Applying deep learning

Style Transfer

As an example of the kind of things you'll be building with deep learning models, here is a really fun project, fast style transfer. Style transfer allows you to take famous paintings, and recreate your own images in their styles! The network learns the underlying techniques of those paintings and figures out how to apply them on its own. This model was trained on the styles of famous paintings and is able to transfer those styles to other images and even videos!



Dependencies

Install Miniconda

**Windows**

For Windows, you'll need to install TensorFlow 0.12.1, Python 3.5, Pillow 3.4.2, scipy 0.18.1, and numpy 1.11.2.

After installing Miniconda, open your command prompt. In there, enter these commands line by line:

conda create -n style-transfer python=3

activate style-transfer

conda install tensorflow scipy pillow

pip install moviepy

python -c "import imageio; imageio.plugins.ffmpeg.download()"

**Download the model for Transferring styles**

Download the fast-style-transfer-master Zip archive from my DL workshop repository and extract it.

Download the Rain Princess checkpoint from https://d17h27t6h515a5.cloudfront.net/topher/2017/January/587d1865\_rain-princess/rain-princess.ckpt. Put it in the fast-style-transfer folder. A checkpoint file is a model that already has tuned parameters. By using this checkpoint file, we won't need to train the model and can get straight to applying it.

Copy the image you want to style into the fast-style-transfer folder.

Enter the Conda environment you created above, if you aren't still in it.

Finally, in your terminal, navigate to the fast-style-transfer folder and enter

python evaluate.py --checkpoint ./rain-princess.ckpt --in-path <path\_to\_input\_file> --out-path ./output\_image.jpg

**Note**:

1. Your checkpoint file might be named rain\_princess.ckpt, notice the underscore, it's not the dash from above.
2. Be careful with the size of the input image. The style transfer can take quite a while to run on larger images.