Generative Modelling of Semantic Segmentation Data in the Fashion Domain

Marie Korneliusson
Chalmers university of technology
fiakorneliusson@gmail.com

John Martinsson
RISE Research institutes of Sweden
john.martinsson@ri.se

Olof Mogren
RISE Research institutes of Sweden
olof.mogren@ri.se

Abstract

In this work, we propose a method to generatively model the joint distribution of images and corresponding semantic segmentation maps using generative adversarial networks. We extend the Style-GAN architecture by iteratively growing the network during training, to add new output channels that model the semantic segmentation maps. We train the proposed method on a large dataset of fashion images and our experimental evaluation shows that the model produces samples that are coherent and plausible with semantic segmentation maps that closely match the semantics in the image.

1. Introduction

Semantic segmentation is the task of assigning a semantic class to each pixel of an image. The problem has recently been successfully attacked with fully convolutional neural networks trained using large datasets of manually tagged images. The training datasets for semantic segmentation is thus required to contain a large number of images with pixel-level labels of the semantic classes of interest; labels referred to as segmentation maps. In the fashion domain, the classes of interest may be different kinds of garments and accessories.

In this paper, we model the joint probability of images together with their semantic segmentation maps. To this end, we propose a generative adversarial network (GAN) [1], based on the Style-GAN architecture [12]. In contrast to previous GAN-based approaches, the proposed model is designed to produce samples with one output channel per segmentation class in addition to the standard three colour channels. A sample image with segmentation maps generated with the proposed method can be seen in figure 1.

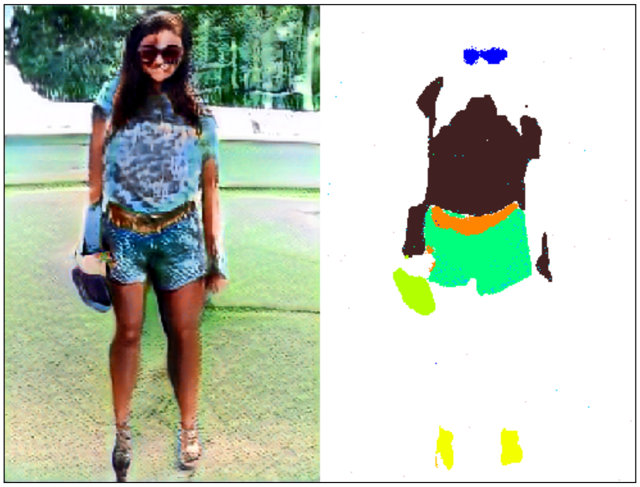

Figure 1: Sample image with segmentation map generated using the proposed method. The normal colour channels are displayed to the left, while the generated corresponding semantic segmentation maps are displayed to the right.

2. Generative semantic segmentation

This work considers the problem of generatively modelling of the joint distribution over images and semantic segmentation maps. We propose a model that extends the Style-GAN model introduced in [12].

Style-GAN. The generator in the original Style-GAN architecture is composed of two main parts – the *mapping network*: a multi-layer perceptron with 8 layers, and the *synthesis network*: a convolutional neural network (see figure 2). The convolutional synthesis network is grown during *progressive training* which means that for each new phase in the training, a new block of layers are added to the generator resulting in output samples with doubled sizes (see section 4).

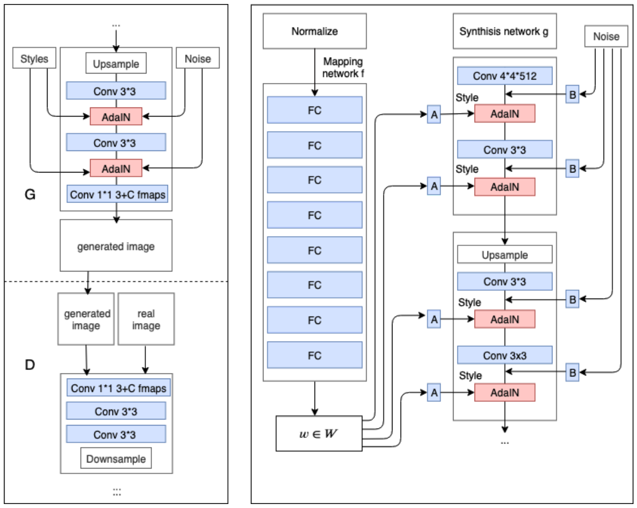

Figure 2: The architecture of Style-GAN and Style-C-GAN. The left figures shows the layers in each building block, where the upper part shows the generator and the lower the discriminator. The right figure shows the entire architecture of the Style GAN generator network.

In Style-GAN the latent code z does not directly control the synthesis network, in contrast to previous GAN architectures. Instead z is fed through the mapping network, resulting in a representation w. A learned affine transformation turns w into the styles $y = (y_s, y_b)$. The style y controls the output of each convolutional layer in the synthesis network through the adaptive instance normalization operation (AdaIN) [8]. This operation aligns mean and variance of each feature map to match those of the styles y. In addition, noise is added by a learned scaling factor to each feature map in the convolutional layers. This addition of noise has been shown to increase the generator's ability to generate images with stochastic variation, such as placement of hairs, freckles etc [12].

Style-C-GAN. In this work we present Style-C-GAN, an architecture based on the layout of the original Style-GAN that is able to model the joint distribution over images and segmentation maps. To accomplish this, we add an extra set of C feature maps to the last convolutional layer (see figure 2) in the Style-GAN discriminator D and the generator G. Hence the number of feature maps in the last convolutional layer of each block in the Style-C-GAN discriminator and generator will now equal $3 + C$, i.e. the input and output is now an object of dimension $(W \times H \times (3 + C))$ that allocates 3 channels for the image and C channels for the semantic segmentation maps. Therefore $C = 13$ when generating all classes in ModaNet.

The Style-C-GAN is progressively grown from a Style-GAN network. After training the Style-GAN model with images without segmentation maps, we add one new output channel to the generator, which is initialized by copying the weights associated with the first colour channel (red). This channel is then trained to generate binary segmentation maps (where 1 means the existence of some of the semantic classes, and 0 is the absence of such a class). After this initial binary map training, new channels are added so the total output channels will be $3+C$. The weights of the new channels are initialized by copying the weights from the newly trained binary segmentation channel. We then proceed to perform the final training with all C classes.

Style-C-GAN-2D. In the Style-C-GAN-2D setting, the generator is trained by alternating between the original Style-GAN discriminator and the Style-C-GAN discriminator. This allows us to train the colour channels of Style-C-GAN generator using a discriminator trained specifically for RGB images.

3. Previous work

Generative adversarial networks were introduced by [1], and have proven to be a very useful way to model image distributions. Using GANs to synthesize fashion images have recently become an active field of study [9, 10, 17, 4, 2, 14, 3, 7].

Recent approaches for fashion image synthesis have studied how to synthesize parts of a fashion image by, e.g., swapping fashion articles on the person in the image [9, 4], using natural language to describe how to "redress" the person in the image [17, 2], swap the clothing of two persons in two different images [14], and generating suggestions on minimal edits to improve "fashionability" [7].

Perhaps the most closely related work to ours is the work by [3] which considers the problem of image inpainting. A structured part of a semantic segmentation map is removed, and a generator is used to generate a new semantically coherent part for the removed part of the semantic map. The new semantic segmentation map is then used in a conditional generative adversarial network to generate a corresponding color image. The result is a coherent pair of color image and semantic segmentation map, but with other semantics in part of the image.

In this work we instead consider the problem of modelling the joint distribution of color images and semantic segmentation maps directly. To our knowledge, this is the first work to demonstrate generative modelling of the joint distribution of images and semantic segmentation maps.

4. Experiments

In this section we present the datasets for evaluation, the evaluation methods used, and the implementation and training details for the model.

Dataset. We train the models using ModaNet [16], which includes 55,176 fully annotated 400x600 pixel fashion images. Each image contains one person with diverse human poses. The dataset contain images with partial views of the human model. Each image has a corresponding ground truth semantic segmentation map, i.e each pixel is labeled into one of the 13 predefined classes. Each class represents a unique fashion attribute.

The dataset is highly unbalanced. The most frequent class, footwear, is roughly ten times more frequent than the least frequent class, scarfs and ties.

Only 52,346 images were available for download when the experiments were conducted. From this set we randomly chose 45,346 images as the training set, 2,000 images as the validation set, and 5,000 images as the test set.

Pre-processing. The RGB images are resized while keeping the aspect ratio. This is followed by normalizing each channel to the range $[-1, 1]$ as in the original Style-GAN setup [12].

The semantic segmentation maps used for training the Style-C-GAN was preprocessed in three steps. First, we resize the maps in the same way as the RGB images. Second, we shift the binary values in the maps such that they are symmetric around zero. Third, we add Gaussian noise with mean 0 and standard deviation 0.01 to each map.

Implementation and training details. We use the code published by NVIDIA[1] as the starting point for the implementation of our network. Style-GAN uses progressive growing [11, 12] and we use a resolution dependent learning rate and mini-batch size ranging from 0.001 to 0.002 and 128 to 16, for resolution 8 by 8 pixels to the target resolution 512 by 512 pixels, respectively. The discriminator and generator networks are optimized using Adam [13], with an exponential decay $\beta_1 = 0.0$, and $\beta_2 = 0.99$. Otherwise, we use the same hyper-parameters as in [12]. The weights are randomly initialized using the He Normal initializer [5].

All models are trained using 4 GeForce GTX 1080 Ti graphics cards each, and the entire training procedure takes 1-2 weeks for each model. We do all model selection based on the best recorded FID using the validation dataset.

Evaluation methods. We use two standard metrics to measure the quality of the generated images: (i) Inception score (IS) [15], and (ii) Frechet Inception distance (FID) [6]. The Inception score favors images that have a high class likelihood under the Inception model and a marginal class distribution with high entropy; higher is better. The Frechet Inception distance estimates the distance between generated and real data samples; lower is better. The original Style-GAN model is used as a baseline in the quantitative comparison. The same initialization and training data was used, but for the baseline, only the RGB images can be used for training.

5. Results

Quantitative results. In table 1 we note that Style-GAN performs best both regarding IS and FID, implying that the RGB image quality of the plain Style-GAN generator have a higher quality than our proposed models. That is, generating both images and semantic segmentation maps seem to affect image quality negatively.

Table 1: The IS and FID for each model. We present IS as the mean of five batches using 10K generated images in each batch, and FID using the test set and 50K generated images.

Model	IS	FID
Style-GAN	5.06 ± 0.11	12.34
Style-C-GAN	4.79 ± 0.12	25.10
Style-C-GAN-2D	4.69 ± 0.12	20.97

Latent space traversal. In figure 3 we show the output response from the generator when traversing the latent space. We interpolate between two latent codes, and generate images using Style-C-GAN. We can see that both poses, humans, backgrounds and clothes changes as we interpolate between latent codes. We also note that the same human continuously appears in different poses and different clothes, and that the generated semantics of the scene correspond well with these changes. In the real dataset we can not continuously interpolate between different images, the result therefore indicates that the generator generates images that do not exist in the real dataset.

6. Discussion and conclusions

In this work, we have demonstrated a GAN architecture (Style-C-GAN) that successfully model the joint distribution over images and the corresponding semantic segmentation maps. Furthermore, we have shown that using two different discriminator architectures (Style-C-GAN-2D), we can perform a semi-supervised training setup, leveraging both tagged and untagged training data. This further improved the quality of generated images. The nature of GANs allows us to traverse the latent space, giving some control over the generated images. The results implies that we can generate data that may be used to enhance training of discriminative semantic segmentation models by augmenting the training data with generated samples. This could also allow us to relieve problems with class imbalances.

[1] https://github.com/NVlabs/stylegan

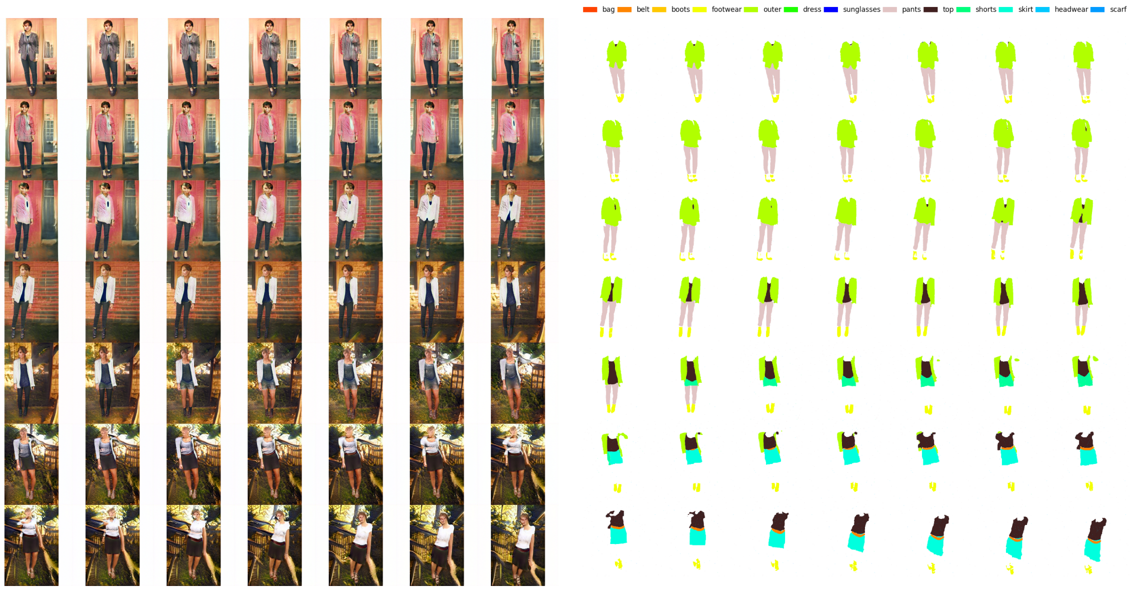

Figure 3: Traversing the latent space. Interpolation between two randomly chosen latent codes, resulting samples generated by Style-C-GAN. *Left:* the generated images. *Right:* the generated semantic segmentation maps.

References

[1] 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*, pages 2672–2680, 2014. 1, 2

[2] M. Gunel, E. Erdem, and A. Erdem. Language guided fashion image manipulation with feature-wise transformations. In *First Workshop on Computer Vision in Art, Fashion and Design) – in conjunction with ECCV 2018*, 2018. 2

[3] X. Han, Z. Wu, W. Huang, M. R. Scott, and L. S. Davis. Compatible and diverse fashion image inpainting. *CoRR*, abs/1902.01096, 2019. 2

[4] X. Han, Z. Wu, Z. Wu, R. Yu, and L. S. Davis. Viton: An image-based virtual try-on network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[5] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015. 3

[6] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems 30*. 2017. 3

[7] W. Hsiao, I. Katsman, C. Wu, D. Parikh, and K. Grauman. Fashion++: Minimal edits for outfit improvement. *CoRR*, abs/1904.09261, 2019. 2

[8] X. Huang and S. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1501–1510, 2017. 2

[9] N. Jetchev and U. Bergmann. The conditional analogy gan: Swapping fashion articles on people images. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017. 2

[10] S. Jiang and Y. Fu. Fashion style generator. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 3721–3727, 2017. 2

[11] T. Karras, T. Aila, S. Laine, and J. Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *ICLR*. OpenReview.net, 2018. 3

[12] T. Karras, S. Laine, and T. Aila. A style-based generator architecture for generative adversarial networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 1, 2, 3

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. dec 2014. 3

[14] A. Raj, P. Sangkloy, H. Chang, J. Lu, D. Ceylan, and J. Hays. Swapnet: Garment transfer in single view images. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[15] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems 29*. 2016. 3

[16] S. Zheng, F. Yang, M. H. Kiapour, and R. Piramuthu. Modanet: A large-scale street fashion dataset with polygon annotations. jul 2018. 3

[17] S. Zhu, R. Urtasun, S. Fidler, D. Lin, and C. Change Loy. Be your own prada: Fashion synthesis with structural coherence. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 2