Different Weights Initialization Methods of D-CNN on Distracted-driving Detection

Jing Zhang, Cheng Lyu, Yanyu Hu, Zhao Zhao, Yanze Wang

Abstract

Each day in the United States, more than 1,000 reported crashes involve a distracted driver. Distracted driving is driving while doing another activity that takes the driver's attention. It is thus very important to develop automatic software to detect distracted driving behaviors with high accuracy. In this research, we use deep neural networks to classify images of drivers[1] into 10 different classes, where one of them indicates non-distracted driving and the other nine indicate different distracted behaviours such as holding a mobile phone on one hand. The neural network is designed to be simple but similar to VGG[2] structure and implemented by using Keras framework[3]. Furthermore, we tried and compared five advanced initialization approaches by looking at their training efficiency. Results showed that our neural network could classify the driver behaviors highly accurately and LSUV[4] init turns out to be the optimal initialization technique in this application.

Neural Network Structure

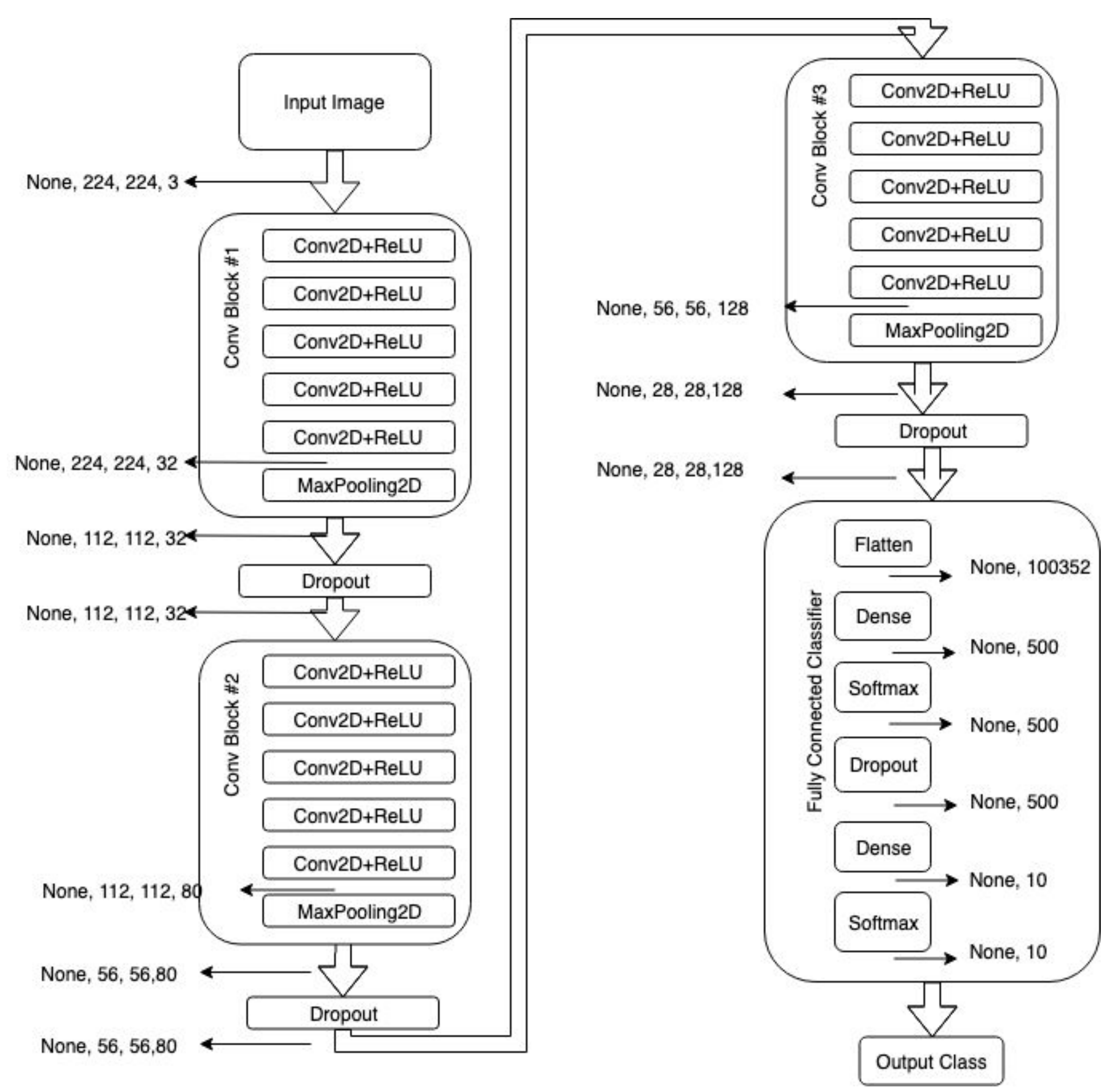


Fig1. The designed neural network structure

Initialization Methods

1. **Xavier normal initializer[3]:** a truncated normal distribution centered at 0

$$\sigma = \sqrt{\frac{2}{fan_{in}+fan_{out}}}$$

2. **Xavier uniform initializer[3]:** a uniform distribution within

$$\left[-\sqrt{\frac{6}{fan_{in}+fan_{out}}}, \sqrt{\frac{6}{fan_{in}+fan_{out}}}\right]$$

3. **He normal initializer[3]:** a truncated normal distribution centered on 0 with

$$\sigma = \sqrt{\frac{2}{fan_{in}}}$$

4. **He uniform initializer[3]:** draws samples from a uniform distribution within

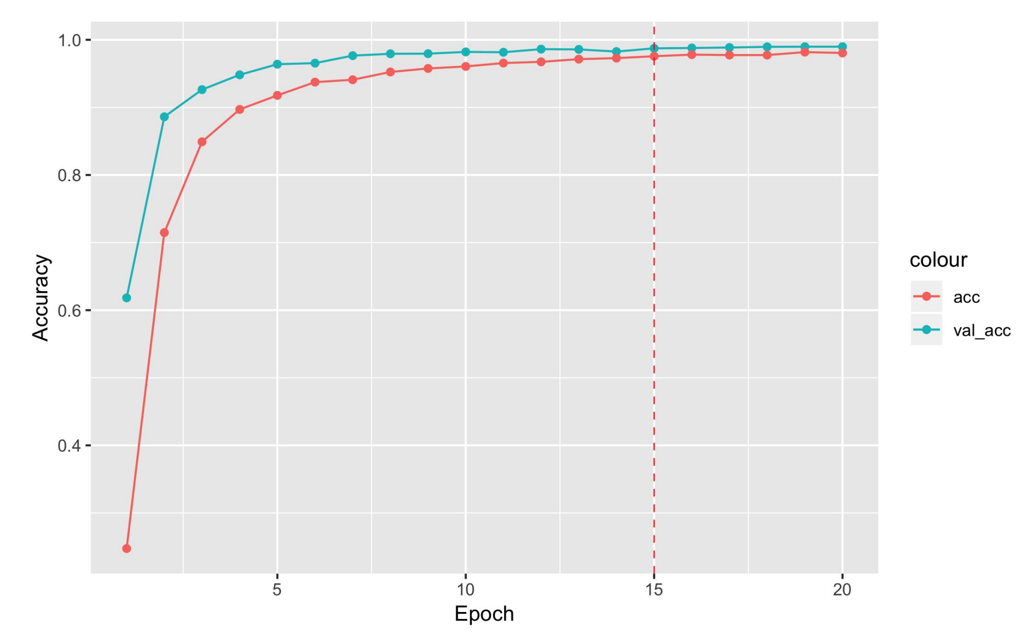
$$\left[-\sqrt{\frac{6}{fan_{in}}}, \sqrt{\frac{6}{fan_{in}}}\right]$$

Note: fan_in and fan_out are the number of input and output units of the layer.

5. **Layer-Sequential Unit-Variance(LSUV)[4]** orthogonal initialization:

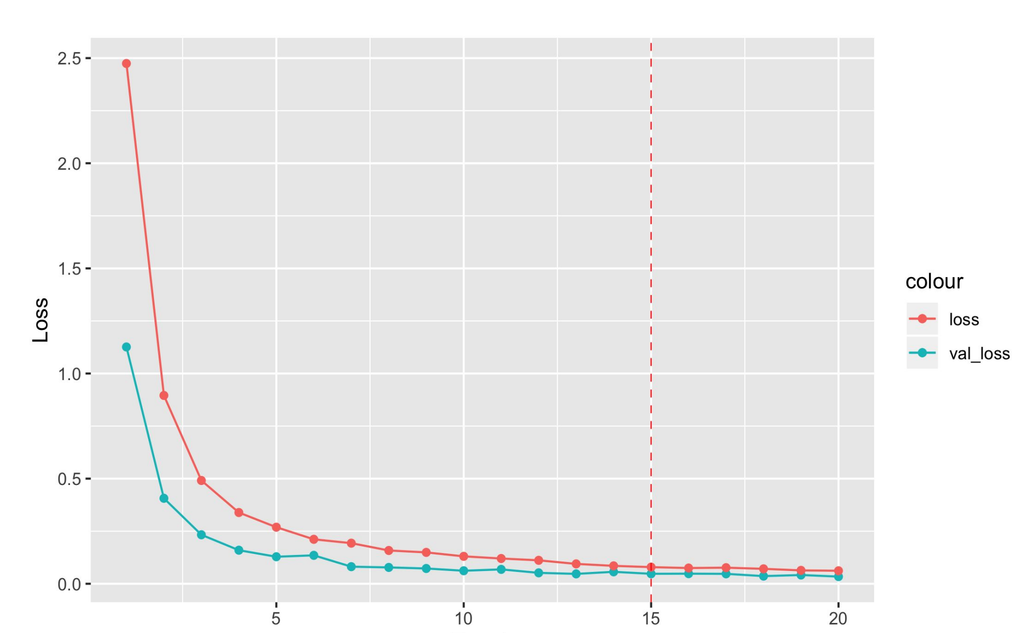
Preinitialize network with orthonormal matrices as Saxe et al[5]
for each layer L **do**:
    while $|Var(B_L) - 1.0| >= Tol_{var}$ and $(T_i < T_{max})$ **do**:
        do forward pass with a mini-batch
        calculate $Var(B_L)$
        $W_L = W_L / sqrt(Var(B_L))$
    end while
end for

Results



The validation accuracy converges about 99.10%.

Fig2. Accuracy at each epoch with LSUV init



The validation loss converges about 0.04.

Fig3. Loss at each epoch with LSUV init

Note: The validation accuracy is better than training accuracy because the convolution blocks are respectively followed by a dropout layer. But Keras fitting would use all features to do validation, leading to higher accuracies.

initialization methods	Xavier normal	Xavier uniform	He normal	He uniform	LSUV
total training time (min/20 epochs)	64.92	68.80	70.16	68.60	69.65
average training time (min/ epoch)	3.25	3.44	3.50	3.43	3.28
initial accuracy (%)	62.83	58.47	72.44	56.67	61.80
converged accuracy (%)	98.60	98.70	98.65	98.80	99.10

Table1. Performance comparison among the five initialization methods

Note & Explanation:
The training times are counted over 20 training epochs.
The model trained with the LSUV initializer achieved the highest validation accuracy of 99.10 percent. .

Conclusion & Future Works

The designed deep neural network based on the structure of VGG is proved to perform well in classifying the 10-class driving behaviour image data set. And the LSUV initialization approach turns out to be the best among the five experimented methods. As a result, our network structure and the specially chosen initializer may have great significance of providing helpful insights for future software development.

Although we achieved high classification accuracy in this experiment, many other distracted driving behaviors have not yet been covered in our training data set, and the pretrained model may not be able to robustly deal with complex situations. In future works, we can explore other neural network structures and improve more components such as regularization, nonlinearity, etc. Moreover, we may invoke computer vision methods to deeply understand the image or video semantics. For example, we can locate effective feature points (such as driver's hands and the steering wheel) on the image to analyze whether the driver has put his/her hands away from the steering wheel.

References

[1] State Farm Mutual Automobile Insurance Company. (2016, April 5). State Farm Distracted Driver Detection Dataset. Retrieved from https://www.kaggle.com/c/state-farm-distracted-driver-detection/data

[2] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

[3] Keras: The Python Deep Learning Library https://keras.io/ Accessed 16 Apr. 2019.

[4] Mishkin, D., & Matas, J. (2015). All you need is a good init. *arXiv preprint arXiv:1511.06422*.

[5] Saxe, Andrew M., James L. McClelland, and Surya Ganguli. "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks." *arXiv preprint arXiv:1312.6120* (2013).