J. Vis. Commun. Image R. 40 (2016) 366–375

Contents lists available at ScienceDirect

J. Vis. Commun. Image R.

journal homepage: www.elsevier.com/locate/jvci

Combining inconsistent textures using convolutional neural networks ☆

Xiuxia Cai, Bin Song *

State Key Laboratory of Integrated Services Networks, Xidian University, Xian 710071, China

ARTICLE INFO

Article history:
Received 11 January 2016
Revised 13 May 2016
Accepted 16 July 2016
Available online 18 July 2016

Keywords:
Large-scale bound-constrained optimization
Image synthesis
Texture interpolation
Convolutional neural network

ABSTRACT

It is difficult for us to generate a gradually changed texture by combing two textures when they have very different textures and structures. This is a common problem in constructing panoramas and in pasting from one texture to another. The inconsistency between the sources is what we want to deal with in this paper. We present a new method for synthesizing a transition region between two source textures, such that inconsistent textures and structural properties all change gradually from one source to the other. We first extract the convolutional neural network (CNN) features of two source textures and one initial target texture at convolution and pooling layers. We set the distortion function to be the square of the difference between feature maps of source texture and target texture. And we based on feature maps compute the Gram matrix which is the inner product between feature maps in each layer. We set the second distortion function to be the square of the difference between Gram matrix of second source texture and target texture. Our target function is the weighted sum of two distortion functions we described before. We use Large-scale bound-constrained optimization method to optimize the target function and get the ultimate result. The model provides a new tool to generate a transition region between two source textures by CNN. Our method is robust to various types of images tested.

1. Introduction

The inconsistency is a common problem when we want to construct panoramas and in pasting from one texture to another. So we need to generate one area of texture gradually changed from one texture to another. Burt and Adelson [1] designed a multiresolution spline method to combine two or more images into a larger mosaic. This method can only generate a small gradually changed area. Rez et al. [2] introduced gradient domain methods to handle color inconstancy by smoothly interpolating the error inside a transition zone which is used to generate small area. These methods are ineffective when the regions have different textures. Darabi et al. [3] combine inconsistent images using patch-based synthesis which has been proven successful. But this method would produce obviously artificial texture which caused by patch based synthesis method. Beside this method requires a preprocessed input image. The input images are processed into the kind of image as shown in Fig. 1. The textures on two sides of the figure are exemplars. The magenta area is the transition region to be generated. In this paper, we present a new method for synthesizing a transition region between two source textures, such that inconsistent textures and structural properties all change gradually from one source to the other. And we don't need a preprocessed image. Since we take feature maps as one whole unit to optimize, the local optimal problem is avoided. Experimental results show that our method has better biological visual effect.

Ever since Russakovsky et al. [6], Krizhevsky et al. [4], and Taigman et al. [7] have made a breakthrough on the ImageNet dataset for convolutional neural network (ConvNet or CNN [5]), there are a lot of interesting research results based on the convolutional neural network [8]. One of interesting research is Gatys et al. [9] generating texture using the CNN features at different layers, which by matching the CNN features of an input texture. Another way of generating images is Simonyan and Zisserman [10] and Simonyan et al. [11] maximizing the responses of CNN units.

Inspired by the prosperity of texture synthesis [12–16] based on CNN features we consider combining two inconsistent textures based on convolutional neural network. Moreover, in light of the similarities between performance of convolutional neural networks and biological vision [17–20], we consider the feature maps in forward processes of convolutional neural network as the important information that how humans understand and perceive image content.

In our method, we first extract the CNN features at convolution and pooling layers for two source textures and the initial target texture. Then we set the first loss energy between first source

☆ This paper has been recommended for acceptance by Dr. Zicheng Liu.

* Corresponding author.
E-mail address: bsong@mail.xidian.edu.cn (B. Song).

http://dx.doi.org/10.1016/j.jvcir.2016.07.009
1047-3203/

Fig. 1. One example of preprocessed image in the method of Darabi et al. [3].

texture and target transition region texture to be the square of the difference between their feature maps, and the second loss energy between the second source texture and target transition region texture is the square of the difference between their Gram matrix [21] which is the inner product between feature maps in each layer. The object function is weighted sum of two finite distortion function. The object function is a convex optimization problem, so we use Large-scale bound-constrained optimization (L-BFGS, [22]) method to optimize the target function and get the ultimate result. It is probably safe to say that no technique based on CNN has yet been found to combine two inconsistent textures. To demonstrate the robust of our method we test it on textures of regular, near regular, irregular, and containing small or large scale structures. The experimental results show that our algorithm can generate comfortable visual texture and is robust on the combination of two inconsistent textures.

The remainder of this paper is organized as follows. In Section 2 we review related work and symmetry theory of L-BFGS, which introduced in two subsections separately. In Section 3, we first present the framework of our method of combining two inconsistent textures. In Section 3.1 we present the texture model we used. Section 3.2 explains how we can generate the gradually changed region. Section 4 gives several experiment results in our method. We conclude this paper in Section 5. According to the limitations of our algorithm we give the plan of the future work.

2. Previous work

2.1. Convolutional neural network

Cimpoi et al. [23,24] have provided a fruitful new analysis tool for studying visual perception with CNN. Based on their work the texture in this paper is represented with CNN. The VGG network was extensively used and introduced in recent object recognition research works which are based on convolutional neural network. Considering the well design of VGG we also use VGG-19 (VGG-16 is suitable in this paper, we're just randomly choosing one of them) network in this paper.

Inspired by VGG-19 network's architecture, our convolutional neural network computation is mainly based on linearly rectified convolution and average pooling. The convolution filters are size of $3 \times 3 \times k$ where k is the number of input feature maps. The size of pooling windows is 2×2 in non-overlapping regions. The convolutional layer is followed by an average pooling layer (Fig. 2).

2.2. Large-scale bound-constrained optimization

Large-scale bound-constrained optimization which is a limited-memory algorithm for solving large nonlinear optimization problems subject to simple bounds on the variables developed from Quasi-Newton method [25]. Before Quasi-Newton method there are steepest descent method, Newton method, conjugate gradient method and so on [26]. All the methods we mentioned before are designed to solve the optimization problem. And these methods are through calculating Hessian matrix or gradient descent to find the optimize result. In Quasi-Newton method the $f(x)$ to be optimized is

$$f(x) = f(x_{i+1}) + (x - x_{i+1})^T \nabla f(x_{i+1}) + \frac{1}{2}(x - x_{i+1})^T H_{i+1}(x - x_{i+1}) + o(x - x_{i+1}), \tag{1}$$

where x_{i+1} is one value in domain, and H_{i+1} is the Hessian matrix in x_{i+1}. Eq. (1) derivation and ignoring high small order item we can get

$$\nabla f(x) \approx \nabla f(x_{i+1}) + H_{i+1}(x - x_{i+1}). \tag{2}$$

Let $x = x_i$ then

$$H^{-1}_{i+1}(\nabla f(x_{i+1}) - \nabla f(x_i)) \approx (x_{i+1} - x_i). \tag{3}$$

If we let $B_{i+1} = H^{-1}_{i+1}$ then we need to calculate the B_i for the optimize result. Let $t_i = \nabla f(x_{i+1}) - \nabla f(x_i)$ and $s_i = x_{i+1} - x_i$, and we save the m nearest t_i and s_i. So in L-BFGS has

$$\begin{aligned} B_i &= V^T_{i-1}B_{i-1}V_{i-1} + r_{i-1}s_{i-1}s^T_{i-1} \\ &\vdots \\ &= \left(V^T_{i-1}\cdots V^T_{i-m}\right)B^0_i(V_{i-m}\cdots V_{i-1}) \\ &\quad + r_{i-m}\left(V^T_{i-1}\cdots V^T_{i-m+1}\right)s_{i-m}s^T_{i-m}(V_{i-m+1}\cdots V_{i-1}) \\ &\quad + \cdots \\ &\quad + r_{i-1}s_{i-1}s^T_{i-}, \end{aligned} \tag{4}$$

where $r_i = \frac{s^T_{i-1}t_{i-1}}{t^T_{i-1}t_{i-1}}$, $B^0_i = r_iI$. In our method of combining two inconsistent textures, we use Eq. (4) to calculate the gradually changed subregions.

3. Algorithm

The proposed architecture of our method is shown in Fig. 2. We can directly visualize the information of each layer which contains about all the information of input textures. By weighted sum the information of feature maps of two textures we can reconstruct one new texture [27,28]. The greater the weight value of the input texture information, the more similar is the output texture to the input texture.

We hope the content of texture in transition region is very similar to one input texture at the beginning. And the style of texture in transition region is more and more similar to another input texture. We take the first source texture as the content control, which means the feature maps between the first source and the target transition region texture are similar. So we set the first loss energy