Neural Networks Project _

The Real Problem

Recognizing multi-digit numbers in photographs captured at street level is an important component of modern-day map making. A classic example of a corpus of such street level photographs is Google's Street View imagery comprised of hundreds of millions of geo-located 360 degree panoramic images. The ability to automatically transcribe an address number from a geo-located patch of pixels and associate the transcribed number with a known street address helps pinpoint, with a high degree of accuracy, the location of the building it represents.

More broadly, recognizing numbers in photographs is a problem of interest to the optical character recognition community. While OCR on constrained domains like document processing is well studied, arbitrary multi-character text recognition in photographs is still highly challenging. This difficulty arises due to the wide variability in the visual appearance of text in the wild on account of a large range of fonts, colors, styles, orientations, and character arrangements. The recognition problem is further complicated by environmental factors such as lighting, shadows, specularities, and occlusions as well as by image acquisition factors such as resolution, motion, and focus blurs.

In this project we will use dataset with images centred around a single digit (many of the images do contain some distractors at the sides). Although we are taking a sample of the data which is simpler, it is more complex than MNIST because of the distractors.

Project Description

In this hands-on project the goal is to build a python code for image classification from scratch to understand the nitty gritties of building and training a model and further to understand the advantages of neural networks. First we will implement a simple KNN classifier and later implement a Neural Network to classify the images in the SVHN dataset. We will compare the computational efficiency and accuracy between the traditional methods and neural networks.

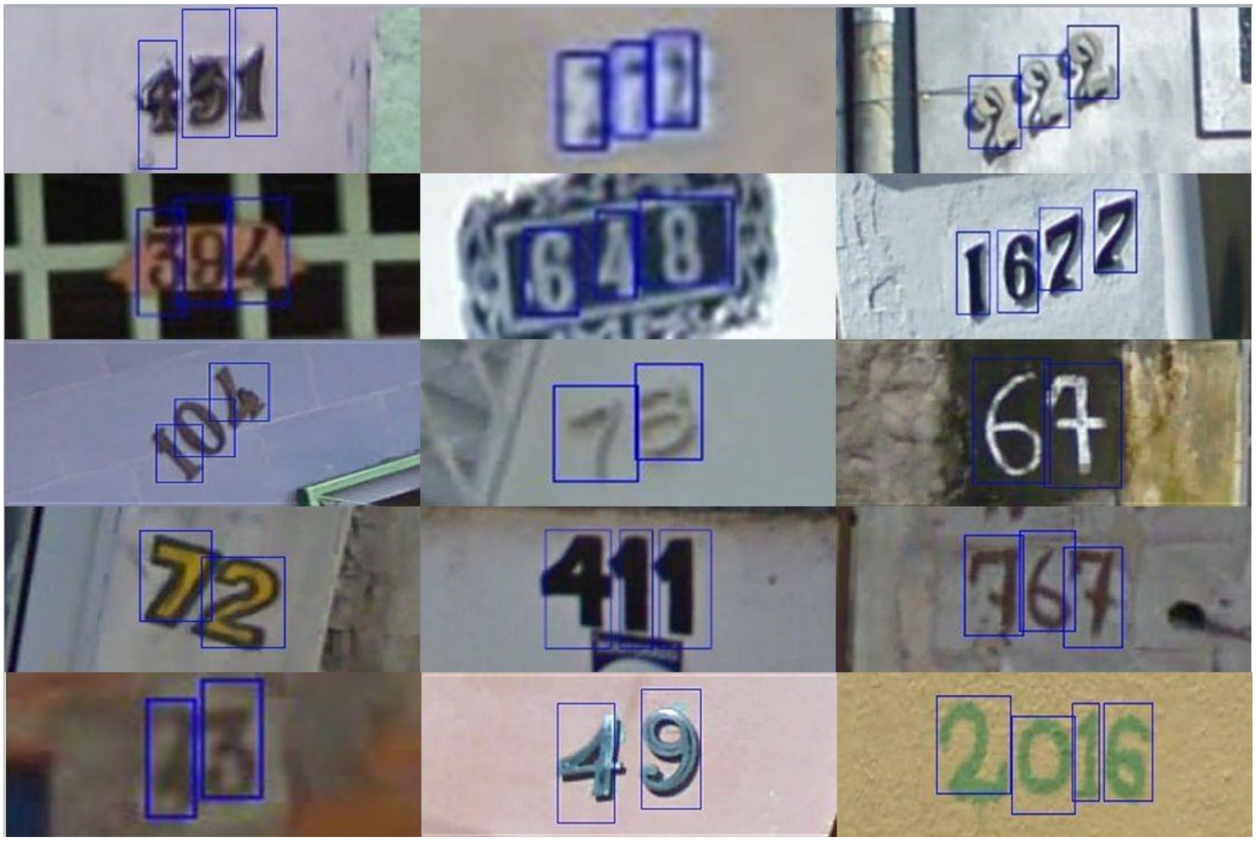
The Street View House Numbers (SVHN) Dataset

SVHN is a real-world image dataset for developing machine learning and object recognition algorithms with minimal requirement on data formatting but comes from a significantly harder, unsolved, real world problem (recognizing digits and numbers in natural scene images). SVHN is obtained from house numbers in Google Street View images.

Overview

The images come in two formats as shown below.

Format 1 : Original images with character level bounding boxes.



Format 2 : MNIST-like 32-by-32 images centered around a single character (many of the images do contain some distractors at the sides).

The goal of this project is to take an image from the SVHN dataset and determine what that digit is. This is a multi-class classification problem with 10 classes, one for each digit 0-9. Digit '1' has label 1, '9' has label 9 and '0' has label 10.

Although, there are close to 6,00,000 images in this dataset, we have extracted 60,000 images (42000 training and 18000 test images) to do this project. The data comes in a MNIST-like format of 32-by-32 RGB images centred around a single digit (many of the images do contain some distractors at the sides).

Reference

Acknowledgement for the datasets.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, Andrew Y. Ng Reading Digits in Natural Images with Unsupervised Feature Learning *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*. PDF

http://ufldl.stanford.edu/housenumbers as the URL for this site when necessary

Downloads

Refer to Olympus for project related files and instructions.

Data Set:

- The name of the dataset is **SVHN_single_grey1.h5**
- The data is a subset of the original dataset. Use this subset only for the project.
- Keep a copy of your dataset in your own google drive.

Project Objectives

The objective of the project is to learn how to implement a simple image classification pipeline based on the k-Nearest Neighbour and a deep neural network. The goals of this assignment are as follows:

- Understand the basic Image Classification pipeline and the data-driven approach (train/predict stages)
- Data fetching and understand the train/val/test splits.
- Implement and apply an optimal k-Nearest Neighbor (kNN) classifier (15 Points)
- Print the classification metric report (5 Points)
- Implement and apply a deep neural network classifier including (feedforward neural network, RELU activations) (10 Points)
- Understand and be able to implement (vectorized) backpropagation (cost stochastic gradient descent, cross entropy loss, cost functions) (5 Points)
- Implement batch normalization for training the neural network (5 Points)

- Understand the differences and trade-offs between traditional and NN classifiers with the help of classification metrics (10 points)