Semantic Image Inpainting with Deep Generative Models

Liangjie Cao

July 26, 2018

Abstract

This paper proposes Markovian Generative Adversarial Networks (MGANs), a method for training generative neural networks for efficient texture synthesis. While deep neural network approaches have recently demonstrated remarkable results in terms of synthesis quality, they still come at considerable computational costs (minutes of runtime for low-res images). This paper addresses this efficiency issue. Instead of a numerical deconvolution in previous work, they precompute a feed forward, strided convolutional network that captures the feature statistics of Markovian patches and is able to directly generate outputs of arbitrary dimensions. Such network can directly decode brown noise to realistic texture, or photos to artistic paintings. With adversarial training, they obtain quality comparable to recent neural texture synthesis methods. As no optimization is required any longer at generation time, their runtime performance (0.25M pixel images at 25Hz) surpasses previous neural texture synthesizers by a significant margin (at least 500 timesfaster). They apply this idea to texture synthesis, style transfer, and video stylization.

1. Introduction

Image synthesis is a classical problem in computer graphics and vision [1, 4]. The key challenges are to capture the structure of complex classes of images in a concise, learnable model, and to find efficient algorithms for learning such models and synthesizing new image data. Most traditional "texture synthesis" methods address the complexity constraints using Markov random field (MRF) models that characterize images by statistics of local patches of pixels.

Recently, generative models based on deep neural networks have shown exciting new perspectives for image synthesis [2]. Deep architectures capture appearance variations in object classes beyond the abilities of pixel-level approaches. However, there are still strong limitations of how much structure can be learned from limited training data. This currently leaves us with two main classes of "deep" generative models: 1) full-image models that generate whole images [2], and 2) Markovian models that also synthesize textures.

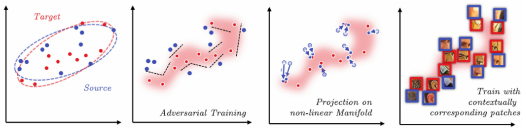

Figure 1. Motivation: real world data does not always comply with a Gaussian distribution (first), but a complex nonlinear manifold (second). They adversarially learn a mapping to project contextually related patches to that manifold.

2. Model

Let us first conceptually motive our method. Statistics based methods match the distributions of source (input photo or noise signal) and target (texture) with a Gaussian model (Figure 1, first). They do not further improve the result once two distributions match. However, real world data does not always comply with a Gaussian distribution. For example, it can follow a complicated non-linear manifold. Adversarial training [2] can recognize such manifold with its discriminative network (Figure 1, second), and strengthen its generative power with a projection on the manifold (Figure 1, third). We improve adversarial training with contextually corresponding Markovian patches (Figure 1, fourth). This allows the learning to focus on the mapping between different depictions of the same context, rather than the mixture of context and depictions.

Figure 2 visualizes our pipeline, which extends the patch-based synthesis algorithm of Li *et al.* [3].Firstly replacing the patch dictionary (including the iterative nearest-neighbor search) with a continuous discriminative network D (green blocks) that learns to distinguish actual feature patches (on VGG_19 layer Relu3_1, purple block) from inappropriately synthesized ones. A second comparison (pipeline below D) with a VGG_19 encoding of the same image on the higher, more abstract layer Relu5_1 can be optionally used for guidance. If they run deconvolution on the VGG networks (from the discriminator and optionally from the guidance content), they obtain deconvolutional

image synthesizer, which is called Markovian Deconvolutional Adversarial Networks (MDANs).

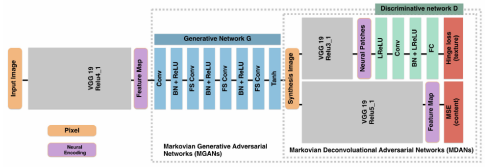

Figure 2. Our model contains a generative network (blue blocks) and a discriminative network (green blocks). We apply the discriminative training on Markovian neural patches (purple block as the input of the discriminative network.)

Formally, they denote the example texture image by $x_t \in \mathbb{R}^{w_t \times h_t}$, and the synthesized image by $x \in \mathbb{R}^{w \times h}$. They initialize x with random noise for un-guided synthesis, or an content image $x_c \in \mathbb{R}^{w \times h}$ for guided synthesis. The deconvolutioiteratively updates x so the following energy is minimized:

$$x = \arg\min_x E_t(\psi(x), \psi(x_t) + \alpha_1 E_c(\psi(x), \psi(x_c) + \alpha_2 \Upsilon(x) \quad (1)$$

MDANs require many iterations and a separate run for each output image. They now train a variational autoencoder (VAE) that decodes a feature map directly to pixels. The target examples (textured photos) are obtained from the MDANs. This generator G (blue blocks in Figure 2) takes the layer relu4_1 of VGG_19 as the input, and decodes a picture through a ordinary convolution followed by a cascade of fractional-strided convolutions (FS Conv). Although being trained with fixed size input, the generator naturally extends to arbitrary size images.

3. Experimental Analysis

Conducting empirical experiments with their model: they study parameter influence (layers for classification, patch size) and the complexity of the model (number of layers in the network, number of channels in each layer). While there may not be a universal optimal design for all textures, they study shed some light on how the model behaves for different cases. For fair comparison, they scale the example textures in this study to fixed size (128-by-128 pixels), and demand the synthesis output to be 256-by-256 pixels.

Testing theD with 4, 64, and 128 channels for the convolutional layer, they observe in general that decreasing the number of channels leads to worse results (fourth column, Fig. 3), but there is no significance difference between 64 channels and 128 channels (second column v.s. fifth column). The complexity requirements also depend on the actual texture. For example, the ivy texture is a rather simple

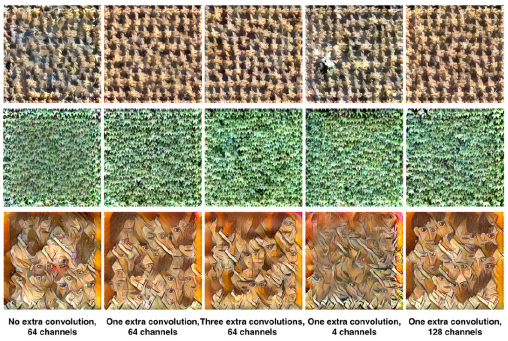

Figure 3. Different depths for training the discriminative netowrk. The input textures are “ropenet”, and Pablo Picassos “self portrait 1907”

MRF, so the difference between 4 channel and 64 channel are marginal, unlike in the other two cases.

References

[1] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In *SIGGRAPH*, 2001. 1

[2] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 1

[3] C. Li and M. Wand. Combining markov random fields and convolutional neural networks for image synthesis. In *CVPR*, 2016. 1

[4] L.-Y. Wei and M. Levoy. Fast texture synthesis using tree-structured vector quantization. In *SIGGRAPH*, 2000. 1