only a single view input is a challenging problem. In this paper, we attack this problem by proposing a novel image generation model termed VariGANs, which combines the merits of the variational inference and the Generative Adversarial Networks (GANs). It generates the target image in a coarse-to-fine manner instead of a single pass which suffers from severe artifacts. It first performs variational inference to model global appearance of the object (e.g., shape and color) and produces coarse images of different views. Conditioned on the generated coarse images, it then performs adversarial learning to fill details consistent with the input and generate the fine images. Extensive experiments conducted on two clothing datasets, MVC and DeepFashion, have demonstrated that the generated images with the proposed VariGANs are more plausible than those generated by existing approaches, which provide more consistent global appearance as well as richer and sharper details.

KEYWORDS

image generation, deep learning, generative adversarial networks

ACM Reference Format:
Bo Zhao, Xiao Wu, Zhi-Qi Cheng, Hao Liu, Zequn Jie and Jiashi Feng. 2018. Multi-View Image Generation from a Single-View. In *2018 ACM Multimedia Conference (MM '18), October 22–26, 2018, Seoul, Republic of Korea.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3240508.3240536

1 INTRODUCTION

Products in e-commerce websites are usually displayed with images from different views to attract customers. Multi-view images provide vivid and appealing product illustrations to potential customers. Unfortunately, such multi-view images are not always available. As shown in Fig. 1, when one occasionally notices a desired clothing item from a magazine, which is from a certain view, she may be interested to see the appearance from other views, such as the side or back. This kind of appearance is usually hard to imagine, especially for a well-designed fashion clothing. An automatic image generation system is desired in such scenario, which has practical meaning for e-commerce platforms and other applications, such as photo/video editing and AR/VR. Given a single-view clothing image, we aim to generate other views of the input image without requiring any extra information.

Figure 1: It is appealing to generate multi-view images from a single-view input, especially for a fashion clothing.

Image generation is a challenging task due to the high dimensionality of images and the complex configuration and layout of image contents. To tackle this challenging problem of generating multi-view images from a single-view observation, many approaches [2, 11, 36] first construct the 3D structure of the object and then generate desired target view images from that model. Other methods [19, 32, 37] learn the transformation between the input view and target view by relocating pixels. However, those methods mainly synthesize rigid objects, *e.g.*cars, chairs with simple textures. The generation of deformable objects with rich details such as clothes or human body has not been fully explored.

Beneficial from advanced models like Variational Autoencoder (VAE) [12]) and Generative Adversarial Networks (GANs) [7], recent works have demonstrated promising performance on realistic image generation. VAE adopts variational inference plus deep representation learning to learn a complex generative model and gets rid of the time-consuming sampling process. However, VAE usually fails to provide rich details in generated images. Another popular generative model, GANs, introduces a real-fake discriminator to supervise the learning of the generative model. Facilitated with the competition between discriminator and generator, GANs are advantageous in providing realistic details, but they usually introduce artifacts to the global appearance, especially when the image to be generated is large.

In this paper, we propose a novel image generation model, named *Variational GANs (VariGANs)* that combines the strengths of variational inference and adversarial training. The proposed model overcomes the limitations of GANs in modeling global appearance, by introducing internal variational inference in the generative model learning. A low resolution (LR) image capturing global appearance is firstly generated by variational inference. This process learns to draw rough shapes and colors of another image with a different view, conditioned on the given image and target view. With the generated LR image, VariGANs then performs adversarial learning to

# indicates the corresponding author.

MM '18, October 22–26, 2018, Seoul, Republic of Korea

ACM ISBN 978-1-4503-5665-7/18/10…$15.00 .
https://doi.org/10.1145/3240508.3240536

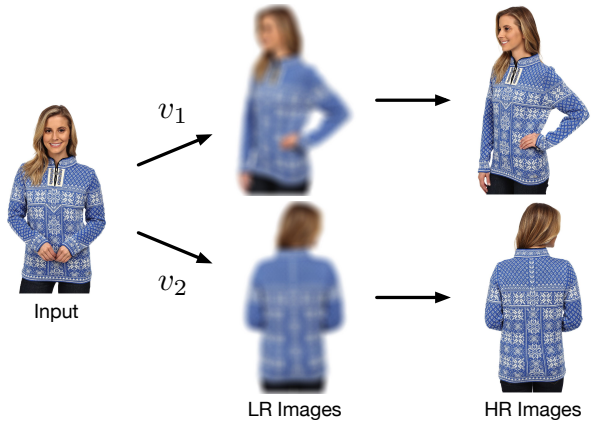


Figure 2: The photo-realistic image generation process of the proposed VariGANs. The low resolution (LR) images are firstly generated by variational inference for new views v_1 and v_2. The high resolution (HR) images are then generated by filling the details and correcting the defects through adversarial learning.

generate realistic high resolution (HR) image by filling richer details to the low resolution image. Since the LR image only has basic contour of the target object in a desired view, the fine image generation module just needs to focus on drawing details and rectifying defects in low resolution images. Fig. 2 illustrates the multi-view image generation process from coarse to fine, conditioned on a single-view input image. Decomposing the complicated image generation process into the above two complementary learning processes significantly simplifies the learning and produces more realistic-look multi-view images. Note that VariGANs is a generic model and can be applied to other image generation applications like style transfer, which will be explored in the future.

The main contributions are summarized as follows:

(1) To our best knowledge, this is the first work to address the new problem of generating multi-view clothing images based on a given clothing image of a certain view, which has practical significance.
(2) A novel VariGANs generation architecture is proposed for multi-view clothing image generation that adopts a novel coarse-to-fine image generation strategy. The proposed model is effective in both capturing global appearance and drawing richer details consistent with the input conditioned image.
(3) The proposed model is verified on two largest clothes image datasets and experiments demonstrate its superiority through comprehensive evaluations compared with other state-of-the-art approaches. The model and relevant code will be released upon acceptance.

2 RELATED WORK

Image Generation. Image generation has been a heated topic in recent years. Many approaches have been proposed with the emergence of deep learning techniques. Variational Autoencoder (VAE) [12] generates images based on the probabilistic graphical models, and are optimized by maximizing the lower bound of the data likelihood. Yan *et.al.* [31] propose the Attribute2Image, which generates images from visual attributes. They modeled an image as a composite of foreground and background and extended the VAE with disentangled latent variables. Gregor *et.al.* [8] propose the DRAW, which integrates the attention mechanism with VAE to generate realistic images recurrently by patches. Different from the generative parametric approaches, Generative Adversarial Networks (GANs) [7] introduce a generator and a discriminator in their model. The generator is trained to generate images to confuse the discriminator, and the discriminator is trained to distinguish between real and fake samples. Since then, many GANs-based models have been proposed, including Conditional GANs [16], Bi-GANs [4, 6], Semi-suprvised GANs [17], InfoGANs [3] and Auxiliary Classifier GANs [18]. GANs have been used to generate images from labels [16], texts [21, 34] and also images [1, 10, 20, 25, 27, 30, 33, 35, 38, 39]. Our proposed model is also an image-conditioned GANs, with generation capability strengthened by variational inference.

View Synthesis. Images with different views of the object can be easily genertaed with the 3D modeling of the object [2, 5, 11, 13, 36]. Hinton *et.al.* [9] proposed a transforming auto-encoder to generate images with view variance. Rezende *et.al.* [22] introduced a general framework to learn 3D structures from 2D observations with a 3D-2D projection mechanism. Yan *et.al.* [32] proposed Perspective Transformer Nets to learn the projection transformation after reconstructing the 3D volume of the object. Wu *et.al.* [29] also proposed the 3D-2D projection layers that enable the learning of 3D object structures using annotated 2D keypoints. They further proposed the 3D-GANs [29] which generates 3D objects from a probabilistic space by leveraging recent advances in volumetric convolutional networks and generative adversarial nets. Zhou *et.al.* [37] propose to synthesize novel views of the same object or scene corresponding by learning appearance flows. Most of these models are trained with the target view images or image pairs which can be generated by a graphic engine. Therefore, in theory, there are infinite amount of training data with desired view to train the model. However, in our task, the training data are limited in both views and numbers, which greatly adds the difficulty to generate image of different views.

3 PROPOSED METHOD

3.1 Problem Definition

We first define the problem of generating multi-view images from a single view input. Suppose we have a pre-defined set of view angles $\mathbf{V} = \{v_1, \cdots, v_i, \cdots, v_n\}$, where v_i corresponds to a specific view, *e.g.*front or side view. An object captured from the view v_i is denoted as I_{v_i}. Given the source image I_{v_i}, multi-view image generation is to generate another image I_{v_j} with a different view $v_j \in \mathbf{V}$ and $j \neq i$. Specifically, the goal is to learn the distribution $p(I_{v_j}|I_{v_i}, v_j)$ from a labeled dataset $(I_{v_j,1}, I_{v_i,1}), \ldots, (I_{v_j,n}, I_{v_i,n})$. Here, v_j is specified by users as an input to the model.

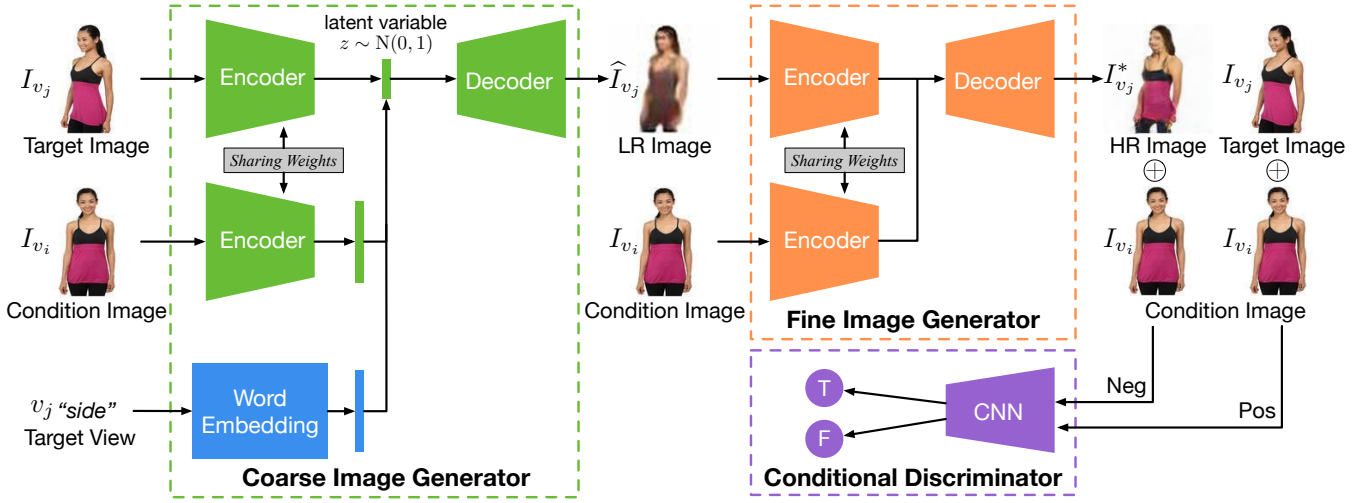


Figure 3: The architecture of the proposed VariGANs consists of three modules: coarse image generator, fine image generator and conditional discriminator. During training, a LR Image is firstly generated by the coarse image generator conditioned on the target image, conditioned image and target view. The fine image generator with skip connections is designed to generate the HR image. Finally, the HR image and the conditioned image are concatenated as negative pairs and passed to the conditional discriminator together with positive pairs (target image and condition image) to distinguish real and fake.

3.2 Variational GANs

Standard GANs will be applied to generate images of the desired properties based on the input. This type of model learns a generative model G based on the distribution of the desired images, sampling from which would provide new images. Different from other generative models, GANs employ an extra discriminative model D to supervise the generative learning process, instead of purely optimizing G to best explain training data via Maximum Likelihood Estimation.

The objective of GANs is formulated as:

$$\min_{\theta_G}\max_{\theta_D}\mathbb{E}_{I_{v_i}\sim p_{\text{data}}(I_{v_i}),I_{v_j}\sim p_{\text{data}}(I_{v_j}|I_{v_i},v_j)}[\log D(I_{v_i},I_{v_j})]+ \mathbb{E}_{z\sim p(z)}[\log(1-D(I_{v_i},G(z,I_{v_i},v_j))],$$

where a generative model G tries to generate real data I_{v_j} given noise $z \sim p(z)$ through minimizing its loss to fool an adversarial discriminator D, meanwhile, D tries to maximize its discrimination accuracy between real data and generated data.

However, because GANs are limited in capturing global appearance, it is difficult to learn a generator G to produce plausible images with high resolution, correct contour and rich details. To address this critical issue and generate more realistic images, the variational GANs (VariGANs) proposed in this work combines the merits of variation inference for modeling correct contours and adversarial learning to fill realistic details. It decomposes the generator into two components. One is for generating a coarse image through the variational inference model V and the other is for generating the final image with fine details based on the outcome from V. Formally, the objective of VariGANs is

$$\min_{\theta_G}\max_{\theta_D,\theta_V}\mathbb{E}_{I_{v_i}\sim p_{\text{data}}(I_{v_i}),I_{v_j}\sim p_{\text{data}}(I_{v_j}|I_v,v_j)}[\log D(I_{v_i},I_{v_j})]+ \mathbb{E}_{z\sim p(z)}[\log(1-D(I_{v_i},G(V(z,I_{v_i},v_j),I_{v_i},v_j))]. \quad (1)$$

Here z is the random latent variable and V is the coarse image generator. This objective can be optimized by maximizing the variational lower bound of V, maximizing the discrimination accuracy of D, and minimizing the loss of G against D. We will elaborate the models of V, G and D in the following subsections.

3.3 Coarse Image Generation

Given an input image I_{v_i} with the view of v_i, target view v_j, and latent variable z, the coarse image generator $V(I_{v_i},z,v_j)$ learns the distribution $p(\widehat{I}_{v_j}|I_{v_i},z)$ which focuses on modeling the global appearance. The parameters of the coarse image generator is denoted as θ_V. To alleviate difficulties of directly optimizing this log-likelihood function and avoid the time-consuming sampling, the variational Bayesian approach is applied to optimize the lower bound of the log-likelihood $\log p_\theta(\widehat{I}_{v_j}|I_{v_i},v_j)$, as proposed in [12, 23]. Specifically, an auxiliary distribution $q_\phi(z|\widehat{I}_{v_j},I_{v_i},v_j)$ is introduced to approximate the true posterior $p_{\theta_V}(z|\widehat{I}_{v_j},I_{v_i},v_j)$.

The conditional log-likelihood of the coarse image generator V is defined as

$$\log p_{\theta_V}(\widehat{I}_{v_j}|I_{v_i},v_j)=\mathcal{L}(\widehat{I}_{v_j},I_{v_i},v_j;\theta,\phi)+ \mathrm{KL}\left(q_\phi(z|\widehat{I}_{v_j},I_{v_i},v_j)||p_\theta(z|\widehat{I}_{v_j},I_{v_i},v_j)\right),$$

where the variational lower bound is

$$\mathcal{L}(\widehat{I}_{v_j},I_{v_i},v_j;\theta,\phi)=-\mathrm{KL}\left(q_\phi(z|\widehat{I}_{v_j},I_{v_i},v_j)||p_\theta(z)\right) + E_{q_\phi(z|\widehat{I}_{v_j},I_{v_i},v_j)}[\log p_\theta(\widehat{I}_{v_j}|I_{v_i},v_j,z)], \quad (2)$$

where the first KL term in Eqn. (2) is a regularization term that reduces the gap between the prior $p(z)$ and the proposal distribution $q_\phi(z|\widehat{I}_{v_j},I_{v_i},v_j)$. The second term $\log p_{\theta_V}(\widehat{I}_{v_j}|I_{v_i},v_j,z)$ is the log-likelihood of samples and is usually measured by the reconstruction loss, *e.g.*, ℓ_1 used in our model.

3.4 Fine Image Generation

After obtaining the low resolution image $\widehat{I}_{v_j}$ of the desired output I_{v_j}, the fine image generation module learns another generator G that maps the low resolution image $\widehat{I}_{v_j}$ to the high resolution image $I^*_{v_j}$ conditioned on the input I_{v_i}. The generator G is trained to generate images that cannot be distinguished from "real" images by an adversarial conditional discriminator, D, which is trained to distinguish as well as possible the generator's "fakes". See Eqn. (1).

Since the multi-view image generator needs to not only fool the discriminator but also be visually similar to the ground truth of the target image with a different view, the ℓ_1 loss is also added for the generator. The ℓ_1 loss is chosen because it alleviates over-smoothing artifacts compared with ℓ_2 loss.

After that, the GANs of fine image generation train the discriminator D and the generator G by alternatively maximizing $\mathcal{L}_D$ in Eqn. (3) and minimizing $\mathcal{L}_G$ in Eqn. (4):

$$\begin{aligned}\mathcal{L}_D =& \mathbb{E}_{(I_{v_i}, I_{v_j})\sim p_{\text{data}}}[\log D(I_{v_i}, I_{v_j})]+ \\ & \mathbb{E}_{z\sim p(z)}[\log(1 - D(I_{v_i}, G(\widehat{I}_{v_j}(z), I_{v_i}, v_j)))], \qquad (3)\\ \mathcal{L}_G =& \mathbb{E}_{z\sim p(z)}[\log(1 - D(I_{v_i}, G(\widehat{I}_{v_j}(z), I_{v_i}, v_j]+ \\ & \lambda \| I_{v_j} - G(\widehat{I}_{v_j}(z), I_{v_i}, v_j)\|_1, \qquad (4)\end{aligned}$$

where $\widehat{I}_{v_j}$ is the coarse image generated by V. The real images I_{v_i} and I_{v_j} are from the true data distribution.

3.5 Network Architecture

The overall architecture of the proposed model in the training phase is illustrated in Fig. 3. It consists of three modules: the coarse image generator, the fine image generator and the conditional discriminator. During training, the target view image I_{v_j} and the conditioned image I_{v_i} are passed through two Siamese-like encoders to learn their representations respectively. By word embedding, the input with desired view angle v_j is transformed into a vector. The representations of I_{v_i}, I_{v_j} and v_j are combined to generate the latent variable z. However, during testing, there is no target image I_{v_i} and the encoder for it. The latent variable z is randomly sampled and combined with the representation of the condition image I_{v_i} and the target view v_j to generate the target view LR image $\widehat{I}_{v_j}$. After that, I_{v_i} and $\widehat{I}_{v_j}$ are sent to the fine image generator to generate the HR image $I^*_{v_j}$. Similar to the coarse image generation module, the fine image generation module also contains two Siamese-like encoders and a decoder. Moreover, there are skip connections between mirrored layers in the encoder and decoder stacks. By the channel concatenation of the HR image $I^*_{v_j}$ and the condition image I_{v_i}, a conditional discriminator is adopted to distinguish whether the generated image is real or fake.

Coarse Image Generator. There are several convolution layers in the encoder of the coarse image generator to down sample the input image to an $M_l \times 1 \times 1$ tensor. A fully-connected layer is then topped to transform the tensor to an M_l-D representation. The encoders for the target image and the condition image share the weights. A word embedding layer is employed to embed the target view into an M_l-D vector. The representations of the target image, the conditioned image and the view embedding are combined and transformed to an M_l-D latent variable. Finally, the latent variable together with the conditioned image representation and the view embedding are passed through a series of de-convolutional layers to generate a $W_{\text{LR}} \times W_{\text{LR}}$ image.

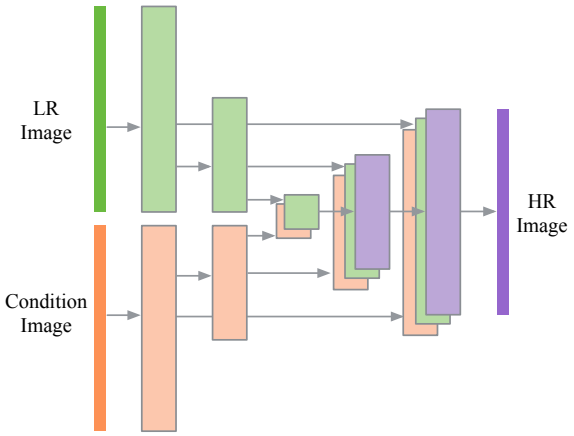


Figure 4: Dual-path U-Net. There are skip connections between the mirrored layers in two encoders and a decoder.

Fine Image Generator with Skips. Similar to the coarse image generation module, the fine image generator also contains two Siamese-like encoders and a decoder. The encoder consists of several convolutional layers to down-sample the image to a $M_h \times 1 \times 1$ tensor. Several de-convolutional layers are then used to up-sample the bottleneck tensor to $W_{\text{HR}} \times W_{\text{HR}}$.

Since the mapping from low resolution image to high resolution image can be seen as a conditional image translation problem, they only differ in surface appearance, but both are rendered under the same underlying structure. Therefore, the shape information can be shared between the LR and HR images. Besides, the low-level information of the conditioned image will also provide rich guidance when translating the LR image to the HR image. It would be desirable to shuttle these two kinds of information directly across the net. Inspired by the work of "U-Net" [24] and image-to-image translation [10], we add skip connections between the LR image encoder and the HR image decoder, and between the conditioned image encoder and the HR image decoder, simultaneously, which are illustrated in Fig. 4. With these skip connections, the decoder up-samples the encoded tensor to the high resolution image with the target view by several de-convolution layers.

Conditional Discriminator. The generated high resolution image $I^*_{v_j}$ and the ground-truth target image I_{v_j} are concatenated with the conditioned image I_{v_i} to form the negative pair and positive pair, respectively. These two kinds of image pairs are passed to the conditional discriminator and the fine image generator is trained adversarially.

3.6 Implementation Details

For the coarse image generator, the encoder network contains 6 convolution layers followed by 1 fully-connected layer. The convolution layers have 64, 128, 256, 256, 256 and 1024 channels with filter size of 5×5, 5×5, 5×5, 3×3, 3×3 and 4×4, respectively. The

fully-connected layer has 1024 neurons. M_l and W_{LR} are set to 1024 and 64, respectively. The representations of the target image and the condition image and the embedding of the target view are concatenated and transformed to the latent variable by a fully-connected layer with 1024 neurons. The decoder network consists of 1 fully-connected layer with 256×8×8 neurons, followed by 6 de-convolution layers with 2×2 up-sampling, which have 256, 256, 256, 128, 64 and 3 channels with filter size of 3×3, 5×5, 5×5, 5×5, 5×5 and 5×5, respectively.

For the fine image generation module, the encoder network contains 7 convolution layers with 64, 128, 256, 512, 512, 512, 512 channels, respectively. M_h is set to 512. The decoder network consists of 7 de-convolution layers with 512, 512, 512, 256, 128, 64 and 3 channels, respectively. The filter size of encoder network and decoder network is 4×4 and the stride is 2. The conditional discriminator consists of 5 convolution layers, which have 64, 128, 256, 512 and 1 channel(s), respectively, with filer size of 4×4 and stride of 2, 2, 2, 1, 1, respectively. W_{HR} is set to 128.

For training, the coarse image generator is first trained for 500 epochs. With the generated low resolution image and the conditioned image, the fine image generator and the conditional discriminator are then iteratively trained for another 500 epochs. All networks are trained using ADAM solvers with batch size of 32 and an initial learning rate of 0.0003.

4 EXPERIMENT

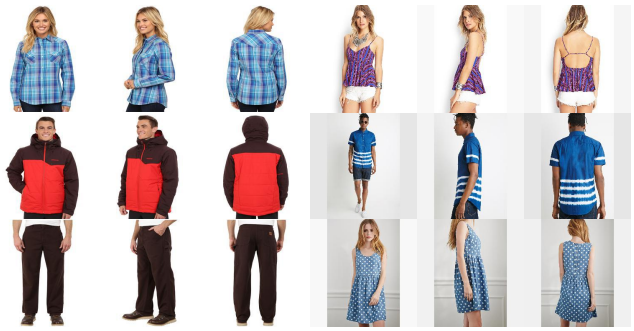
To verify the effectiveness of the proposed VariGANs model, extensive quantitative and qualitative evaluations have been conducted. In addition to the performance comparison with state-of-the-art models, we do ablation studies to investigate the design and important components of our proposed VariGANs.

4.1 Datasets and Evaluation Metrics

Datasets. Experiments are conducted on MVC [14] and DeepFashion [15] datasets, which contain a huge number of clothing images with different views. MVC[1] contains 36,323 clothing items. Most of them have 4 views, *i.e.*, front, left side, right side and back side. DeepFashion[2] contains 8,697 clothing items with 4 views, *i.e.*front, side (left or right), back and full body. Example images from the two datasets are demonstrated in Fig. 5. We can see that the view and scale variance of images from DeepFashion is much larger than those in MVC. The total number of images in DeepFashion is also smaller than MVC. The high variance and the limited number of training samples bring great difficulties for multi-view image generation on DeepFashion.

To give a consistent task specification of multi-view image generation on the two datasets, we define that the view set contains the front, side and back view. Two generation goals are considered: (1) to generate the side view and back view images conditioned on the front view image; (2) to generate the front view and back view images from side view image. These two scenarios are most popular in real life. We split the MVC dataset into the training set with 33,323 groups of images and the testing set with 3,000 groups of images. Each group contains three views of clothing images.

[1]http://mvc-datasets.github.io
[2]http://mmlab.ie.cuhk.edu.hk/projects/DeepFashion.html



(a) MVC (b) DeepFashion

Figure 5: Examples of multi-view images from MVC [14] and DeepFashion [15], respectively.

The training set of DeepFashion dataset consists of 7,897 groups of clothing images, and there are 800 groups of images in the testing set.

Evaluation Metrics. In previous literature on image generation [10, 33, 34], the performance is usually evaluated by human subjects, which is subjective and time-consuming. Instead of quantitative evaluation by user study, we firstly measure a pixel-level dissimilarity by Root Mean Square Error (RMSE) between a generated image and a target image over the test set, since the ground-truth images with target views are provided in the datasets. The smaller value of RMSE indicates more similarity between the generated image and target image. Besides, the Structural Similarity (SSIM) [28] is also adopted to measure the similarity between the generated image and ground truth image. It is widely used in many other image generation works, e.g., [19, 33]. In essence, the similarity measures the quality of the result. It can faithfully reflect the similarity of two images regardless of the light condition or small pose variance, which models the perceived changes in the structural information of the images.

The SSIM between two images I_x and I_y is defined as:

$$\text{SSIM}(I_x, I_y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)},$$

where x and y are the generated image and the ground-truth image, respectively. μ and σ are the average and variance of the image. c_1 and c_2 are two variables to stabilize the division, which are determined by the dynamic range of the images.

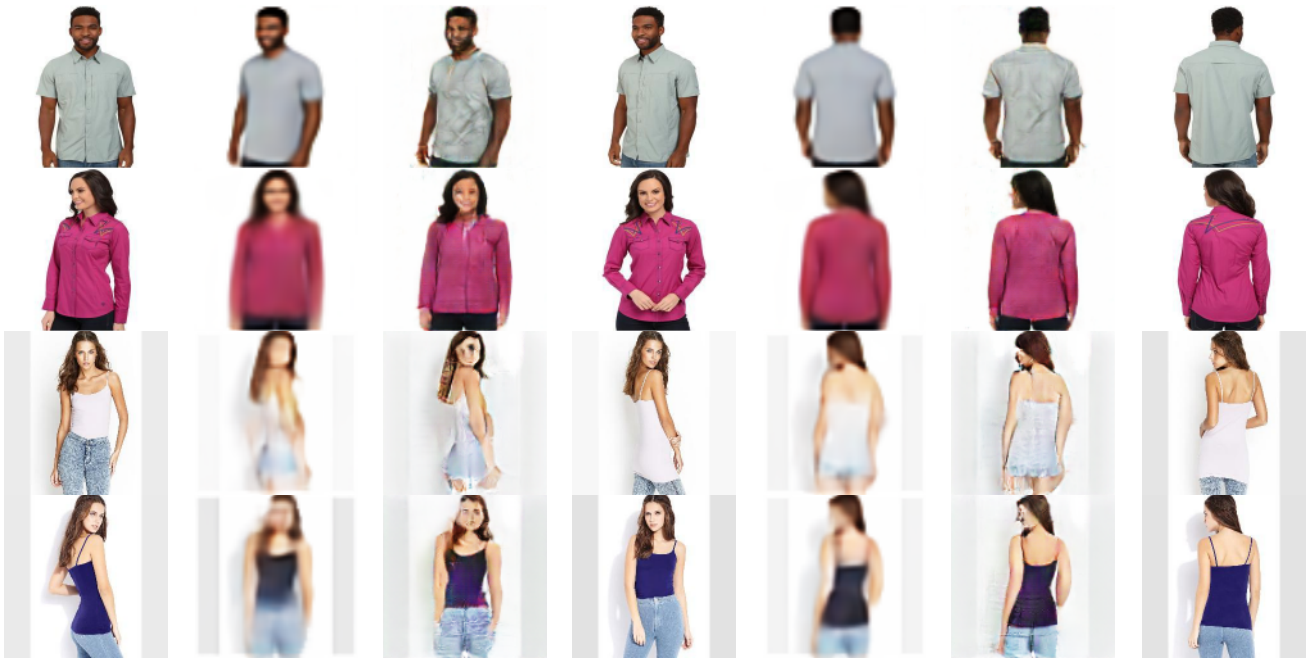
Another popular evaluation metric which is widely adopted to evaluate generative models is the Inception Score [25]. It expects the model to generate images contain meaningful objects. Moreover, the generated images should have high diversity. However, in our problem setting, the diversity of the target image is not the main concern, since the generated image and the condition image should only differs in the view point.

4.2 Experimental Results and Analysis

We compare the performance with two state-of-the-art image generation models: Conditional VAE (cVAE) [26] and Conditional GANs (cGANs) [16] on MVC and DeepFashion datasets. The cVAE has a similar architecture with the coarse image generator of our VariGANs. It has one

(a) Input (b) cVAE (c) cGANs (d) VariGANs (e) GT

Figure 6: Example results of the proposed method and the state-of-the-arts approaches. (a) are input images with front view, (b) and (c) are the results generated by cVAE and cGANs respectively. (d) demonstrate the coarse and fine images generated by our VariGANs.



(a) Input Images (b) Coarse Images (c) Fine Images (d) GT Images (e) Coarse Images (f) Fine Images (g) GT Images

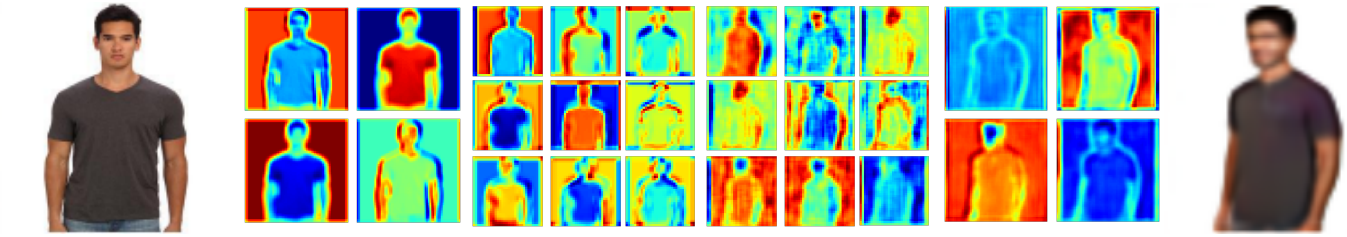
Figure 7: Multi-view images generated by our VAriGANs. The first and last two rows demonstrate example results generated from MVC and DeepFashion datasets, respectively. The images are generated from coarse to fine conditioned on the input images of different views.

Table 1: Performance comparison of the proposed method with the state-of-the-arts methods.

Methods	RMSE ↓		SSIM ↑	
	MVC	DF	MVC	DF
cVAE [26]	0.19 ± .05	0.22 ± .05	0.66 ± .09	0.58 ± .08
cGANs [16]	0.17 ± .04	0.20 ± .05	0.69 ± .10	0.59 ± .08
VariGANs	**0.14** ± .04	**0.18** ± .05	**0.70** ± .10	**0.62** ± .08

more convolution layer in the encoder and one more de-convolution layer in the decoder, which directly generate the HR Image. The cGANs have one encoder network to encode the conditioned image and one word embedding layer to transform the view to the vector. The encoded conditioned image and the view embedding are concatenated and fed into the decoder to generate the HR image.

The performance comparison is listed in Table 1. We can see that cVAE has the worst performance on both datasets, while cGANs improves the performance compared to cVAE. Our proposed VariGANs outperform these baselines on both datasets, which indicate



(a) (b) (c) (d) (e) (f)

Figure 8: Visualization of feature maps in the first two convolution layers ((b) & (c)) in the encoder of coarse image generation and the last two de-convolution layers ((d) & (e)) in the decoder of coarse image generation. Our model learns how to transform the image into the desired view. (a) and (f) are the input image and the generated LR image, respectively.

(a) (b) (c) (d) (e) (f) (g)

Figure 9: Generated images with different variants of our proposed method. (b), (c), (d) and (e) are the results of the model without V, Dual-path U-Net, ℓ_1 loss and conditional discriminator, respectively. (f) shows the results generated by VariGANs. (a) and (g) are the input images and the ground truth image.

that the proposed method is able to generate more realistic images conditioned on the single-view image and the target view.

Some representative examples generated by the state-of-the-arts methods and the proposed method are illustrated in Fig. 6. It can be seen that the samples generated by cVAE are blurry, and the color is not correct. However, it correctly draws the general shape of the person and the clothes in the side view. The images generated by cGANs are more realistic with more details, but present severe artifacts. Some of the generated images look unrealistic, such as the example in the second row. The low resolution image generated by the coarse image generator of our proposed VariGANs presents better shape and contour than cVAE, beneficial from a more reasonable target setting for this phase (*i.e.* only generating LR images). Besides, the generated LR image looks more natural than those generated by other baselines, in terms of the view point, shape and contour. Finally, the fine image generator fills correct color and adds richer and realistic texture to the LR image.

More examples generated by our VariGANs associated with coarse, fine and ground truth images are demonstrated in Fig. 7. The first two rows are from MVC dataset, while the others are from DeepFashion dataset. The first and third rows show the generated side and back view images from the front view. Given the side view image, the second and fourth rows demonstrate the generated front and back view images. From Fig. 7, we can see that the generated coarse images have the right view based on the conditioned image. The details are reasonably added to the coarse images with the fine image generation module. The results also demonstrate that the generated images need not be the same as the ground-truth image. There may be pose variance in the generated images like the generated front view image of the second example. Note that the proposed model focuses on clothes generation and does not consider humans in the image. Besides, some blocky artifacts can be observed in some examples, In the future, we will explore how to remove such artifacts by adopting more complicated models to generate sharper details. Nevertheless, current results present sufficient details about novel views for users.

Figure 10: Images generated from different z and same conditions. The images in first column are condition images with front view, and the rest columns contains three different images generated from different z with side view.

Generation Results of Different z. For the coarse image generator, we aim to generate target view images with reasonable variance and our model indeed performs this well. Diverse results which capture shape and coarse details with large probability are generated. We present two groups of examples sampled from different z but the same condition image and target view in in Fig. 10. The images in first column are condition images with front view, and the rest columns contains three different images generated from different z with side view.

Visualization of the Feature Maps. To provide a deeper insight to the mechanism of the multi-view image generation in the proposed model, we also visualize the feature maps of the first two convolution layers in the encoder of coarse image generation and their corresponding de-convolution layers in the decoder (*i.e.*the last two), as shown in Fig. 8. The visualization demonstrates that the model learns how to change the view of different parts of the image. From the visualization results, we can observe that the generated feature maps effectively capture the transition of view angles and the counterparts from another view.

4.3 Ablation Study

In this subsection, we analyze the effectiveness of the components in our proposed model to further validate the design of our model.

The Variational Inference (w/o V).. To investigate the role of variational inference in our proposed VariGANs, we first conduct the experiment which replaces the variational inference in coarse image generator with GANs. In this implementation of our model, the LR image is firstly generated by GANs conditioned on the input single-view image and the target view, and then used by fine image generator to generate the HR image. This variant of our model is similar to StackGAN [34], which synthesizes images from textual description in two stages.

The Dual-path U-Net (w/o U-Net). To verify the effectiveness of the Dual-path U-Net, we implement the proposed model without the skip connections. The low resolution image and the conditioned image go through the siamese encoders in the fine image generation module until a bottle-neck and the outputs of the encoders are concatenated and fed into the decoder networks.

Reconstruction Loss (w/o ℓ_1). The traditional reconstruction loss, *i.e.*ℓ_1 loss is important in our task to generate plausible images, which is adopted in many previous research. To prove that we also conduct experiments without ℓ_1 loss.

The Conditional Discriminator (w/o cGANs). Finally, we train our model without the conditional discriminator in the fine image generation, *i.e.*only the generated images and ground truth images are used to train the discriminator. In this way, the discriminator is designed only to distinguish the image is real or generated, not considering the condition image.

The effect of individual components of VariGANs on MVC and DeepFashion is listed in Table 2. It can be seen that the performance is dropped after removing or replacing any component of our model. As shown in the first line in Table 2, stacking two GANs together decreases the performance, since the generated LR images do not have good structure, and therefore make the HR images generated based on the LR images worse. As shown in the second row of Table 2, without the skip connections, both the performances on RMSE and SSIM of the generated images drop dramatically comparing other components, indicating the importance of the lower features of both the conditioned image and the LR image. The direct connections to the low level feature of the LR image provide strong shape and view information, and the connections to the low level feature of the conditioned image give clues about the color and details that need displayed in the generated image. Since the ℓ_1 loss provides strong supervision to generate the image with complete structure and correct view, removing it will also greatly decrease the performance of the model as show in the third line in Table 2. Not concatenating the generated images and the target image with condition image slightly decrease the performance of our model comparing to other components as reported in the forth line in Table 2. It indicates that although the LR image generated by the coarse image generation module already has good structure and correct target view, the fine image generation module can further refine it and generate more reasonable results by feeding the generated image and the input image with the condition image to the discriminator.

Table 2: The effect of individual components of VariGANs.

Methods	RMSE ↓		SSIM ↑	
	MVC	DF	MVC	DF
w/o V	0.18 ± .04	0.22 ± .05	0.69 ± .11	0.59 ± .07
w/o U-Net	0.25 ± .03	0.32 ± .05	0.56 ± .08	0.53 ± .07
w/o ℓ_1	0.22 ± .04	0.25 ± .05	0.58 ± .09	0.49 ± .06
w/o cDisc	0.19 ± .04	0.23 ± .04	0.66 ± .09	0.55 ± .09
VariGANs	**0.14** ± .04	**0.18** ± .05	**0.70** ± .10	**0.62** ± .08

The images generated by different variants of our VariGANs are illustrated in Fig. 9. Conditioned on the LR image generated by GANs, the result in Fig. 9(b) displays relative good shape and right texture. However, some parts are missing in the generated image, *i.e.*, the left hand. The result without the dual-path U-Net has incomplete areas and unnatural colors, as shown in Fig. 9(c). Without ℓ_1 loss, the detailed texture is not well learned, such as the upper part of the cloth in Fig. 9(d). VariGANs without conditional discriminator generate comparative result (Fig. 9(e)) as VariGANs (Fig. 9(f)).

5 CONCLUSION

In this paper, we propose a Variational Generative Adversarial Networks (VariGANs) for synthesizing realistic clothing images with different views as input image. The proposed method enhances the GANs with variational inference, which generate image from coarse to fine. Specifically, the coarse image generator first produces the basic shape of the object with the target view. The fine image generator then fills the details into the coarse image and corrects the defects. Comprehensive experiments demonstrate that our model can generate more plausible results than the state-of-the-arts methods. The ablation studies also verify the importance of each component in the proposed VariGANs.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (Grant No: 61772436), Sichuan Science and Technology Innovation Seedling Fund (2017RZ0015), and the Fundamental Research Funds for the Central Universities. Bo Zhao was supported by China Scholarship Council (Grant No. 201507000032). Jiashi Feng was partially supported by NUS IDS R-263-000-C67-646, ECRA R-263-000-C87-133 and MOE Tier-II R-263-000-D17-112. Zhi-Qi Cheng was partially supported by Sichuan Science and Technology Innovation Seedling Fund (Grant No. 2017018, Grant No. 2017020). We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] Bo Chang, Qiong Zhang, Shenyi Pan, and Lili Meng. 2018. Generating Handwritten Chinese Characters using CycleGAN. In *WACV*.

[2] Tao Chen, Zhe Zhu, Ariel Shamir, Shi-Min Hu, and Daniel Cohen-Or. 2013. 3-Sweep: Extracting Editable Objects from a Single Photo. *ACM Transactions on Graphics* (2013).

[3] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2016. InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets. *arXiv:1606.03657* (2016).

[4] Jeff Donahue, Philipp Krähenbähl, and Trevor Darrell. 2016. Adversarial Feature Learning. *arXiv:1605.09782* (2016).

[5] Alexey Dosovitskiy, Jost Tobias Springenberg, Maxim Tatarchenko, and Thomas Brox. 2015. Learning to Generate Chairs, Tables and Cars with Convolutional Networks. In *CVPR*.

[6] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. 2017. Adversarially Learned Inference. *arXiv:1606.00704* (2017).

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *NIPS*.

[8] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. 2015. DRAW: A Recurrent Neural Network For Image Generation. In *ICML*.

[9] Geoffrey E. Hinton, Alex Krizhevsky, and Sida D. Wang. 2011. Transforming Auto-Encoders. In *ICANN*.

[10] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2016. Image-to-Image Translation with Conditional Adversarial Networks. *arXiv:1611.07004* (2016).

[11] Natasha Kholgade, Tomas Simon, Alexei Efros, and Yaser Sheikh. 2014. 3D Object Manipulation in a Single Photograph using Stock 3D Models. *ACM Transactions on Graphics* (2014).

[12] Diederik P Kingma and Max Welling. 2014. Auto-encoding Variational Bayes. In *ICLR*.

[13] Tejas D. Kulkarni, Will Whitney, Pushmeet Kohli, and Joshua B. Tenenbaum. 2015. Deep Convolutional Inverse Graphics Network. In *NIPS*.

[14] Kuan-Hsien Liu, Ting-Yen Chen, and Chu-Song Chen. 2016. MVC: A Dataset for View-Invariant Clothing Retrieval and Attribute Prediction. In *ICMR*.

[15] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. 2016. DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations. In *CVPR*.

[16] Mehdi Mirza and Simon Osindero. 2014. Conditional Generative Adversarial Nets. *arXiv:1411.1784* (2014).

[17] Augustus Odena. 2016. Semi-Supervised Learning with Generative Adversarial Networks. *arXiv:1606.01583* (2016).

[18] Augustus Odena, Christopher Olah, and Jonathon Shlens. 2016. Conditional Image Synthesis With Auxiliary Classifier GANs. *arXiv:1610.09585* (2016).

[19] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C. Berg. 2017. Transformation-Grounded Image Generation Network for Novel 3D View Synthesis. In *CVPR*.

[20] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. 2016. Context Encoders: Feature Learning by Inpainting. In *CVPR*.

[21] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. 2016. Generative Adversarial Text-to-Image Synthesis. In *ICML*.

[22] Danilo Jimenez Rezende, S. M. Ali Eslami, Shakir Mohamed, Peter W. Battaglia, Max Jaderberg, and Nicolas Heess. 2016. Unsupervised Learning of 3D Structure from Images. In *NIPS*.

[23] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In *ICML*.

[24] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *MICCAI*.

[25] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. 2016. Improved Techniques for Training GANs. *arXiv:1606.03498* (2016).

[26] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning Structured Output Representation using Deep Conditional Generative Models. In *NIPS*.

[27] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. 2017. Adversarial Cross-Modal Retrieval. In *ACM MM*. 154–162.

[28] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612.

[29] Jiajun Wu, Tianfan Xue, Joseph J. Lim, Yuandong Tian, Joshua B. Tenenbaum, Antonio Torralba, and William T. Freeman. 2016. Single Image 3D Interpreter Network. In *ECCV*.

[30] Wei Xiong, Wenhan Luo, Lin Ma, Wei Liu, and Jiebo Luo. 2018. Learning to Generate Time-Lapse Videos Using Multi-Stage Dynamic Generative Adversarial Networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[31] Xinchen Yan, Jimei Yang, Kihyuk Sohn, and Honglak Lee. 2016. Attribute2Image: Conditional Image Generation from Visual Attributes. In *ECCV*.

[32] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. 2016. Perspective Transformer Nets: Learning Single-View 3D Object Reconstruction without 3D Supervision. In *NIPS*.

[33] Donggeun Yoo, Namil Kim, Sunggyun Park, Anthony S. Paek, and In So Kweon. 2016. Pixel-Level Domain Transfer. *arXiv:1603.07442* (2016).

[34] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaolei Huang, Xiaogang Wang, and Dimitris Metaxas. 2016. StackGAN: Text to Photo-realistic Image Synthesis with Stacked Generative Adversarial Networks. *arXiv:1612.03242* (2016).

[35] Bo Zhao, Bo Chang, Zequn Jie, and Leonid Sigal. 2018. Modular Generative Adversarial Networks. In *ECCV*.

[36] Youyi Zheng, Xiang Chen, Ming-Ming Cheng, Kun Zhou, Shi-Min Hu, and Niloy J. Mitra. 2012. Interactive images: cuboid proxies for smart image manipulation. *ACM Transactions on Graphics* (2012).

[37] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A. Efros. 2016. View Synthesis by Appearance Flow. In *ECCV*.

[38] Yipin Zhou and Tamara L. Berg. 2016. Learning Temporal Transformations From Time-Lapse Videos. In *ECCV*.

[39] Jun-Yan Zhu, Philipp Krähenbühl, Eli Shechtman, and Alexei A. Efros. 2016. Generative Visual Manipulation on the Natural Image Manifold. In *ECCV*.