Around Deep Image Prior

Alexander Borzilov
Optimization Class Project. MIPT

Introduction

Deep convolutional networks have become a popular tool for image generation and restoration. Generally, their excellent performance is imputed to their ability to learn re- alistic image priors from a large number of example images. But recent work [1] has demonstrated that, on the contrary, the structure of a generator network is sufficient to capture a great deal of low-level image statistics prior to any learning.

Main Idea

In image restoration problems the goal is to recover original image x having a corrupted image x_0. Such problems are often formulated as an optimization task:

$$\min_x E(x; x_0) + R(x)\,, \tag{1}$$

where $E(x; x0)$ is a data term and $R(x)$ is an image prior. The data term $E(x; x0)$ is usually easy to design for a wide range of problems, such as super-resolution, denoising, inpainting, while image prior $R(x)$ is a challenging one. Today's trend is to capture the prior $R(x)$

with a ConvNet by training it using large number of examples.

We first notice, that for a surjective $g : \theta \mapsto x$ the following procedure in theory is equivalent to (1):

$$\min_\theta E(g(\theta); x_0) + R(g(\theta))\,.$$

In practice g dramatically changes how the image space is searched by an optimization method. Furthermore, by selecting a "good" (possibly injective) mapping g, we could get rid of the prior term. We define $g(\theta)$ as $f_\theta(z)$, where f is a deep ConvNet with parameters θ and z is a fixed input, leading to the formulation

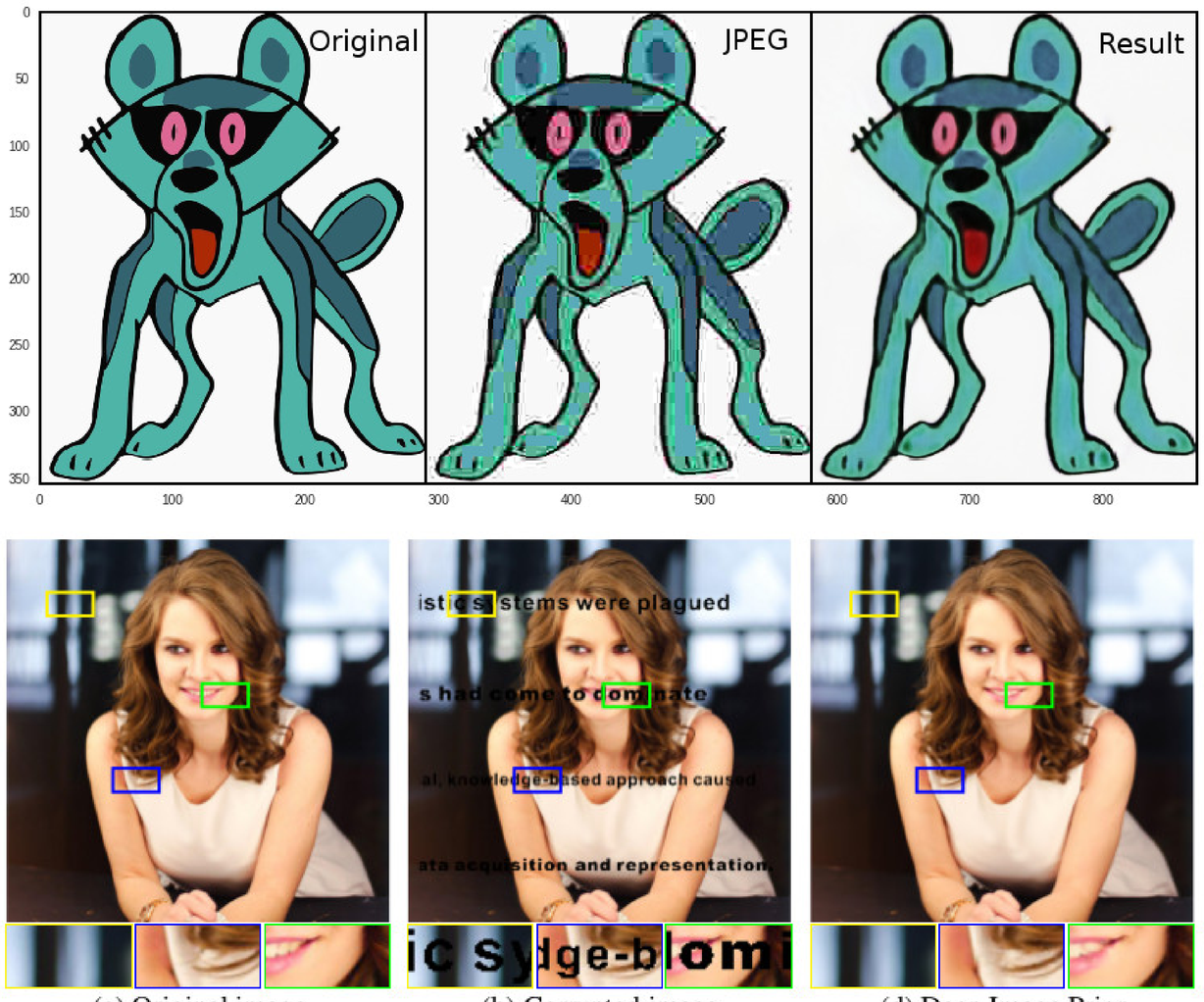
$$\min_\theta E(f_\theta(z); x_0)\,.$$

is filled with noise and fixed.

In other words, instead of searching for the answer in the image space we now search for it in the space of neural network's parameters. We emphasize that we never use a pretrained network or an image database. Only corrupted image $x0$ is used in the restoration process.

Some examples

You can run python notebooks with different examples using google collaboratory [2]. Here you can see results of aplying Deep Image Prior for image denoising and inpainting:



(a) Original image (b) Corrupted image (d) Deep Image Prior

References

[1] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep Image Prior," *ArXiv e-prints*, Nov. 2017.

[2] A. Borzilov, "Run examples." `https://drive.google.com/drive/folders/1sk6qVo1UPAKkF3Lc1t4TcSRG-SYLrzeA?usp=sharing`, 2017. [Online; accessed 19-July-2018].

[3] D. Ulyanov, "Deep Image Prior." `https://dmitryulyanov.github.io/deep_image_prior#mjx-eqn-eq1`, 2017. [Online; accessed 19-July-2018].