Computational Photography Research Midterm Report

Chetankumar Mistry

October 2018

1 Introduction

In 2015, Google engineer Alexander Mordvintsev released DeepDream, which combines images in particular ways, creating an output which has a dream-like/hallucinogenic feel to it. From hereon it will be referred to as an Inceptionism style.

DeepDream uses a convolutional neural network (CNN) to produce these results. This project will try to replicate similar results by using alternative methods and avoiding the use of a neural network entirely; primarily by using pure image processing techniques.

By not using a CNN, the hope is that the quality of results (output images) will be similar but the time it takes to process an image will be faster and overall, the final system will be easier to understand and require less dependencies than DeepDream.

2 Justification

There are 30+ different layers of neurons which each determine how the final output will appear. By tweaking each layer, it is possible to emphasise certain features. Using the dream.ipynb which can be downloaded here:

https://github.com/google/deepdream

We can see that the way the images are produced are via a gradient ascent by maximising the L2 (Euclidean) Norm at each layer. If low-level layers are weighted higher, then edges and corners are brought out more, whereas high-level layers produce more complex results. This project will seek to find out the best combination of features to enhance to produce similar results.

Figure 1: An example image which can be created by using DeepDream

One reason for this is computational speed; having used the DeepDream program to generate some images, one thing that became apparent is that it is very taxing on resources due to the innate complexity involved in image processing but also because each layer of the CNN has its own processing stage, it can take a lot of time just to produce a single image. By trying to create a system which avoids the use of a CNN, processing speeds can be increased by only processing wanted features.

Another reason is that, due to the complexity of using a CNN, and also the fact that it is written using Python2, it is not the easiest of systems to initially set up and start using so that the system can start to be learnt and understood.

3 Current System

At the moment, the current system available is DeepDream itself, which works very well in creating images in an Inceptionism style. Whilst DeepDream can be used directly from the .ipynb file found on the linked GitHub repository, there are also online tools available to produce the same kind of results. For example, at:

https://deepdreamgenerator.com/

it is possible to create images as shown in Figures 2-4.

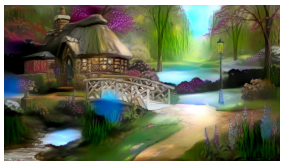

Figure 2: Style image used for Deep Dream

Figure 3: Content image used for Deep Dream

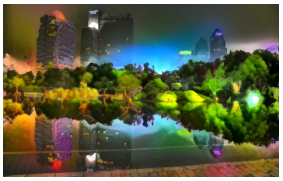

Figure 4: Output image from Deep Dream using the previous 2 images

DeepDream is currently available at https://github.com/google/deepdream as the "dream.ipynb". This is written in Python2 and so some conversions are needed for it to work on Python3 (NB. this project will be written in Python3), not only this, but any potential user who wishes to use "dream.ipynb" file will also have to set up "Caffe" on their system so that all the dependencies are met.

Finally there are some additional files which aren't immediately provided in the repository and so, if the user does not already have them, will have to download/create these files themselves. For example: "deploy.prototxt" and "bvlc_googlenet.caffemodel".

Also currently available at https://github.com/kesara/deepdreamer is an alternative program written in Python3 with clear instructions on how to set up and use DeepDream. As such, all of the aforementioned changes needed between Python2 and Python3 have already been made, and, whilst the two files mentioned above also aren't provided; there are clear instructions on where to get them and how to use them.

4 Solution

The plan for this project is to build another system which will produce images similar to that which can be done by DeepDream, but without the use of a convolutional neural network. As such, this will be an entirely separate project and will not build on top of the pre-existing system, but will be heavily inspired and influenced by what is already available.

The system will work by manually simulating what is done at particular layers of the neural network which DeepDream uses. For example, we know that the lower levels enhance more physical elements of an image such as edges and corners; from this the new system will use image processing techniques to find the edges and corners in an image, modify them in some way and then superimpose the change onto the original image, effectively applying an Inceptionism style onto the image.

Whilst this project will try to create recreate what DeepDream does, it is doubtful that the new system will actually surpass the performance of it. There is only one potential area where it can perform better than DeepDream and that is with regards to its Time Performance, in that, the new system will not be as computationally complex and challenging, and so would output images faster than the original.

From this, the way that performance of the new system could be tested will be through time benchmark tests, and through visually comparing the output images and seeing whether or not they exhibit the Inceptionism style which it should do and whether or not it is comparable to DeepDream.

However, time benchmark tests will be difficult to really measure, since, as previously stated, the new system will not be as complex or do as much as DeepDream, and so, the output between the 2 systems will be vastly different. As such, using time may not be a good comparitor for the 2 systems.