Description-aware Fashion Image Inpainting with Convolutional Neural Networks in Coarse-to-Fine Manner

Furkan Kınlı
Özyeğin University
Department of Computer Science
Çekmeköy/İstanbul
furkan.kinli@ozyegin.edu.tr

Barış Özcan
Özyeğin University
Department of Computer Science
Çekmeköy/İstanbul
baris.ozcan.10097@ozu.edu.tr

Furkan Kıraç
Özyeğin University
Department of Computer Science
Çekmeköy/İstanbul
furkan.kirac@ozyegin.edu.tr

ABSTRACT
Inpainting a particular missing region in an image is a challenging vision task, and promising improvements on this task have been achieved with the help of the recent developments in vision-related deep learning studies. Although it may have a direct impact on the decisions of AI-based fashion analysis systems, a limited number of studies for image inpainting have been done in fashion domain, so far. In this study, we propose a multi-modal generative deep learning approach for filling the missing parts in fashion images by constraining visual features with textual features extracted from image descriptions. Our model is composed of four main blocks which can be introduced as textual feature extractor, coarse image generator guided by textual features, fine image generator enhancing the coarse output, and lastly global and local discriminators improving refined outputs. Several experiments conducted on FashionGen dataset with different combination of neural network components show that our multi-modal approach is able to generate visually plausible patches to fill the missing parts in the images.

CCS Concepts
• **Computing methodologies ~ Artificial intelligence ~ Computer vision ~ Computer vision problems ~ Reconstruction**

• **Computing methodologies ~ Machine learning ~ Machine learning approaches ~ Neural networks**

Keywords
deep learning; image inpainting; image reconstruction; fashion analysis; generative learning; multi-modal neural networks

1. INTRODUCTION
Numerous studies on generating photo-realistic images from a noise, or constraining the visual input by the other input sources have been done in recent years. Generating high-resolution realistic images is still a difficult task, although a number of studies show that VAEs [1, 2, 3] and GANs [4, 5, 6, 7] achieved promising results to generate photo-realistic images from scratch. The most challenging part of generating new images is that neural network models are not robust to the changes in the distribution of pixels among different samples. This leads to generate images that contain blurry components and some artefacts, especially in high-resolution cases. Image inpainting is one of the sub-tasks of image generation where a given image with a particular missing region, and this region needs to be filled with semantically and visually plausible contents (see Fig. 1). In fashion domain, although there are some numbers of studies on generating fashion images have been done in the literature, this particular task has not been investigated in detail, as well as in natural scene inpainting and face inpainting.

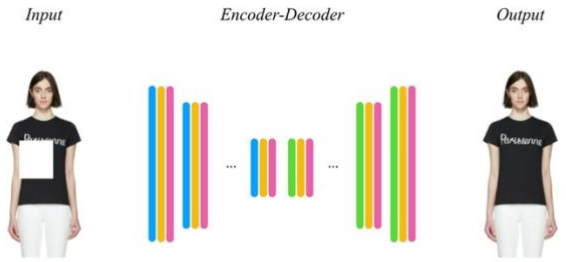

Figure 1. Filling the missing region with semantically and visually plausible contents by using an encoder-decoder network.

The main reason why we are interested in this problem is that achieving this task could help the fashion analysis systems getting help from AI-based solutions. For example, fashion trend analysis is an important topic that needs to provide some insights to the companies from the future fashion trends, and color tone analysis is one of the most valuable factors leading to change the current trends in fashion. In such systems, clothing images in the wild (from social media sources like Instagram and Twitter) have some missing or occluded parts, and to analyze these images in detail and correctly, the missing regions need to be completed in both plausible and exclusive way. Starting from this, inpainting the clothing images with deep learning-based solutions may be critical, and thus, it may have a significant impact on the performance of such AI-based fashion analysis systems.

ICCTA 2020, April 14–16, 2020, Antalya, Turkey

ACM ISBN 978-1-4503-7749-2/20/04…$15.00
https://doi.org/10.1145/3397125.3397155

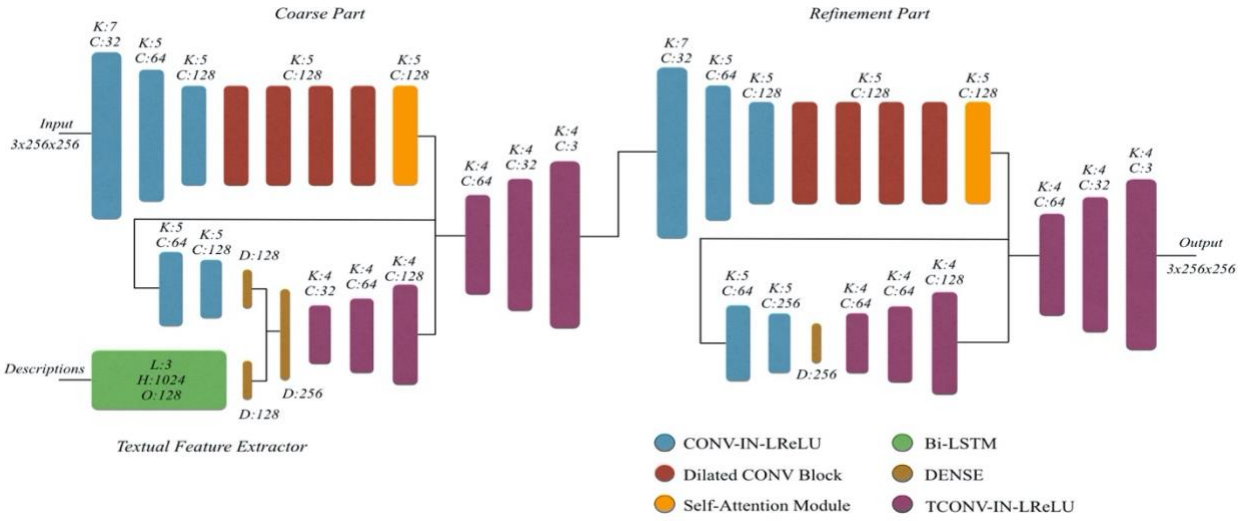

Figure 2. The overall architectural design. Left: Coarse Network, Right: Refinement Network. Each convolutional layer is skip-connected to corresponding decoder layer at the same level. Dilated convolutional blocks have a residual connection to the next one. In our experiments, we also use batch normalization and upsampling layers, instead instance normalization (IN) and transpose-convolutional layers, but we introduce a single model design with the best performance on validation.

Our main contributions are as follows:

1.) We propose a multi-modal generative model using both the images and their descriptions as knowledge resources, and trained it with an adversarial approach.

2.) We investigate the effects of different visual feature extraction blocks, image generation blocks and the objective functions on inpainting performance of our generative model.

3.) We present the qualitative and quantitative inpainting performances of our multi-modal generative model on FashionGen dataset [8].

2. RELATED WORKS

A large number of approaches have been proposed for image inpainting in different domains. In traditional approaches, progressively extending the pixels near to the boundaries of the missing parts is a method of searching the most similar patches to fill the missing parts [9, 10]. Simakov *et al.* [11] improves this idea to better capture non-stationary images by using bi-directional similarity synthesis method. While tree-based acceleration structures of memory [12] and randomized algorithms [13] are proposed to reduce the cost of finding suitable patches, others try to improve the quality of generated patches by employing local features such as image gradients and off-set statistics of patches [14, 15, 16]. Recently, researches on image inpainting solutions are overtaken by deep learning methods since they can achieve outstanding results in various vision tasks. To generate high resolution images, Iizuka *et al.* [17] improves Fully Convolutional Networks (FCNs) with both local and global consistency mechanisms. Specializing in rectangular missing parts, contextual attention mechanism [18] captures the long-range dependency between pixels, and improves the quality of generated parts. On the other hand, to achieve better results on free-formed missing parts, Partial convolution is proposed by Liu *et al.* [19], which only takes valid pixels into account, and infers the missing parts with a rule-based update step. All aforementioned image inpainting studies are in the domains of natural scene understanding, basic object reconstruction and face recovering. However, considering the practical advantages on the applications in fashion industry, we alter the main focus of inpainting tasks to fashion domain.

3. METHODOLOGY

In this study, we propose a multi-modal coarse-to-fine generative deep learning model for filling the missing parts in fashion images by utilizing their descriptions. Our model contains four main blocks for extracting textual and visual features, generating new images in both coarse and refined manner, and global and local discriminator networks to improve refined outputs. These blocks are introduced in this part with the implementation details. The overall architecture design can be seen on Figure 2.

3.1 Textual Feature Extractor

In the first module of our multi-modal generative deep learning model, textual features are extracted by 3-layer bi-directional LSTM module containing 1024 units in hidden layers followed by dropout and linear rectifier. The descriptions are, first, tokenized by spaCy English, including initializer, end of the sentence and unknown word tokens, and each word in the descriptions is represented as 32-dimensional vector. At the end, textual feature extractor module generates 128-dimensional feature embedding as an output that represents the descriptions of the images as weak features. This embedding is concatenated with the output of the encoder part of coarse network to constraint the latent space of visual features.

3.2 Coarse Network

Coarse network is basically a *U-Net-like* [20] auto-encoder that contains 6 convolutional blocks for the encoder part and 6 transpose-convolutional or upsampling blocks for the decoder part where corresponding encoder layers and decoder layers are skip-connected. All convolutional blocks, except the first one (*7x7*), have convolutional layers with *5x5* filters and different number of channels, followed by instance normalization or batch normalization, leaky or standard linear rectifiers and dropout. In addition to these, there are 4 dilated convolutional blocks (i.e. between third and fourth convolutional blocks) that contain 2 residual-connected convolutional layers with dilation of 2. The

Figure 3. Example images and their descriptions from FashionGen dataset [8].

output of the last dilated convolutional block is followed by a self-attention module to maintain long-term dependency on the output of dilated feature map. Using self-attention mechanism in a generative model is inspired by the idea of SAGAN [6], but used in encoder part of the network. After concatenating the visual features with the textual features as previously mentioned, the decoder blocks with stride of 2 are employed for upsampling the feature maps. Likewise, the output of third decoder block is concatenated with the output of self-attention module of the encoder part. At the end, the decoder part generates the output with the size of *3x256x256* as a new image. The objective functions used for training of coarse network as follows:

$$L_p = \sum_i smooth_{L1}(x_i - x_i') \tag{1}$$

$$L_s = \left\| G_j^{\phi}(x) - G_j^{\phi}(x') \right\|^2 \tag{2}$$

$$G_j^{\phi}(x)_{c,c'} = \frac{1}{C_j H_j W_j} \sum_i^H \sum_i^W \phi(x_i)_{h,w,c} \phi(x_i)_{h,w,c'} \tag{3}$$

$$L_{coarse} = \lambda_p L_p + \lambda_s L_s \tag{4}$$

where x_i and x_i' are the input and the generated image by our model. $G_j^{\phi}(x)$ represents Gram matrix of corresponding image, which computes the dot product of each feature in spatial dimension. λ_s and λ_p are scale coefficients of style loss L_s and pixel loss L_p, respectively.

3.3 Refinement Network

Similar to coarse network, there is a *U-Net-like* [20] auto-encoder design for the refinement network that contains 6 convolutional blocks for the encoder part, and 6 transpose-convolutional or bilinear upsampling blocks for the decoder part. All blocks in the encoder part has a skip-connection to the blocks in the corresponding level of decoder part. The components of encoder and decoder, the dilated convolutional blocks and self-attention module are identical with the coarse network, but we do not employ LSTM module to the refinement network since the main aim of this network is only to refine the coarse output, and there is no need to constrain the latent vector by textual features. The decoder part is also identical to the coarse network, and the filled images have a size of *3x256x256*. 4 different objective functions are used for training of the refinement network, that are pixel loss (see Equation 1), style loss (see Equation 2), global loss and local loss as introduced in Equation 5.

$$L_g = L_l = \max_D \mathbf{E}_x[D(x)] \tag{5}$$

$$L_{refine} = \lambda_p L_p + \lambda_s L_s + \lambda_g L_g + \lambda_l L_l \tag{6}$$

where $D(x)$ is the output of corresponding discriminator for realness of the images. λ_p, λ_s, λ_g and λ_l are the scale coefficients of corresponding objective function.

Since the generators utilizing only reconstruction loss basically try to map the latent space to the data space in order to make the output closer to the input in pixel-wise manner, the result images generally do not look like natural, and contain completely unnatural artefacts. This problem escalates in more complex domains. To overcome this, the network needs another objective function that reveal the properties of statistical knowledge in the images, so that the generator part can learn to generate more plausible outputs with less or no artefacts. In this study, we employ 2 different discriminator networks to contribute the inpainting performance of the refinement network by adversarial training of that the whole image and only the missing part are real or fake. Therefore, the outputs of both discriminators inferring for global and local images are included to the training of the refinement network to increase the quality of the filled part of the images.

4. EXPERIMENTAL DETAILS

4.1 Dataset

We conduct our experiments on FashionGen dataset [8] that contains fashion products collected from an online platform selling the luxury goods from independent designers. Each product is represented by an image, the description, its attributes, and relational information defined by professional designers. We pick 9 different categories (top, sweater, pant, jean, shirt, dress, short, skirt, coat) that fit better to our study purpose, and extract the instances belonging to these categories from the dataset. At this

point, our training set contains 200K training clothing images from corresponding categories and their descriptions, for each of validation and testing sets 30K clothing images and their descriptions. Some examples from FashionGen dataset can be seen in Figure 3. Moreover, we erase rectangular patches from the original images with a particular heuristic which ensures that each patch contains at least a few pixels of clothing object in order to generate our final training and test datasets for our task.

4.2 Experimental Setup

In our training, we use coarse-to-fine approach for our generative model to achieve inpainting task in fashion domain. As in the recent AutoEncoder-based generative networks, we use Adam optimizer [21] with β_1 = 0.5 and β_2 = 0.999, for coarse, refinement and discriminator networks with different learning rates, 0.002, 0.001 and 0.0001 respectively, and decaying the learning rate exponentially by 0.95 at the beginning of each epoch. We train our multi-modal generative network for 120.000 steps with batch size of 32. For both objection functions of coarse and refinement networks, the weight ratio of pixel loss to style loss, λ_p to λ_s, is adjusted to 1:5. For the objective function of the discriminator network, global loss λ_g and local loss λ_l are calculated with the ratio of 1:1. Lastly, the components of reconstruction loss (e.g. pixel loss, style loss) are weighted as twice as the components of adversarial loss (e.g. global loss, local loss). Horizontal flipping is the only data augmentation method adopted during training. All experiments are conducted on a PC equipped with 2 parallel NVIDIA GTX 1080Ti GPUs.

Table 1. Experimental results of our model with different components. BN: Batch Normalization, IN: Instance Normalization, US: Bilinear Upsampling, TC: Transpose-convolution, CL: Content Loss, D: Descriptions, SA: Self-Attention

Models	ℓ_1 Loss %	ℓ_2 Loss %	SSIM	PSNR
BN + US	3.38	1.14	0.927	19.56
IN + US	3.35	1.11	0.932	19.61
BN + TC	3.31	1.08	0.942	19.78
IN + TC	**3.21**	**0.99**	**0.946**	**19.86**
IN + TC + CL	**3.20**	**0.97**	0.943	19.78
IN + TC - D	3.31	1.07	0.936	19.74
IN + TC - SA	3.37	1.12	0.931	19.59

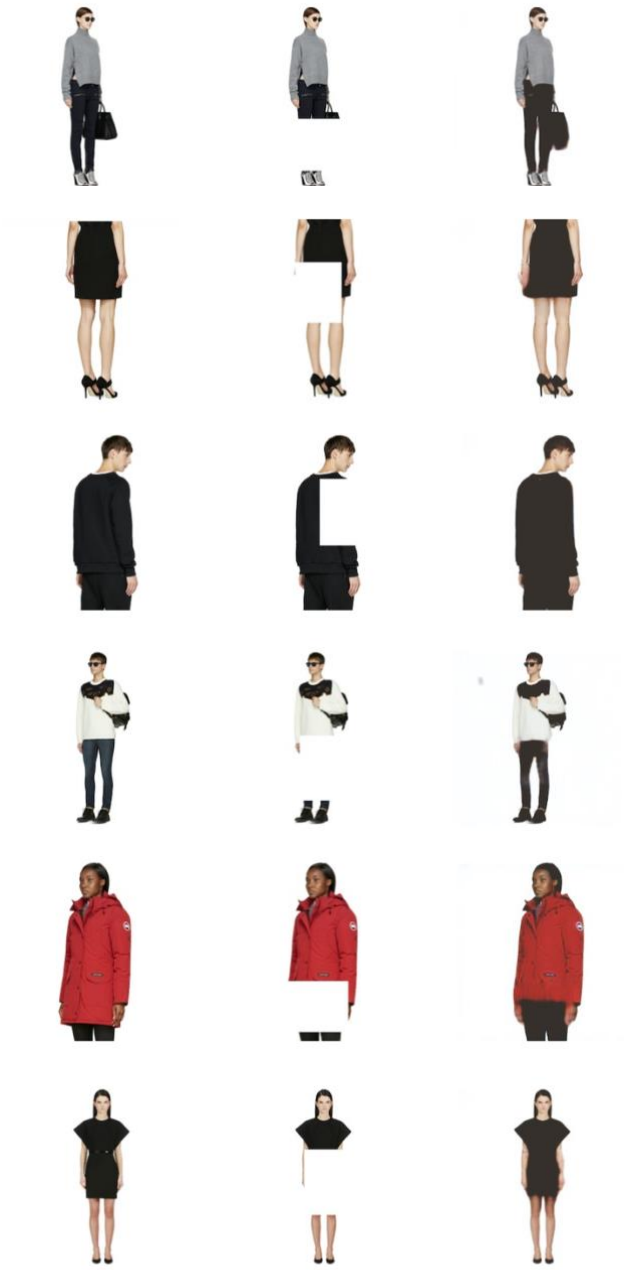

Figure 4. Qualitative results of our generative model.

5. RESULTS & DISCUSSION

5.1 Quantitative Results

Generative learning approaches have the deficiency of specialized quantitative evaluation metrics. To evaluate our results, we follow the studies in the literature, and employ four types of evaluation metrics to our experiments, which are ℓ_1 and ℓ_2 losses, structural similarity index (SSIM) and peak signal-to-noise ratio (PSNR). The first two metrics measure the pixel-wise difference between the actual image and the output. In addition, SSIM measures the distortion between two images, while PSNR examine the perceptual quality of generated image according to the actual image. Note that SSIM is a similarity metric and PSNR is a measurement of quality of reconstructed image, different than other metrics that we use, thus higher value of SSIM/PSNR represents the better result. The quantitative results of our experiments on fashion image inpainting by using multi-modal generative approach is introduced in Table 1. Although all experiments conducted on this task with different components have similar figures on all metrics, the model used transpose-convolutional layers in decoder blocks and instance normalization method in all blocks outperform the other models by a narrow margin. Moreover, using extra modalities such as textual features and self-attention mechanism improves overall performance of our model, while employing content loss does not give any contribution on inpainting performance, but increase the training time.

5.2 Qualitative Results

Next, we introduce the qualitative performances of our best model. During our experiments, the main motivation is to show that there are open research questions for image recovering or inpainting task in fashion domain, another motivation of this study is to observe the effect of the changes in the components of a generative architecture to the qualitative performance. Figure 4 introduces fashion image inpainting results of proposed description-aware coarse-to-fine (*i.e.* stacked) CNNs trained on fashion images ruined by rectangular masks. For all experiments, no post-processing step is applied to the results. Moreover, while Figure 5 demonstrates the performance of our model on images that contain different texture details, Figure 6 shows how our model can preserve the shape consistency on generated images.

5.3 Discussion

For generation part of our model, transpose-convolution has surprisingly better performance than Bilinear upsampling in our quantitative results, but no significant difference in qualitative representations. Although it is demonstrated that transpose-convolutional layers in decoder part causes the checkerboard artefact on the images generated from scratch [22], for filling a particular missing region in an image, they still produce semantically and visually more plausible qualitative results of our experiments, when compared to the results of bilinear upsampling layers followed by *1x1* convolutional layer. The possible reason for this issue is that a transpose-convolutional layer that has a number of trainable weights transforms the feature maps in such a way that it maintains the pattern in the input, on the other hand, using interpolation for upsampling without any trainable weights may not differentiate the patterns in the part of already available input and in the missing part. Therefore, this may lead to have better performance of transpose-convolutional layers for filling the missing regions when some pixels are available in the inputs.

The model with instance normalization (IN) produces more plausible texture details and color similarity than the one with batch normalization (BN). Batch normalization [23] computes one single mean and standard deviation, and thus, assumes that the whole layer is from a single Gaussian distribution, while instance-based version [24] does this job for each individual instance in the batch. In other words, BN normalizes all images across the batch and the spatial locations at the same time, IN does this job on only the spatial location. As parallel to the studies in the literature [25], [18], the strategy of using IN in encoder/decoder blocks performs better than using BN in our experiments on filling the missing parts in the images, and thus it is possible to generate better patches according to the whole image.

The benefits of content loss do not afford its computational cost. Content loss is introduced by [26], and measures the semantic differences between images using a pre-trained classification network like VGG [27] and ResNet [28]. This measurement is done by the Euclidean distance between high-level representations of the images in different layer levels of the network. Employing this kind of objective function to our model may increase the perceptual quality of our results a little, but it doubles the training time of our model, even this pre-trained classification network does not require the gradient computation during training, Therefore, it generates a bottleneck for our training in terms of both memory and speed, and we decide to not use this objective function for training of our model due to the limited computational resources.

Constraint by the other information sources like textual descriptions leads to faster convergence. Aforementioned before, we employ the descriptions of the images to our model by our textual feature extractor module, besides to the images. The main reason to do that is to investigate the performance variances on image inpainting task when utilizing different knowledge sources on different modalities. At this point, supporting the perceptual quality of generated images has not been improved when using the textual descriptions, there is a minor difference among the quantitative results. However, these weak features extracted by textual feature extractor module constrains the latent space for the input composed of higher-level feature maps, and this mechanism acts as a kind of regularization method. As a result, we can propound that using textual features besides images in vision tasks may lead to better convergence behavior of models during training.

Figure 5. Some results containing different texture details.

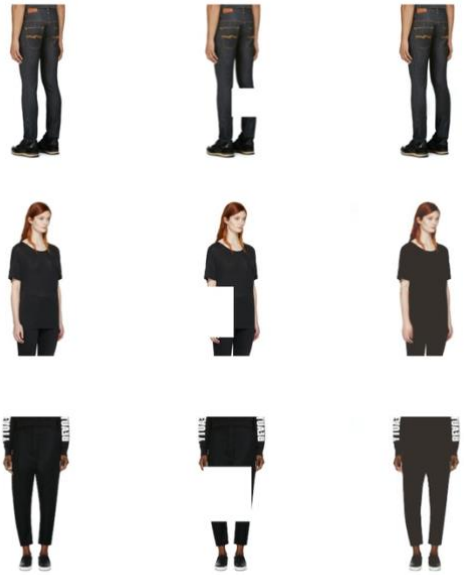

Figure 6. Some results preserving the shape consistency.

Qualitative results show that our approach can generate visually plausible contents for the missing region in the images. To look further details, the results can be fundamentally interpreted in 3 different aspects, shape, texture and color. As can be seen in detail in Figure 4, Figure 5 and Figure 6, our approach is able to generate the filled images while maintaining the shape information in the missing corner parts or near-to-center of the clothing items, and has the ability to distinguish different overlapping texture details more or less. By increasing the quality of predicted local missing parts, the texture details may be semantically more plausible for harder cases with more texture details. Lastly, with the help of instance normalization in decoder blocks, our model can fill the missing region in an image by an exact or close color code as appropriate to the semantic details.

6. CONCLUSION

In this study, we propose a multi-modal description-guided generative deep learning model for fashion image inpainting. While our approach is able to generate visually plausible patches with such appropriate texture details for the missing parts, it still needs to be investigated for improving inpainting quality. As a future work, we plan to improve our refinement modality for higher quality inpainting of the missing parts in the images. In addition, our experiments can be extended as employing larger and realistic datasets to our training to show the performance of proposed approach on different datasets.

7. ACKNOWLEDGMENTS

Our thanks to Prof. Dr. Ethem Alpaydın for his valuable comments on this study.

REFERENCES

[1] D. P. Kingma, M. Welling, "Auto-Encoding Variational Bayes". *CoRR*, 2013, abs/1312.6114.

[2] K. Sohn, H. Lee, X. Yan, "Learning Structured Output Representation using Deep Conditional Generative Models", *NIPS*, pages 3483-3491, 2015.

[3] L. M. Mescheder, S. Nowozin, A, Geiger, "Adversarial Variational Bayes: Unifying Variational Autoencoders and Generative Adversarial Networks", *ICML*, 2017.

[4] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, X. Bing, D. Warde-Farley, S. Ozair. A. C. Courville, Y. Bengio, "Generative Adversarial Networks", *arXiv preprint*, 2014.

[5] A. Radford, L. Metz, S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", *ICLR*, 2016.

[6] Z. Han, I. J. Goodfellow, D. Metaxas, A. Odena, "Self-Attention Generative Adversarial Networks", *ICML*, 2018.

[7] T. Karras, S. Laine, T. Aila, "A Style-Based Generator Architecture for Generative Adversarial Networks", *CVPR*, 2019.

[8] N. Rostamzadeh, S. Hosseini, T. Boquet, W. Stokowiec, Y. Zhang, C. Jauvin, C. Pal, "Fashion-Gen: The Generative Fashion Dataset and Challenge", arXiv preprint, 2018, jun, arXiv:1806.08317.

[9] A. A. Efros, W. T Freeman, "Image quilting for texture synthesis and transfer" In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346. ACM, 2001.

[10] A. A. Efros, T. K. Leung, "Texture synthesis by non-parametric sampling" In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, volume 2, pages 1033–1038. IEEE, 1999.

[11] D. Simakov, Y. Caspi, E. Shechtman, M. Irani, "Summarizing visual data using bidirectional similarity", *CVPR*, pages 1–8. IEEE, 2008.

[12] D. M. Mount, S. Arya, "Ann: library for approximate nearest neighbour searching" 1998.

[13] C. Barnes, E. Shechtman, A. Finkelstein, D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing" *ACM Transactions on Graphics (TOG) (Proceedings of SIGGRAPH 2009)*, 2009.

[14] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels", *IEEE transactions on image processing*, 2001.

[15] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, P. Sen, "Image melding: Combining in-consistent images using patch-based synthesis" *ACM Transactions on Graphics (TOG) (Proceedings of SIGGRAPH 2012)*, 2012.

[16] K. He, J. Sun, "Image completion approaches using the statistics of similar patches" *IEEE transactions on pattern analysis and machine intelligence*, 36(12):2423– 2435, 2014.

[17] S. Iizuka, E. Simo-Serra, H. Ishikawa, "Globally and locally consistent image completion" *ACM Transactions on Graphics (TOG)*, 36(4):107, 2017.

[18] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, T. S. Huang, "Generative image inpainting with con-textual attention", *CVPR*, pages 5505– 5514, 2018.

[19] G. Liu, F. A. Reda, K. J. Shih, T. Wang, A. Tao, B. Catanzaro, "Image inpainting for irregular holes using partial convolutions", *ECCV*, pages 85–100, 2018.

[20] O. Ronneberger, P. Fischer, T. Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation", *MICCAI*, 2015.

[21] D. Kingma, J.Ba, "Adam: A method for stochastic optimization", *ICLR*, 2015.

[22] A. Dosovitskiy, J. T. Springenberg, T. Brox, "Learning to generate chairs with convolutional neural networks", *CVPR*, 2015.

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

[24] D. Ulyanov, A. Vedaldi, V. S. Lempitsky, "Instance normalization: The missing ingredient for fast stylization", *CoRR*, 2016, abs/1607.08022.

[25] Y. Yang, X. Guo, J. Ma, L. Ma, H. Ling, "LaFIn: Generative Landmark Guided Face Inpainting", *arXiv preprint*, 2019.

[26] J. Johnson, A. Alahi, L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution", *ECCV*, pages 694-711, 2016.

[27] K. Simonyan, A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", 2014.

[28] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition", *CVPR*, pp. 770-778, 2016.