CS6476 Project: Classification and Detection with Convolutional Neural Networks

Prasanna Venkatesan Srinivasan

psrinivasan48@gatech.edu

Abstract

Recognition of digits from street view images may prove to provide valuable information. In the project, an approach to detect and classify multi-digit sequences has been proposed. The solves the problem using Maximally Stable Extremal Regions (MSER) algorithm and Convolutional Neural Networks (CNN) to accurately detect and classify the numbers in the input images. The convolution neural network is trained on Google's Street View House Numbers dataset [1]. The proposed CNN converges in less than 10 epochs and provides an accuracy of around 95%. The custom-built model has been found to be on par with the VGG 16 built from scratch and the pre-trained VGG 16 models.

Literature Survey

The problem is composed of two sub-problems – Object detection and Object recognition. Sequences of numbers must be detected in a real-world image and then classified. [2] proposes a model to identify the centroid point of individual digits in an image, construct regions of interest and classify using a convolutional neural network. [3] proposes a technique called Region based Convolutional Neural Network (RCNN) which uses generative search for region proposal and CNN for object classification.

Modern approaches combine the object detection and recognition to a single model. [4] introduces a model called Fast RCNN where a single CNN is composed of fully connected layer using softmax activation function for predicting the output classes and a bounding box regressor to detect the regions of interest in the input images. [5] modifies Fast RCNN and proposes a model called Faster RCNN which sandwiches another pretrained model (like VGG16) for region proposal and a FC layer using softmax activation for classification.

[6] explains an algorithm called You Only Look Once (YOLO), which is different from the region-based algorithms cited above. The model splits the image to an NXN grid and assigns m bounding boxes to each grid. The CNN is trained to output the class probabilities and the offset of each of the bounding boxes. Those probabilities that exceed a threshold are selected and provided as the output.

Proposed Methodology:

Training the custom model:

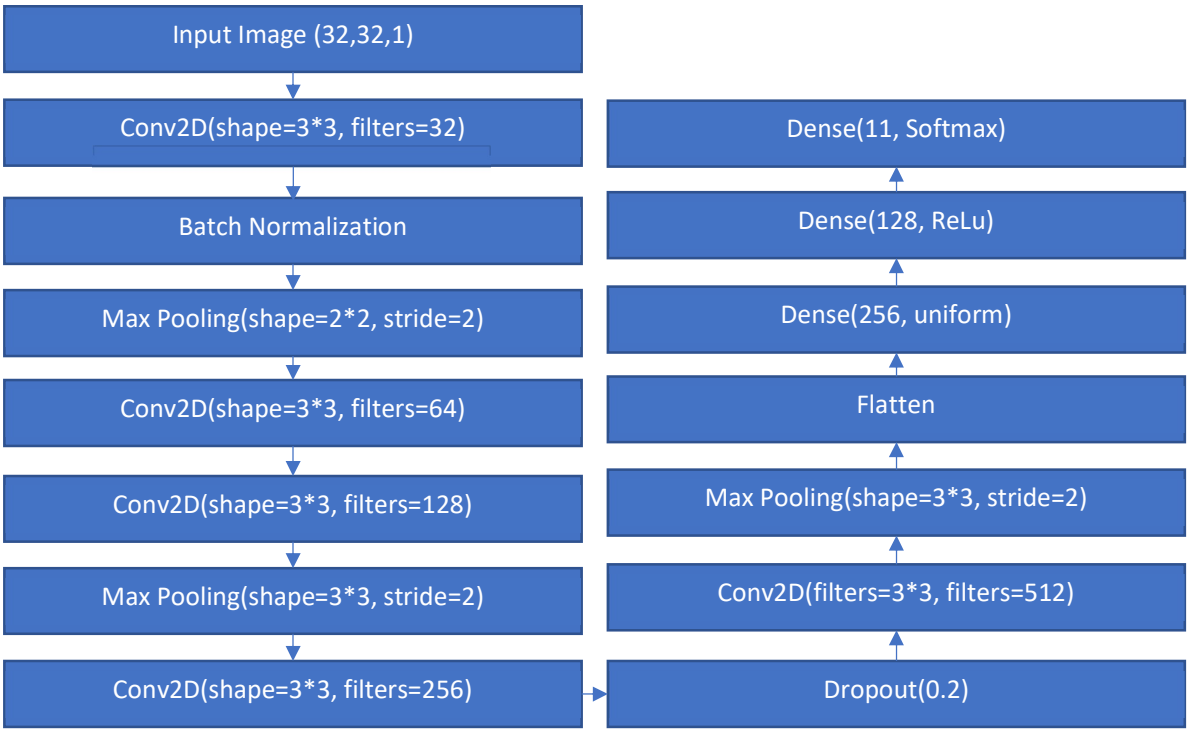


Fig 1: Architecture of the digit recognition CNN

The architecture of the digit recognition model is depicted in Figure 1. A model is built using Convolutional Neural Network of 5 convolution layers, 3 max pooling layers, 1 batch normalization layer, 1 dropout layer and 3 dense layers. ReLu has been used an activation function in all the max pooling and convolution layers. The Format 2 dataset of Google's Street View House Numbers (SVHN) was used for the model training purposes. The images are 32 X 32, single digit images that have been annotated with classes 1 to 10, 1-9 denoting the digits 1 to 9 and 10 denoting the digit 0. For this problem, the digit recognition model should be constructed such that it can also classify non-digit classes.

The input layer is provided with the grayscale image. The color information is discarded from the input image, as it contributes very little towards the classification despite increasing the complexity. The input to the model is a grayscale image and the output in one of the classes between 0-10 where 0-9 classes is mapped to digits 0-9 and class 10 indicates non-digit. The loss function is set to categorical cross entropy, which is given by,

$$-\sum_{c=1}^{M} y_{o,c} \log(p_{o,c})$$

Where p – probability of image o belonging to class c, M - the number of classes, y – binary indicator to denote that image o has been correctly classified to class c. Stochastic Gradient Descent Optimizer has been used for the purposes of training. The training was stopped when the cross-entropy loss of the validation dataset did not improve over 5 successive epochs. The model was built using Keras and trained on Nvidia GeForce GTX 1080Ti GPU. The model took 8 epochs to converge and the training time was around 10 minutes.

Object detection:

The object detection algorithm to identify regions containing numbers in the input images must be position, scale and shape invariant. Sliding window approaches are most commonly used, where a window of a fixed size is slid over the entire image and over successive levels of the gaussian pyramid of the image. The model can be trained to classify if the regions contain the object of interest. The problem with this approach is that, the number of regions that a sliding window algorithm generates in the order of thousands. This approach extremely slow, rendering them incapable of handling real-time detection and recognition.

To facilitate near-real-time object detection, Maximally Stable Extremal Regions (MSER) can be used. MSER is invariant of changes to illumination, shape and scale. The object detection workflow begins by feeding the input grayscale image to MSER algorithm. MSER produces a set of regions indicating the blobs present in the input image. It must be noted that, very small, very large or wider regions may not contain a single digit and hence are discarded. The regions predicted by MSER are then fed to the proposed neural network model. The non-number regions predicted by CNN (class -10) are rejected, and numeric regions are preserved as regions of interest. The regions of interest contain overlaps and hence the output regions are obtained by sending ROIs to non-maximal suppression algorithm.

Results:

The loss and accuracy of the custom digit recognition model (Figure 1), VGG 16 pre-trained and VGG 16 models are tabulated below.

Model	Training Accuracy	Validation Accuracy	Test Accuracy	Loss (Test)	Number of Epochs to Train
Custom CNN	96.69%	95.82%	94.93%	0.2285	8
VGG 16	99.25%	97.91%	94.80%	0.2576	15
VGG 16 pretrained	98.81%	97.46%	95.90%	0.1707	8

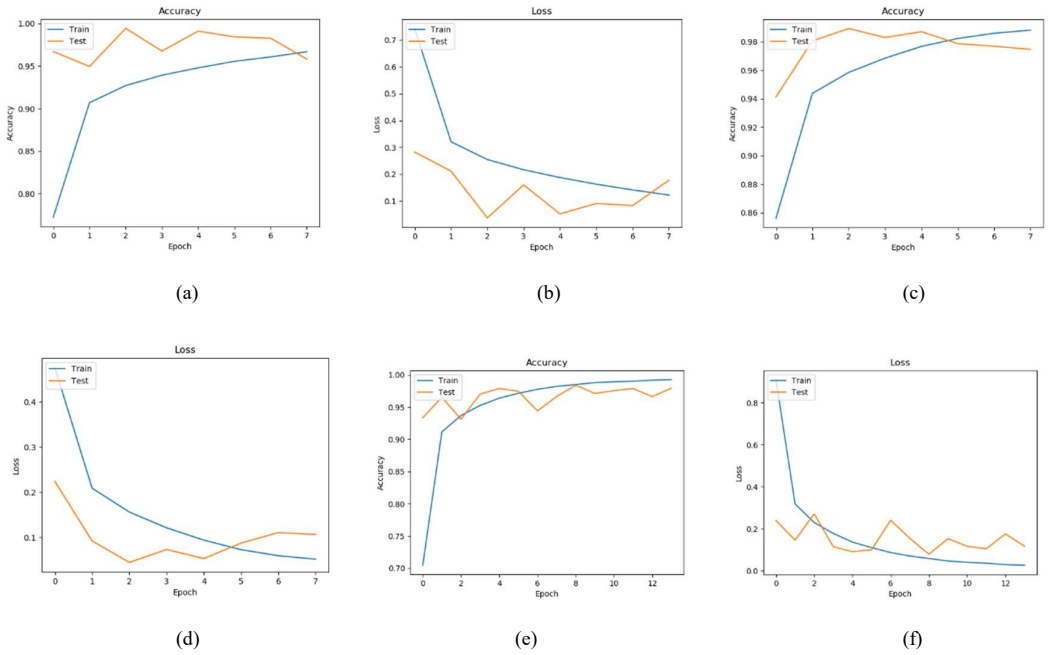


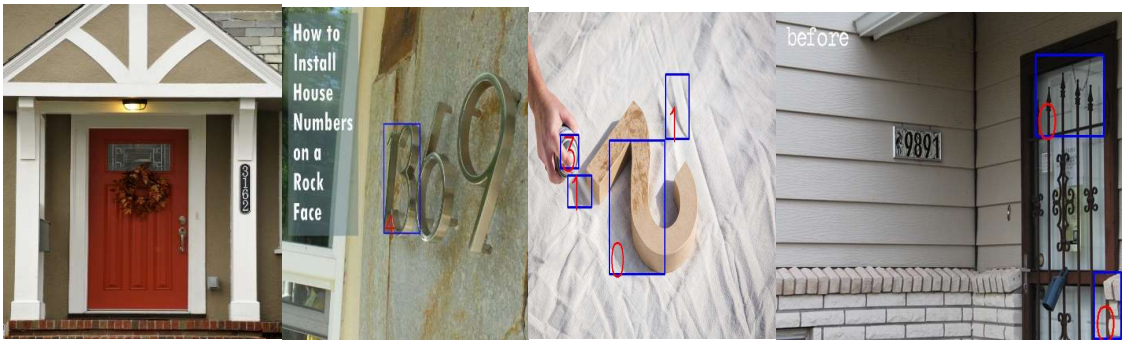
Figure 2: (a), (c), (e) – accuracy vs epoch plot of Custom, VGG16 pretrained and VGG16 models. (b), (d), (f) – loss vs epoch plot of Custom, VGG16 pretrained and VGG16 models

The loss and accuracy plot of the three models as a against epochs are given in Figure 2. It can be noted that performance of the proposed model is on par with the other two models.

Some of the images where the workflow performed well.

Figure 3: Correctly classified images

Some of the incorrectly labelled images are given below.



Conclusion:

The accuracy of the model can be improved by adopting a modern approach of object detection and detection cited as [4],[5],[6]. The model and weights file of the custom built CNN has been provided in https://drive.google.com/drive/folders/1gXxFgreVXnuddnRfXK_DTqxQnaa3TeUm . A demo of how the approach works in practice can be seen in the video https://youtu.be/RHfe8vs8S0Q.

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, Vinay Shet. Multi-digit Number Recognition from Street View Imagery using Deep Convolutional Neural Networks. arXiv preprint arXiv:1312.6082v4. 2014.

[3] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. arXiv preprint arXiv: 1311.2524v5, 2014.

[4] Ross Girshick. Fast R-CNN. arXiv preprint arXiv:1506.01497, 2015

[5] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. arXiv preprint arXiv:1506.01497, 2015

[6] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You Only Look Once: Unified, Real-Time Object Detection. The proceedings of The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.