Street View House Number Recognition using Deep Convolutional Neural Networks

JYOTI NARWARIYA & NIKHIL JAISWAL

15 April,2017

1 Abstract of the project

Recognizing arbitrary multi-digit numbers from Street View imagery is one of the challenging task which can be tackled with the help of deep learning. To achieve this proposed target, we have used deep convolution neural network with multiple hidden layers that operates directly on image pixels. Having gone through some of the literature, we inferred the knowledge that to achieve such aim, we had to perform per digit recognition task. If we wanted to train on the digit-level, we had to follow basically three steps: localization, segmentation & recognition.

2 Introduction

In this project, We have used Convolutional Neural Networks(CNN) to detect house number with correct sequence. Here, we have taken only those images that contains only 4 digit. CNN classify the digits among 11 different classes (0-class for 0, 1- class for 1,....., 9-class for 9 and 10-class for if there is no digit). We have used Softmax that generated predicted logits for each digit. At last, we have calculated the accuracy by matching predicted logits with actual labels of image digits.

2.1 Literature survey

- Paper 1: Reading Digits in Natural Images with Unsupervised Feature Learning, NIPS Workshop 2011
  Summary: In this paper the researchers have presented an approach to tackle digit recognition problem in real world image using unsupervised feature learning methods. They have used several existing unsupervised feature learning algorithms that learn these features from the data itself. Each of these methods involves an unsupervised training stage where the algorithm learns a parametrized feature representation from the training data. The result of this training stage is a learned feature mapping that takes in an input image and outputs a fixed-length feature vector to be used for supervised training. Given an input image (32-by-32 pixels, grayscale) they have experimented with two different feature learning algorithms (i) stacked sparse auto-encoders and (ii) the K-means-based system to yield a fixed-length feature vector. Then they trained a linear SVM classifier from the labeled training data using these features as input, and test the classifier on the test set.
  Their solution combined three stages: detection, segmentation and classification. They have used a sliding window classifier based on a set of elementary features computed from image intensities and gradients . The building number detections are then sent to the recognition module. The recognition module consists of two steps: character segmentation, in which they found a set of candidate vertical boundaries between characters (breakpoints), and character hypothesis search, where they incrementally searched left-to-right the breakpoints pairs, and for each pair they evaluated using the character classifier.

  In this paper they have shown the major advantages of learned representations features over hand crafted features and had finally concluded with the expectation of the development of more sophisticated methods to reach human level performance.

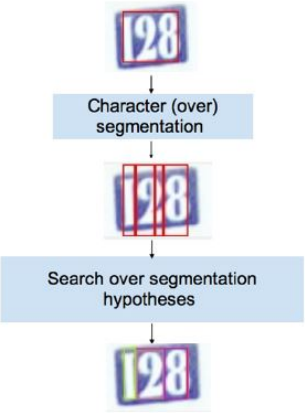
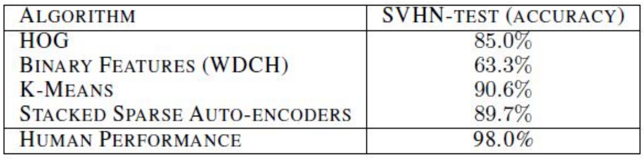

Character (over) segmentation
Search over segmentation hypotheses

Figure 1: Approach used in this paper



Algorithm	SVHN-test (accuracy)
HOG	85.0%
Binary Features (WDCH)	63.3%
K-Means	90.6%
Stacked Sparse Auto-encoders	89.7%
Human Performance	98.0%

Figure 2: Accuracies on SVHN-test

- Paper 2: Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks, Google Research 2013.
  Summary: In this paper the researchers have proposed an unified approach that integrated the detection, segmentation and classification steps via the use of a deep convolutional neural network. They have also proposed a new kind of output layer which provides a conditional probabilistic model of sequences.
  They trained a probabilistic model of sequences given input images. Suppose S represent the output sequence and X represent the input image. The goal was to learn a model of P(S | X) by maximizing log P(S | X) on the training set. Let S be a collection of N random variables $S_1, ..., S_N$ representing the elements of the sequence and an additional random variable L representing the length of the sequence. The probability of a specific sequence $s = s_1, ..., s_N$ is given by
  $P(S = s|X) = P(L = n|X)\Pi_{i=1}^{n} P(S = s_i|X)$

  They have used a softmax classifier that receives as input features extracted from X by a convolutional neural network. To train the model, they have maximize log P(S | X) on the training set using a generic method like stochastic gradient descent.

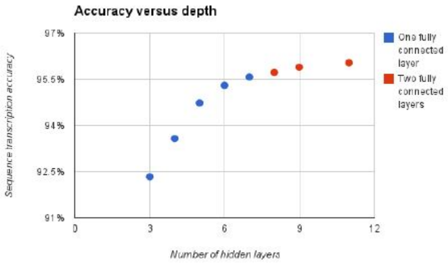


Figure 3: Performance analysis experiments on the public SVHN dataset

Their best architecture consisted of eight convolutional hidden layers, one locally connected hidden layer, and two densely connected hidden layers. All connections were feedforward and go from one layer to the next. Their best model obtained a sequence transcription accuracy of 96.03

3 Resources

We have taken the help of publicly available Street View House Number (SVHN) dataset. We have downloaded the dataset from the following url: http://ufldl.stanford.edu/housenumbers/
We have used Python to implement our code. We have used TensorFlow, numpy and any other libraries which were required during the implementation time.

3.1 Work done

- Description of the data: SVHN is obtained from house numbers in Google Street View images. It can be seen as similar in flavor to MNIST (e.g., the images are of small cropped digits), but incorporates an order of magnitude more labeled data (over 600,000 digit images) and comes from a significantly harder, unsolved, real world problem (recognizing digits and numbers in natural scene images).

  Features:
  1. 10 classes, 1 for each digit. Digit '1' has label 1, '9' has label 9 and '0' has label 10.
  2. 73257 digits for training, 26032 digits for testing, and 531131 additional, somewhat less difficult samples, to use as extra training data

  Comes in two formats:
  1. Original images with character level bounding boxes.
  2. MNIST-like 32-by-32 images centered around a single character (many of the images do contain some distractors at the sides).

  We have used the full number format dataset.
  Full Numbers: train.tar.gz, test.tar.gz , extra.tar.gz

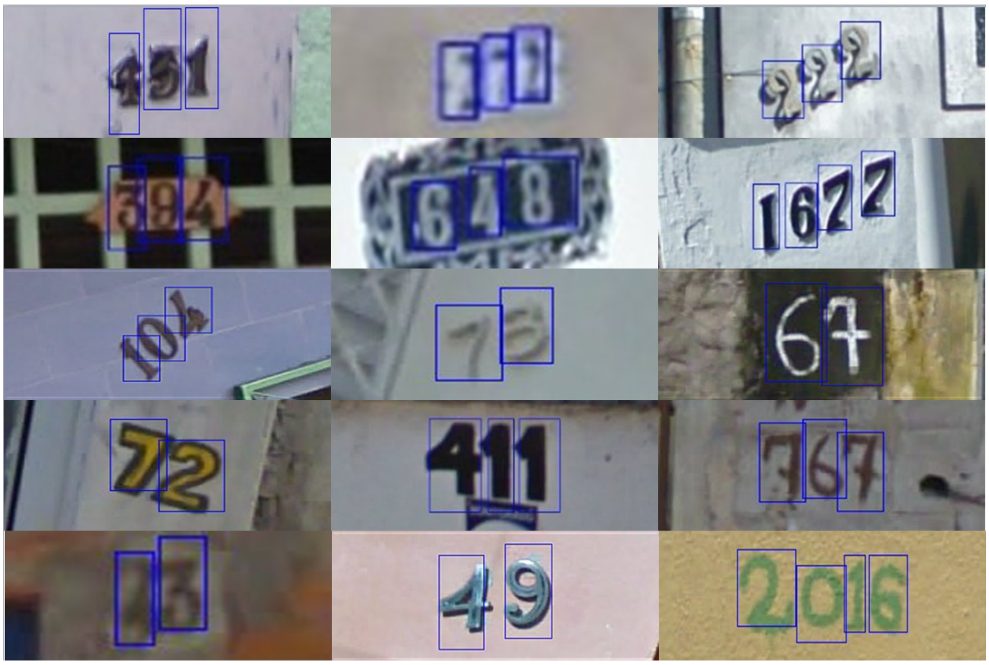


Figure 4: Sample from Full number format of dataset with bounded boxes for digits

- Exploration of different neural networks and observation: We have used Convolutional Neural Network due to their ability to work on the raw pixels of the image.

- Error plot for validation set: The following plots were obtained during training & testing on different conditions.

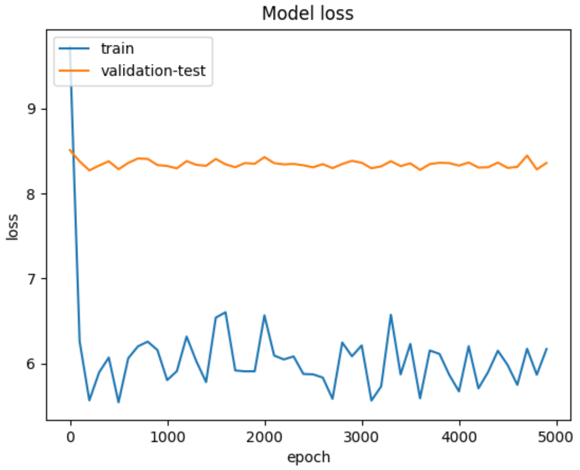


Figure 5: 4layers CONV POOL CONV Dropout FC

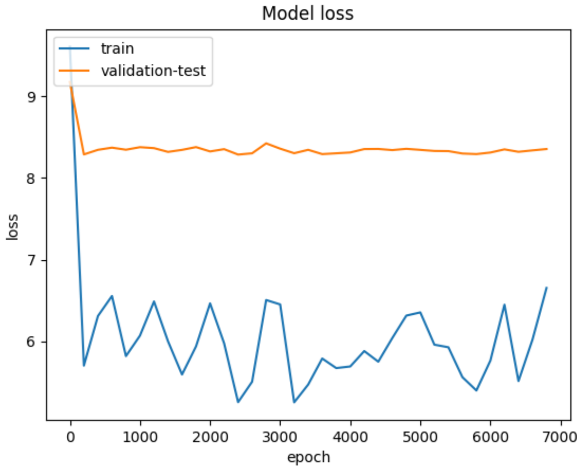


Figure 6: 6layers CONV POOL CONV POOL CONV FC NoDropout

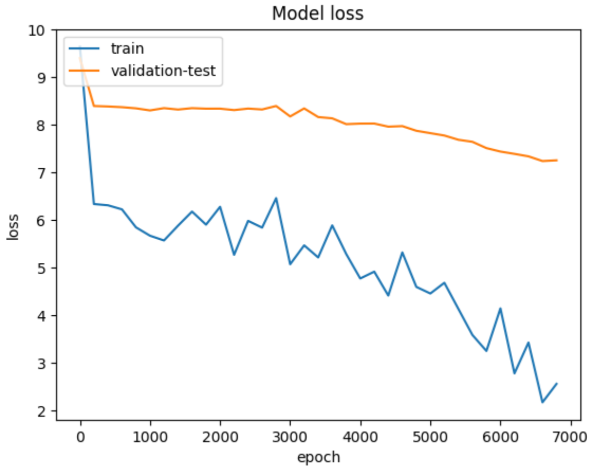


Figure 7: 6layers CONV POOL CONV POOL CONV Dropout FC

.

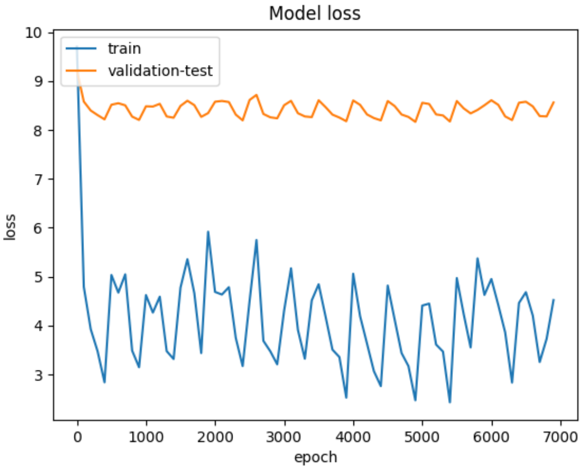


Figure 8: 8layers CONV POOL CONV CONV POOL CONV Dropout FC Dropout FC

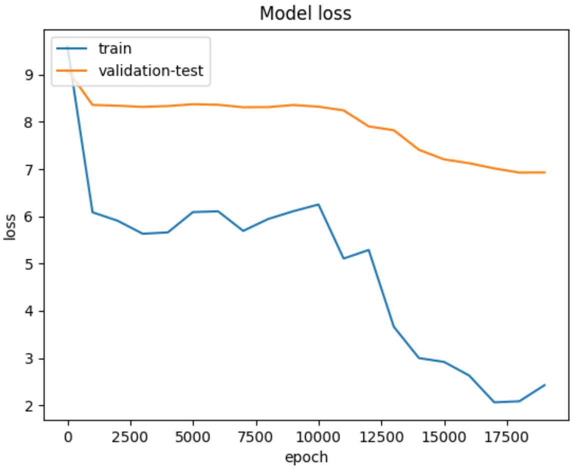


Figure 9: 8layers CONV POOL CONV POOL CONV POOL CONV Dropout FC

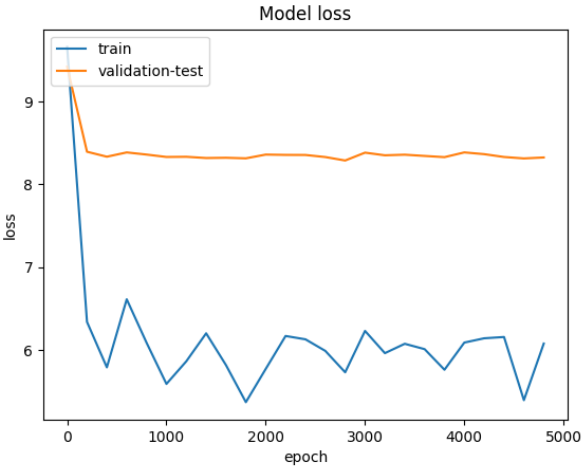
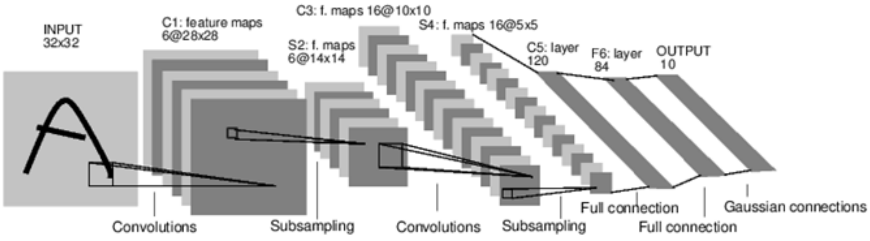


Figure 10: 9layers CONV POOL CONV POOL CONV POOL CONV Dropout FC Dropout FC

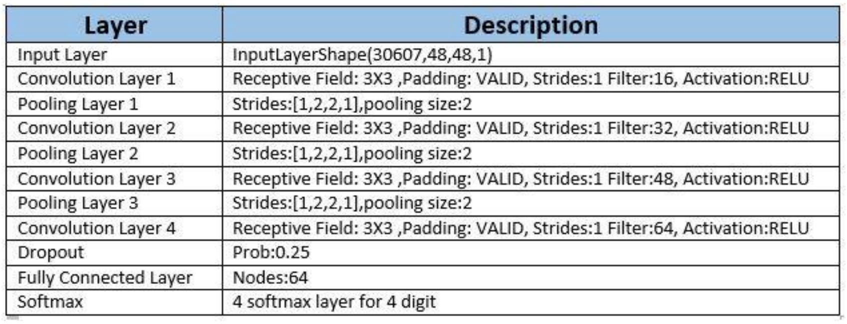
- Final architecture: The steps which we have followed are:
  1.Downloaded and extracted the entire SVHN multi dataset
  2.Cropped images (48x48) to bounded region to find digits.
  3.Found Images with more than 4 digits and deleted it from training set.
  4.Used the preprocessed dataset and trained a multi layer Convolution Neural Network.
  5.Used test data to check for accuracy of the trained model to detect number from street house number image.
  Final Architecture: In final architecture, we have used 8 layered CNN model.



A Full Convolutional Neural Network (LeNet)

Figure 11: Sample Convolution Neural Architecture

Convolution layer –> Pooling layer –> Convolution layer –> Pooling layer –> Convolution layer –> Pooling layer —> Convolution layer —> Fully Connected Layer



Layer	Description
Input Layer	InputLayerShape(30607,48,48,1)
Convolution Layer 1	Receptive Field: 3X3 ,Padding: VALID, Strides:1 Filter:16, Activation:RELU
Pooling Layer 1	Strides:[1,2,2,1],pooling size:2
Convolution Layer 2	Receptive Field: 3X3 ,Padding: VALID, Strides:1 Filter:32, Activation:RELU
Pooling Layer 2	Strides:[1,2,2,1],pooling size:2
Convolution Layer 3	Receptive Field: 3X3 ,Padding: VALID, Strides:1 Filter:48, Activation:RELU
Pooling Layer 3	Strides:[1,2,2,1],pooling size:2
Convolution Layer 4	Receptive Field: 3X3 ,Padding: VALID, Strides:1 Filter:64, Activation:RELU
Dropout	Prob:0.25
Fully Connected Layer	Nodes:64
Softmax	4 softmax layer for 4 digit

Figure 12: Model Description

- Results from different optimization techniques: .

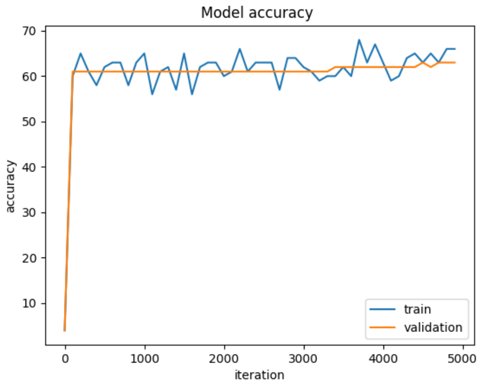


Figure 13: Optimizer-Adam and 6 layer

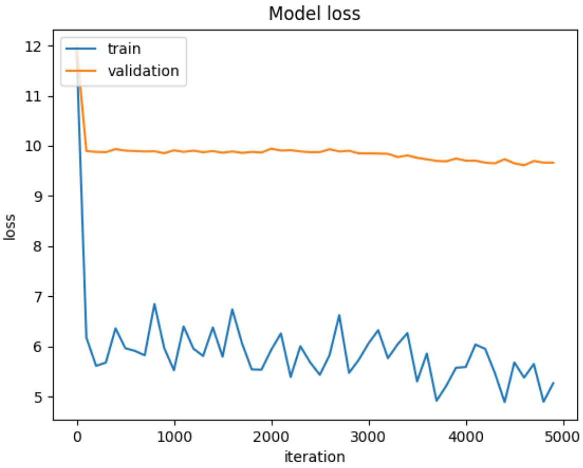


Figure 14: Optimizer-Adam and 6 layer

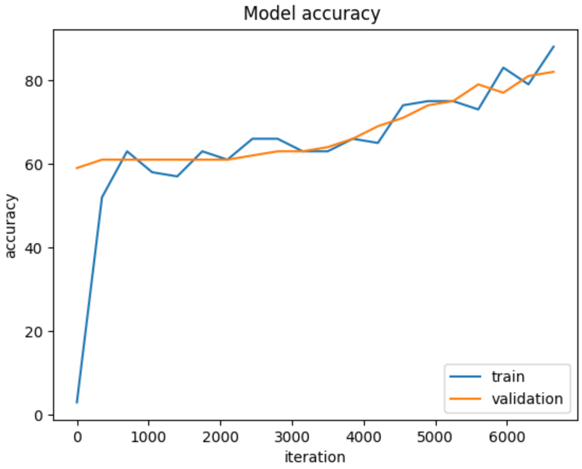


Figure 15: Optimizer-GradientDescentOptimizer and 6 layer

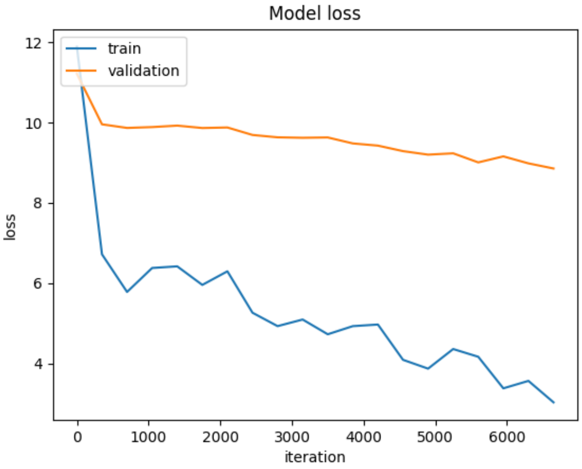


Figure 16: Optimizer-GradientDescentOptimizer and 6 layer

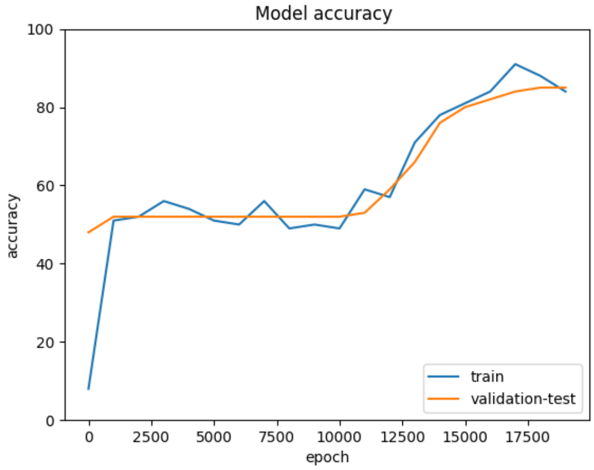


Figure 17: Optimizer-GradientDescentOptimizer and 8 layer

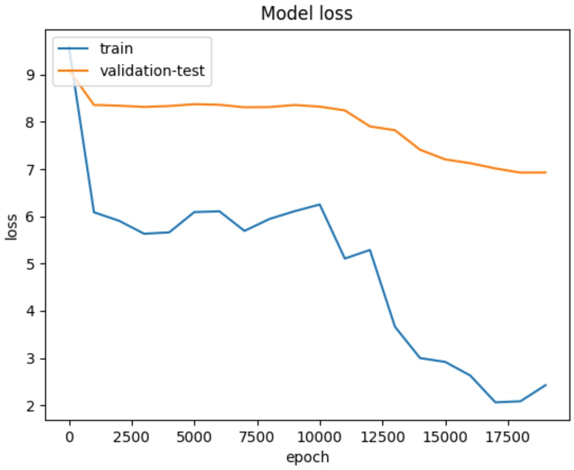


Figure 18: Optimizer-GradientDescentOptimizer and 8 layer

- github location of code :
https://github.com/NikhilJyoti/House-Number-Recognition-from-Street-view.git

3.2 Future work

The final conclusion which we inferred are that when our model consisted of few layers, the accuracy was quite low, on increasing the number of convolution and pooling layers, the accuracy increased to certain extent then finally started decreasing when we further increased the layers. We got the best accuracy of 88% when the total layers were 8.
There can be further improvements in our approach. Firstly, we can improve our accuracy by training our model by using the extra dataset which are available in extra.tar.gz folder. We could also vary the number of layers and number of epochs to improve the accuracy.

4 References

1.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.
2.Ian J. Goodfellow, Yaroslav Bulatov, Julian Ibarz, Sacha Arnoud, and Vinay Shet. Multi-digit num-

ber recognition from street view imagery using deep convolutional neural networks, 2014.
3.http://ufldl.stanford.edu/housenumbers (dataset link)
4.http://cs231n.github.io/