Digital Image Processing Laboratory: Deep Learning Lab

April 9, 2019

1 Introduction

The goal of this laboratory is to introduce the methods of neural networks for image processing. In particular, we will focus on the methods of convolutional neural networks (CNN), and deep learning (DL), which are widely used in image processing applications.

In order to simplify developing, debugging, and running the code, we will use the standard tools of the Python programming language together with the Keras software library. Keras provides an easy-to-use interface to Google's TensorFlow open source machine learning framework. We will implement our code using the jupyter notebook environment in Google's Colaboratory or Colab. The Colab is a cloud based system with an interactive coding environment that is easy to use and widely available.

The laboratory will focus on three important techniques that are representative of how CNNs are used in image processing.

- Image classification — Here, you will implement a deep CNN that will classify hand written digits from the MINIST dataset.
- Image autoencoding — An autoencoder first compresses an image into a small number of parameters, and then reconstructs the original image from that compressed representation. The autoencoder uses unsupervised training, so it does not require labeled training data. In addition, it is a key component in many important DL algorithms.
- Image denoising — Here, you will design an image denoiser based on CNNs. Both the input and output of this CNN are images, and the training data can be generated by adding synthetic noise to natural images.

As in the lecture material, we learned the basic concept behind convolutional neural networks and autoencoders. In this lab, we will learn how to use these network structures to process real data sets. **All solutions to this laboratory should be implemented in Python/Keras.** In addition, most lab instructions are specified in the jupyter notebook files. So you will need to access to these notebooks in order to do your labs.

Questions or comments concerning this laboratory should be directed to Prof. Charles A. Bouman, School of Electrical and Computer Engineering, Purdue University, West Lafayette IN 47907; (765) 494-0340; bouman@ecn.purdue.edu

2 Environment setup

While any browser should work, we recommend you use Google's Chrome browser for this labs since it tends to work more seamlessly with the Google Colab tools. In order to set the environment up:

1. Download the zip file named **DL-LAB.zip** from the main web page of this lab and save it to your computer's desktop.

2. Set up a Google account if you don't already have one, login to your Google account, and go to your Google Drive at `https://www.google.com/drive`.

3. Upload both Jupyter notebook files in **DL-LAB.zip** to your Google Drive. You should be able to do this by selecting "File upload" using either a mouse-right-click or by selecting the symbol "+ new" in the upper left hand corner.

4. Double click on the Jupyter notebook file, and then select **Open with Colaboratory** on top of your page. This should put you into the Colab working space.

5. On Menu, select **Edit** → **Notebook Settings** → **Hardware Accelerator** → **GPU**. This step will allow your code to run on GPU accelerators which are optimized for DL operations.

Colab runs on cloud based servers, so it is crucial to run this lab on computers that have a stable internet connection. If you prefer to use your own computers to host the Python environment, you are welcome to do so, but you will need to install the following packages into your environment:

- Python (v3.6.7)
- Numpy (v1.14.6)
- Matplotlib (v3.0.2)
- Scipy (v1.0.0)
- Keras (v2.2.4) using TensorFlow backend

3 Jupyter Notebook

A jupyter notebook is an interactive text file. It will allow you to run codes in the browser while you are reading it. The executable code blocks are dark shaded texts. You can click these blocks and execute them in the following ways:

- Click the **Play icon** in the left gutter of the block
- Type **Cmd/Ctrl+Enter** to run the cell in place;

In your lab report, **only submit the PDF version of your 'DL_Lab_Exercise.ipynb' notebook**. Here is how you create the PDF:

1. For each exercise, write your code under one block as provided in the skeleton.
2. Run all the blocks on your **'DL_Lab_Exercise.ipynb'** notebook file so that all outputs are displayed immediately after each block.
3. Close the content list on the left side, and on **Menu**, select **File** → **Print**
4. Change the print destination to **Save as pdf** to download the pdf version of your jupyter notebook.

4 Useful Colab Tips

- **Q:** How to run all the cells before the current cell?
- **A:** On **Menu**, select **Runtime** → **Run before**. Note: This is particularly useful if you are opening a Jupyter notebook and immediately want to run blocks that are not the first block. Sometimes, previous blocks provide variables or package support for the current block, it is always safe to click **Run before** if you are running your blocks of code not sequentially from the start.
- **Q:** How to save your current work in Colab?
- **A:** If you are connected to the Internet, all work is automatically saved on your Google Drive.