Machine Learning Engineer Nanodegree

Capstone Project

Kenta Suzuki

July 14, 2017

I. Definition

Project Overview

Image recognition has gained a lot of attention since 1990s when an algorithm called 'Convolutional Networks' [1]was used to read checks. Accuracy of the networks has been improved over the last few years. For example, an error rate of winning algorithms of ImageNetILSVRC where researchers submit their solutions to the image classification improved from over 26% to over 3% in 5 years.

In this project, convolutional networks are used to classify two types of pictures; a picture containing a invasive species and a picture not containing the species.

Problem Statement

Computer vision can be extended to natural environment. For example, there are many species that have harmful effects on the ecosystem. Currently, trained scientists visit some areas and check the existence of these harmful species. They would be able to concentrate on the research where some creativity and tacit knowledge are required if a reliable algorithm automate this classification process.

Thus the purpose of this project is to develop an algorithm which classify the images with low error rate in order to reduce the burden imposed on the scientists.

Metrics

Metrics used are as follows:

1) Binary accuracy: the proportion of samples an algorithm correctly classifies. This is a useful metric not only for measuring the how many sample are correctly classified but for the detection of overfitting which often occurs when small data sets or complex algorithms are used.

2) Binary cross-entropy: loss function to minimize. This is useful for deciding an appropriate learning rate of optimization.

$$C = -\frac{1}{n}\sum_{x}[y \ln a + (1-y)\ln(1-a)],$$

Figure 1: Formula of binary cross-entropy

3) AUC: This is useful especially when the data sets are imbalanced such as many samples are drawn from one class and few from the other class. Predicting every sample as major class will result in high accuracy rate in such data sets and a trained algorithm would not classify the minority class correctly.

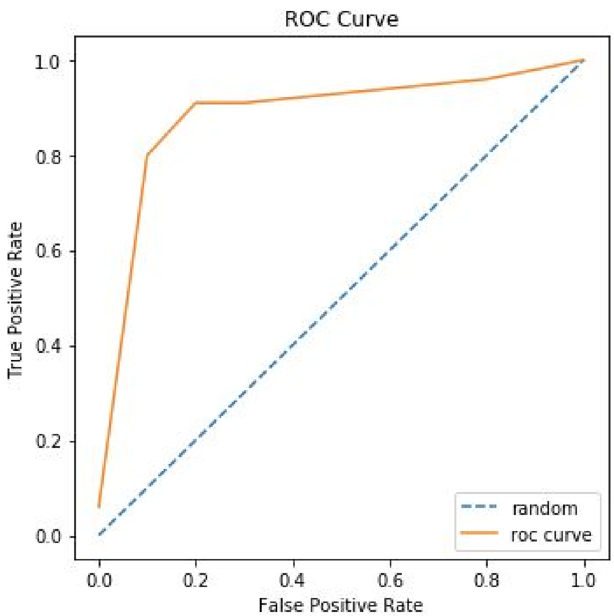

Figure2: ROC curve

II. Analysis

Data Exploration

The datasets are pictures (RGB format) collected in a Brazilian national forest. Some of the images contain invasive species, *Hydrangea*, and the objective is to predict the presence of this species. This datasets are publically available on Kaggle competition[2].

Training data contains 2295 images, testing data contains 1531 images. Labels for the training data are given but not available for testing data.

Data sets are imbalanced. There are 847 images with no Hydrangea, and 1448 images with Hydrangea appearing.

Figure 3. (left) No presence of Hydrangea

(right) Presence of Hydrangea

Exploratory Visualization

To understand the class distribution visually, bar chart for sample proportion for each class is plotted below.

Figure 4: Sample proportion for each class. Ratio is about: 3.5:6.7

Algorithms and Techniques

Convolutional neural networks are used for this problem. They have a main advantage over fully connected neural networks; They retains the spatial information of the image.

Possible parameters to consider are: activation function, optimizer, depth of net architecture, regularization

Benchmark

As a benchmark, simple convolutional networks with six layers are used. Neither data augmentation nor class weight for tackling the imbalance problem are used. This model is used for a first attempt since it becomes clear how AUC score is if not taking class imbalance into account within reasonable time.

The AUC score for this model is 0.817. Reliability of other models are compared to this score. Final training loss is 0.029, validation loss is 0.61, training accuracy is 0.99, validation accuracy is 0.82. This model overfitting on the data. This might be that data sets are small so that it captures the idiosyncrasy of training data. For every test sample, this model predict the ex

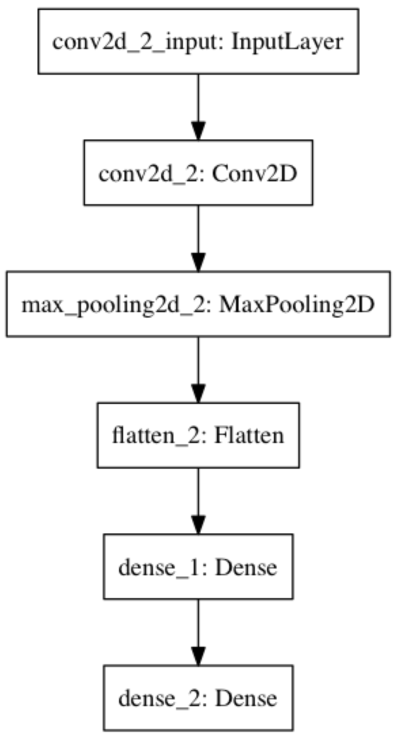

Figure 5: Architecture of convolutional networks as a benchmark

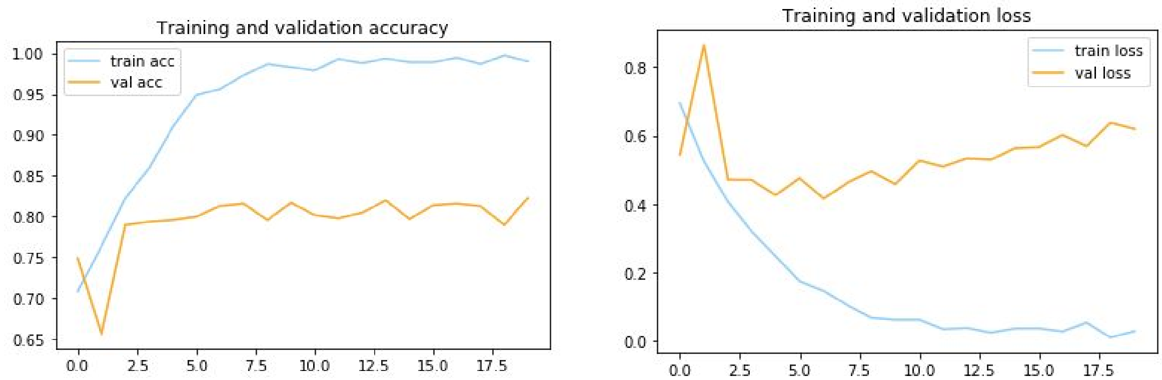

Figure 6: Rraining, validation accuracy (left)

Training, validation loss (right)

III. Methodology

Data Preprocessing

Each batch samples are normalized by setting their mean to 0 and divide them by its standard deviation so that it ranges from 0 to 1.

For addressing the class imbalance, data augmentation and class weights are used. Since large data sets are required for convolutional networks to work properly, undersampling is not conducted.

Weights are initialized from the gaussian distribution with mean 0 and with standard deviation square root of 2 over the number of units, as discussed in He et al[3].

$$\frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \times \sqrt{\frac{2}{\#of\,inputs}}$$

Figure 7: He initialization

Implementation

All models are implemented by Keras as Tensorflow backend. Transfer learning is used for Pre-trained models on imagenet. Due to the absence of GPU, this method is used as a feature extraction.

Refinement

In order to get out of plateau when minimizing the cost function, learning rate is divided by 10 when validation loss is not reduced for certain periods such as 2 successive epochs.

Batch normalization is added to the convolutional nets with relu function. This solves, internal covariate shift and make the convergence faster.

L2 regularization is added to the dense layer in order to prevent overfitting.

IV. Results

Model Evaluation and Validation

The best model is an ensemble with weighted average of the seven models that result in top AUC score . Since each model captures the different traits of the images, an ensemble method often results in a better score than single model does. Models with L2 regularization are not performed well on the test data. Even for training and validation data, accuracy for each does not exceed 90% which most of the models implemented without L2 regularization achieved. This is considered to be caused by the small datasets and using L2 regularization results in underfitting.

Justification

AUC score of the ensemble model is 0.958. This is 16.8 % improvement than the score produced by benchmark model. The seven models used are: VGG16 with dropout rate 0.5, VGG16 with dropout rate 0.7, Convolutional Networks with 13 layers, Resnet, VGG19, InceptionV3, Xception. Models except the convnets with 13 layers are trained on imagenet and applied to this binary classification as transfer learning. The AUC score is significant enough to solve the objective; it gives a score better than a random predictor does and it minimizes the false positive rate.

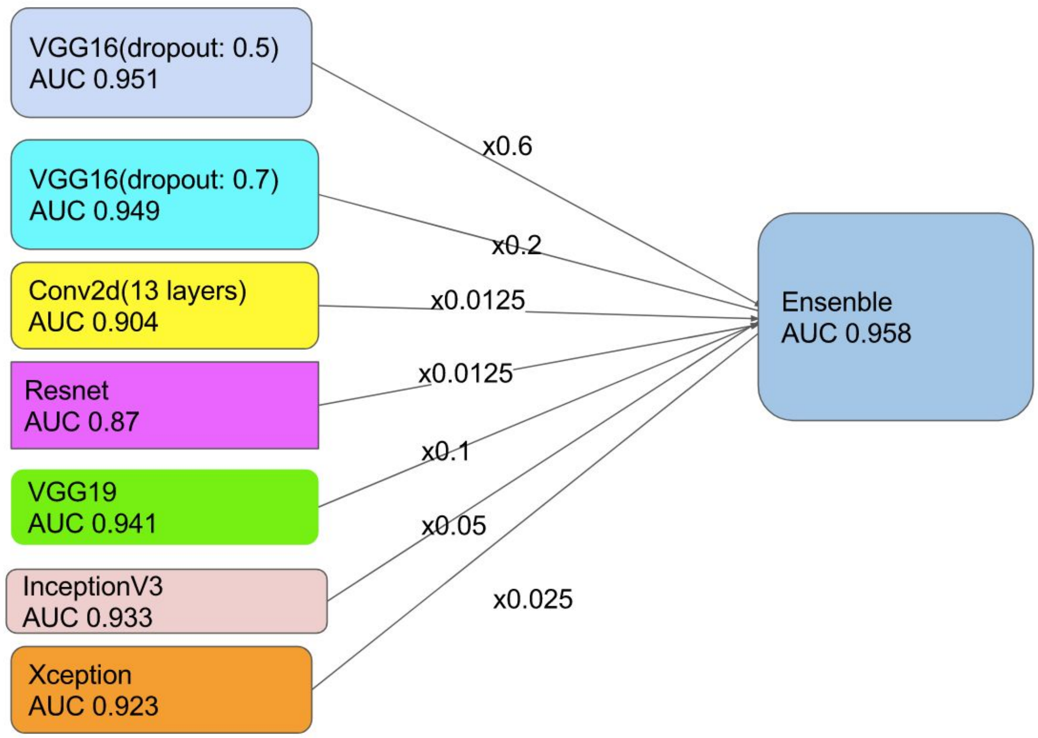

Figure 8: Final model, ensemble model of 7 models. Each model is weighted for its importance.

V. Conclusion

Visualization

Visualization of filters of convolutional neural networks makes it possible to see which layer learns what characteristics of an input image. For visualization, convolutional neural networks with 5 convolutional layers are used. Below are images for what each 5 convolutional layer recognizes the image of invasive species. Although there is no clear pattern for each layer, first layers seems to detect tiny edge patterns and last layer sees more abstract image.

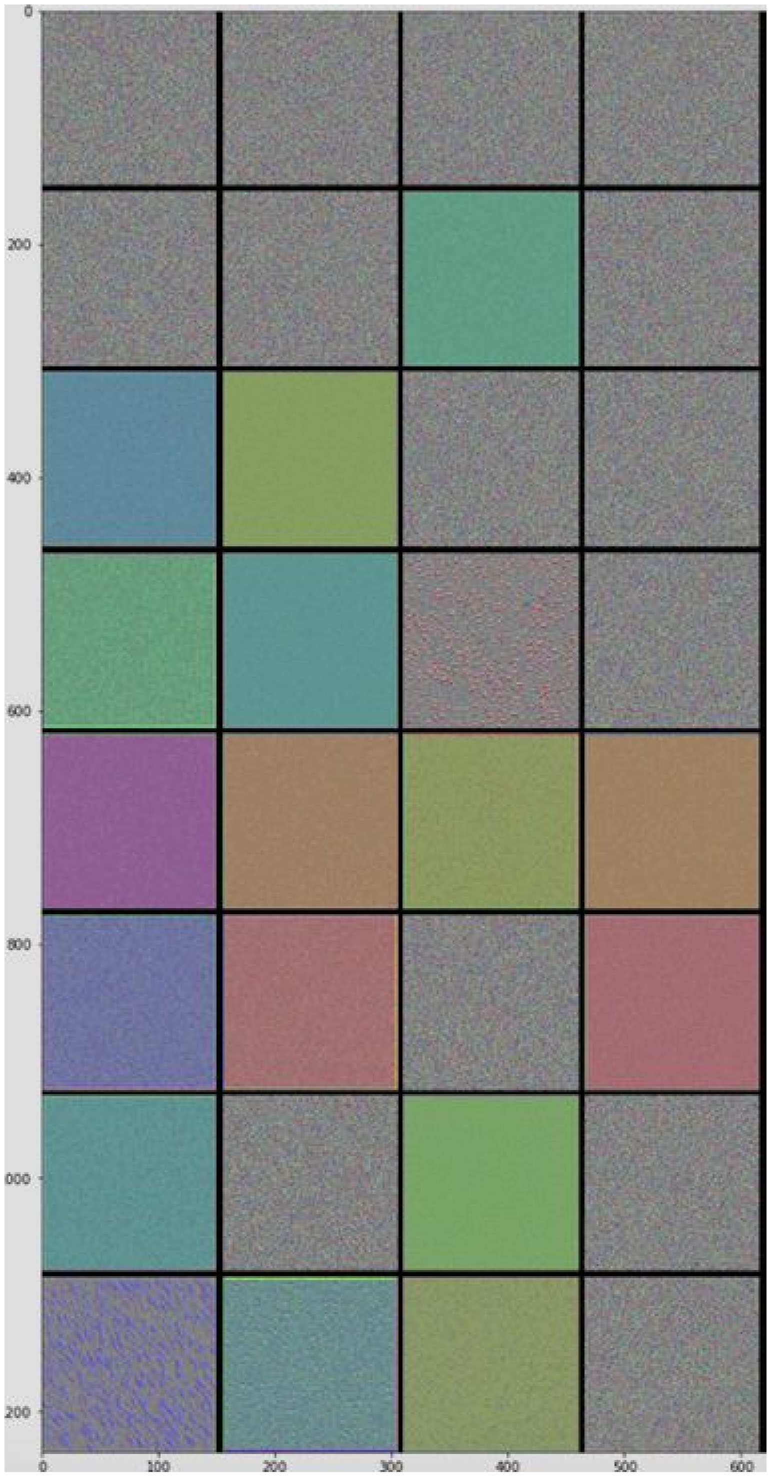

Fig 9 : Filter patterns for convolutional layer 1

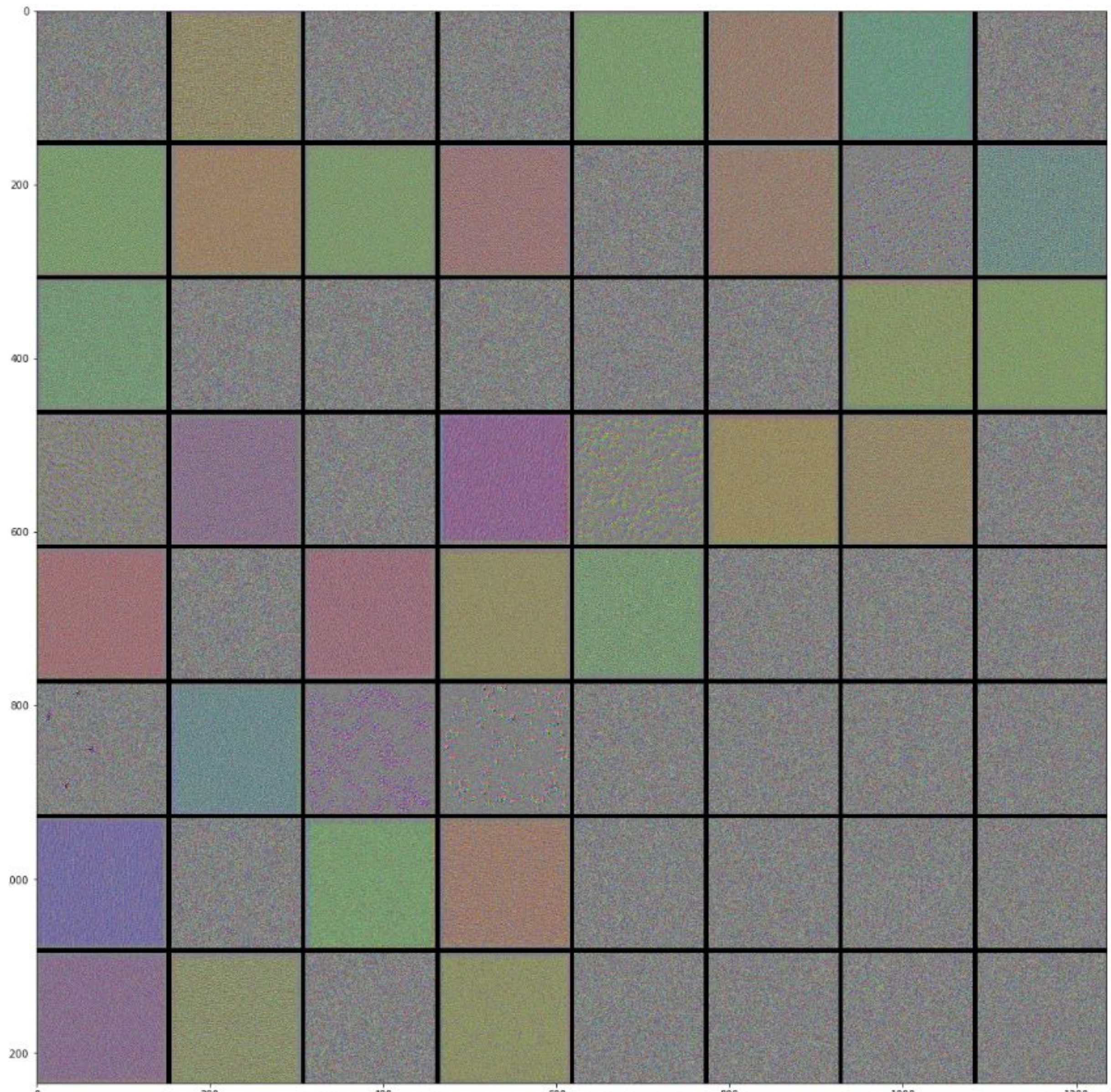

Fig 10: Filter patterns for convolutional layer 2

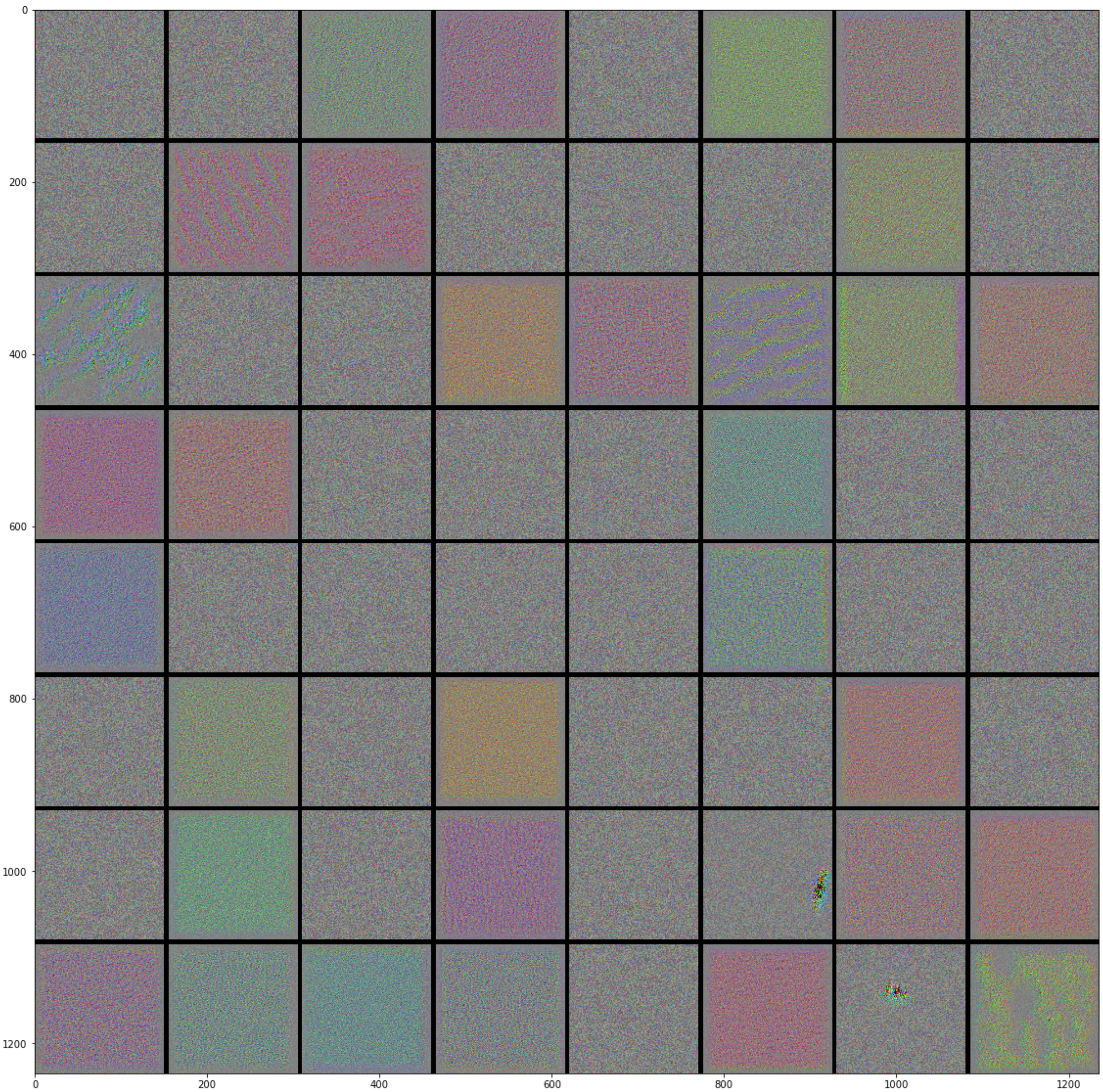

Fig 11: Filter patterns for convolutional layer 3

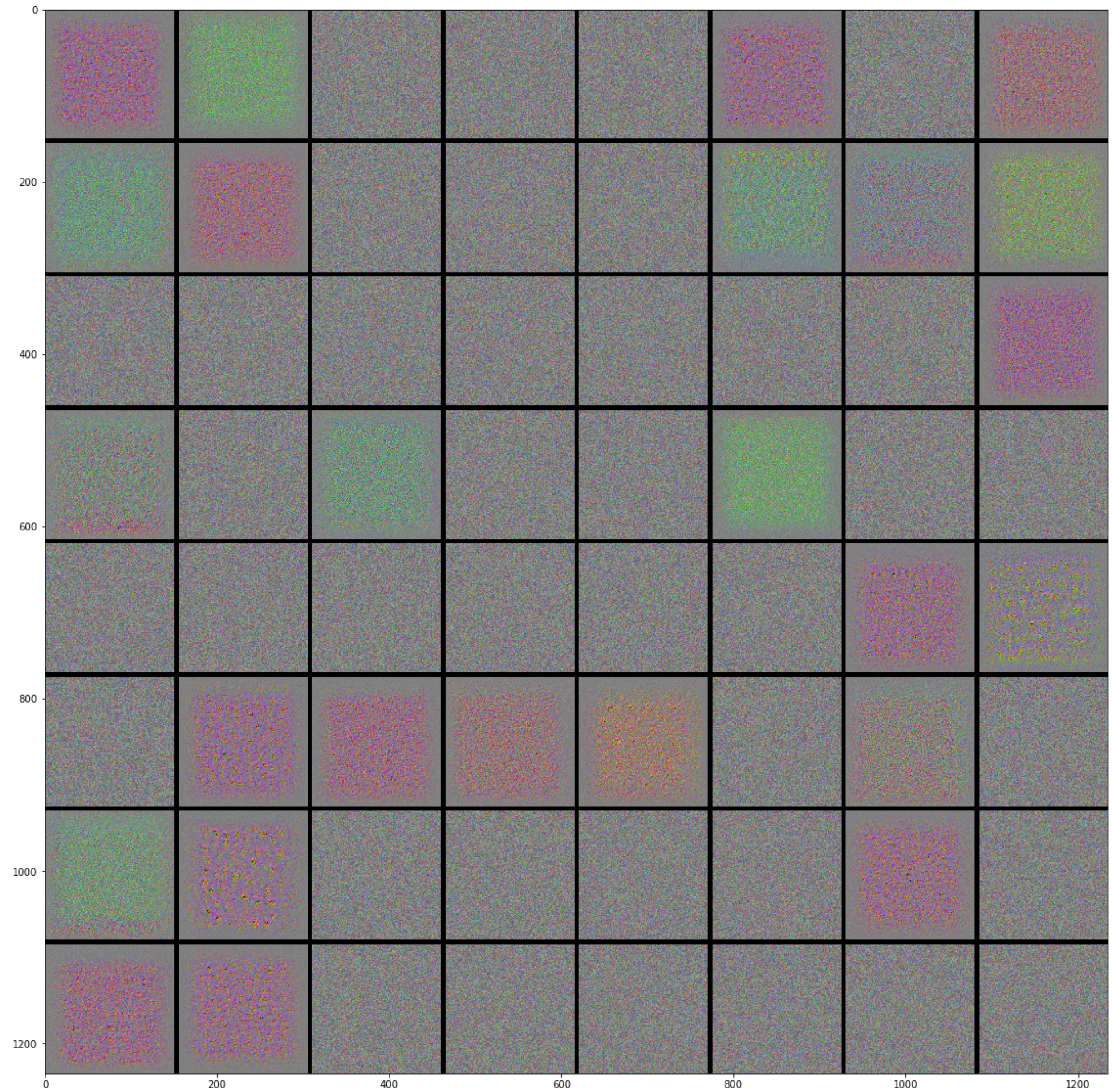

Fig 12: Filter patterns for convolutional layer 4

Reflection

In theory, elu activation produces better result than relu does since it helps vanishing gradients problem[4]. However, any improvement cased by chaining relu to elu was not seen. The size of data sets are small and overfitting to some extent seems inevitable.

Improvement

Since all models were trained without GPU, training convolutional base and conducting fine-tuning were impossible given its computational expensiveness. Since pre-trained models are tailored to the images of imagenets, changing the architectures of the networks for this specific images will improve AUC score.

Collecting more data from other sources might improve the accuracy especially for ResNet since data size is too small for its number of complex architecture with many parameters.

References:

[1] LeCun.Y. LeNet-5 - Yann LeCun http://yann.lecun.com/exdb/lenet/

[2] Invasive Species Monitoring | Kaggle https://www.kaggle.com/c/invasive-species-monitoring/data, 2017

[3] He.K. , Zhang.X. , Ren.S. , Sun.J, Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification https://arxiv.org/abs/1502.01852, 2015

[4] Clevert.D., , Unterthiner.T. , Hochreiter.S. https://arxiv.org/abs/1511.07289 , 2015