VEHICLE MAKE AND MODEL CLASSIFICATION USING CONVOLUTIONAL NEURAL NETWORKS

Presented By

Rajitha Meka

Syed Hasib Akhter Faruqui

The University of Texas at San Antonio™

PROBLEM DEFINITION

- Existing convolutional neural network models that are being used to predict the generic images proved to achieve high accuracies
- These models when applied to fine grained images, the accuracy drops significantly
- This creates a scope to improve the accuracy of the model
- With the limited time, computational power and size of the dataset this is a challenging task to train the model to get high accuracy

APPLICATION

- Trained model with large dataset, giving high accuracies helps mainly with following applications:

  ✓Traffic control and management

  ✓Detecting the car at times of accident

  ✓Identifying the car from the images of security cameras in the situations like car theft

INTRODUCTION

- Globally, lot of automotive companies are producing cars with almost a new model every year with improved design and efficiency
- Each car make will have different models matching the requirement and taste of the customer
- Cars are classified as fine grained images as they have subtle differences between classes
- Differentiating among cars is a fairly easy job for a human eye, but it is quite hard for a machine given a picture of any viewing angle of the car

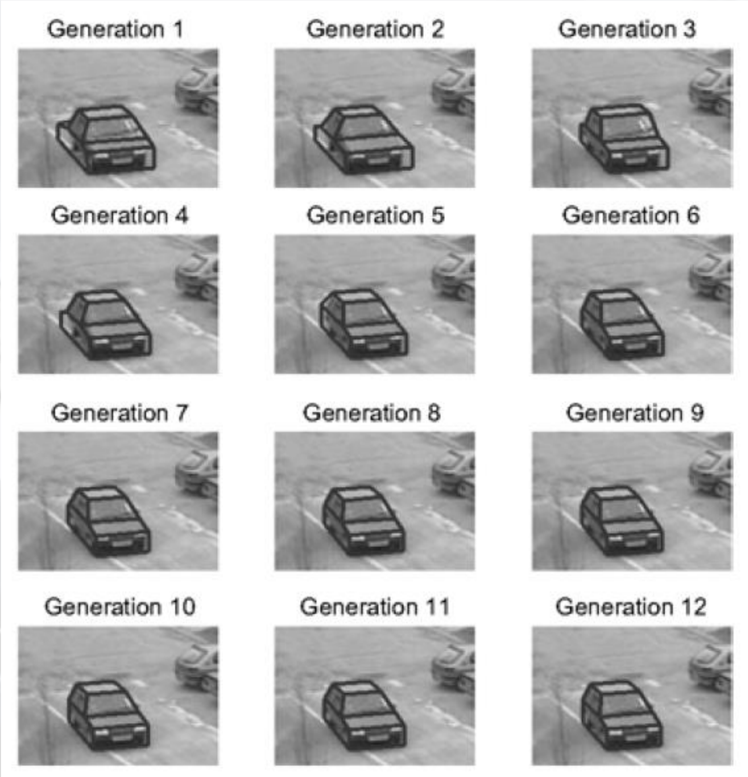
Different view angles of the car

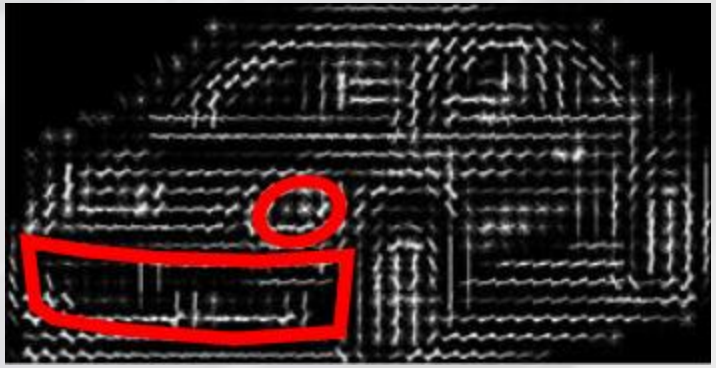
OBJECTIVE

- To identify the Make and Model of Cars using Convolution Neural Network (CNN)
- Implementation of a state of art CNN model (GoogleNet)
- Design and Implementation of a New CNN architecture
- Testing and Comparing the models for the same test data

PREVIOUS WORK

- Identification using Histogram of Oriented Gradients (HOG Feature)
- Fitting wireframe models to car structures to identify them based on structure
- AlexNet
- VGGNet
- GoogleNet





PROJECT DESCRIPTION

- Stanford Cars Dataset (http://ai.stanford.edu/~jkrause/cars/car_dataset.html)
- Images – 16,185, Classes – 196
- GoogleNet and NhLNet (new Designed Architecture) was implemented
- Two separate datasets created
  - ✓ 25 classes
  - ✓ 196 classes
- Due to time limitations, the models using dataset of 196 classes were trained for a shorter period of time

196 Classes
- GoogleNet
- NhLNet

25 Classes
- GoogleNet
- NhLNet

STRUCTURE OF THE IMAGES

ACURA

2012

RL SEDAN

TL SEDAN

TSX SEDAN

DATA

- Pre-Processing Steps
  - ✓ Conversion of jpg to png
  - ✓ Identifying black and white images and removing from the dataset
  - ✓ Cropping all images according to the bounding boxes given to remove the noise
  - ✓ Resizing all images to same size: 227*227*3
  - ✓ Splitting the data in the proportion of 70:30 for train and test purposes

Before and After Pre-Processing

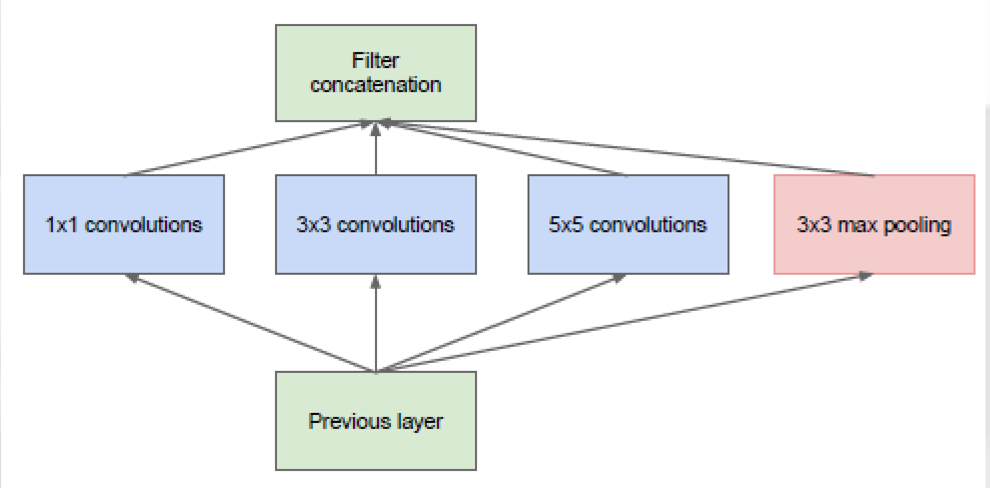
Sample Black and White Images

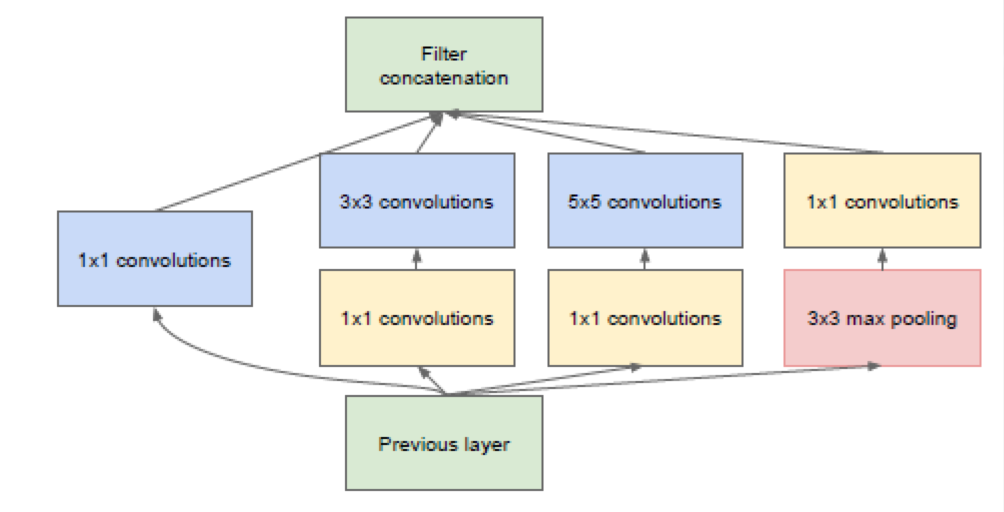
DATA (CONT'D)

- Creation of dataset with 25 classes
  - ✓Classes with high number of images were selected
  - ✓All 25 classes have greater than equal to 90 images per class with total 2334 images
  - ✓Data augmentation was done to increase the size of the dataset with 25 classes

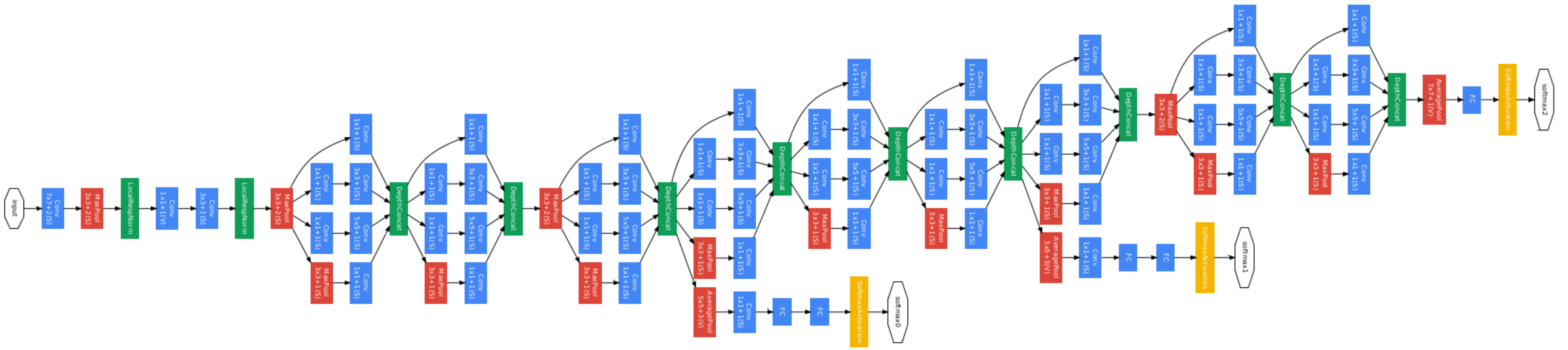
IMPLEMENTATION (GOOGLENET/INCEPTIONNET)

- 22 layers
  - ✓9 inception layers
  - ✓2 stand alone convolution layer
  - ✓Max Pooling
  - ✓Normalization
  - ✓1 Fully Connected Network
- Dropout (40% rate)
- 3 output layers
  - ✓We only used the last output





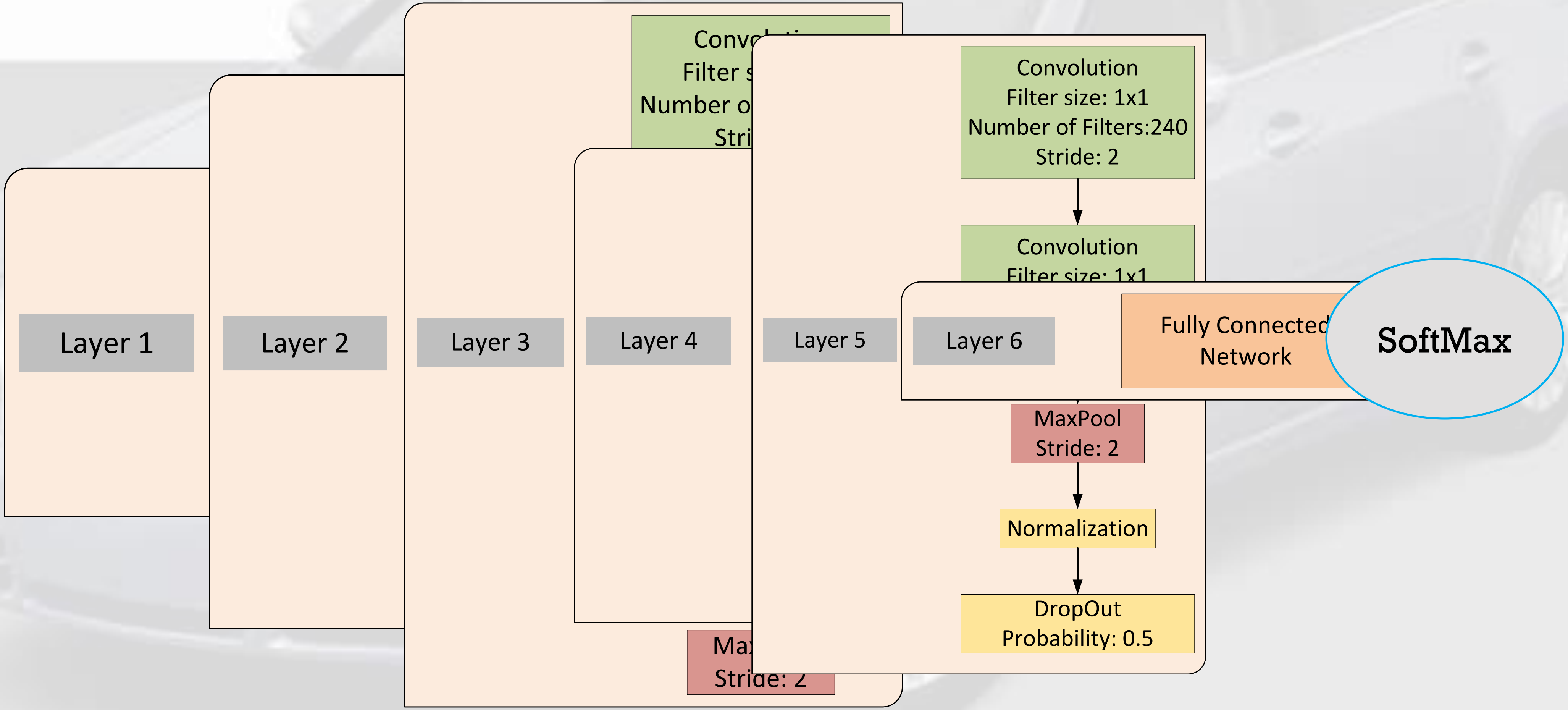
IMPLEMENTATION (GOOGLENET/INCEPTIONNET)



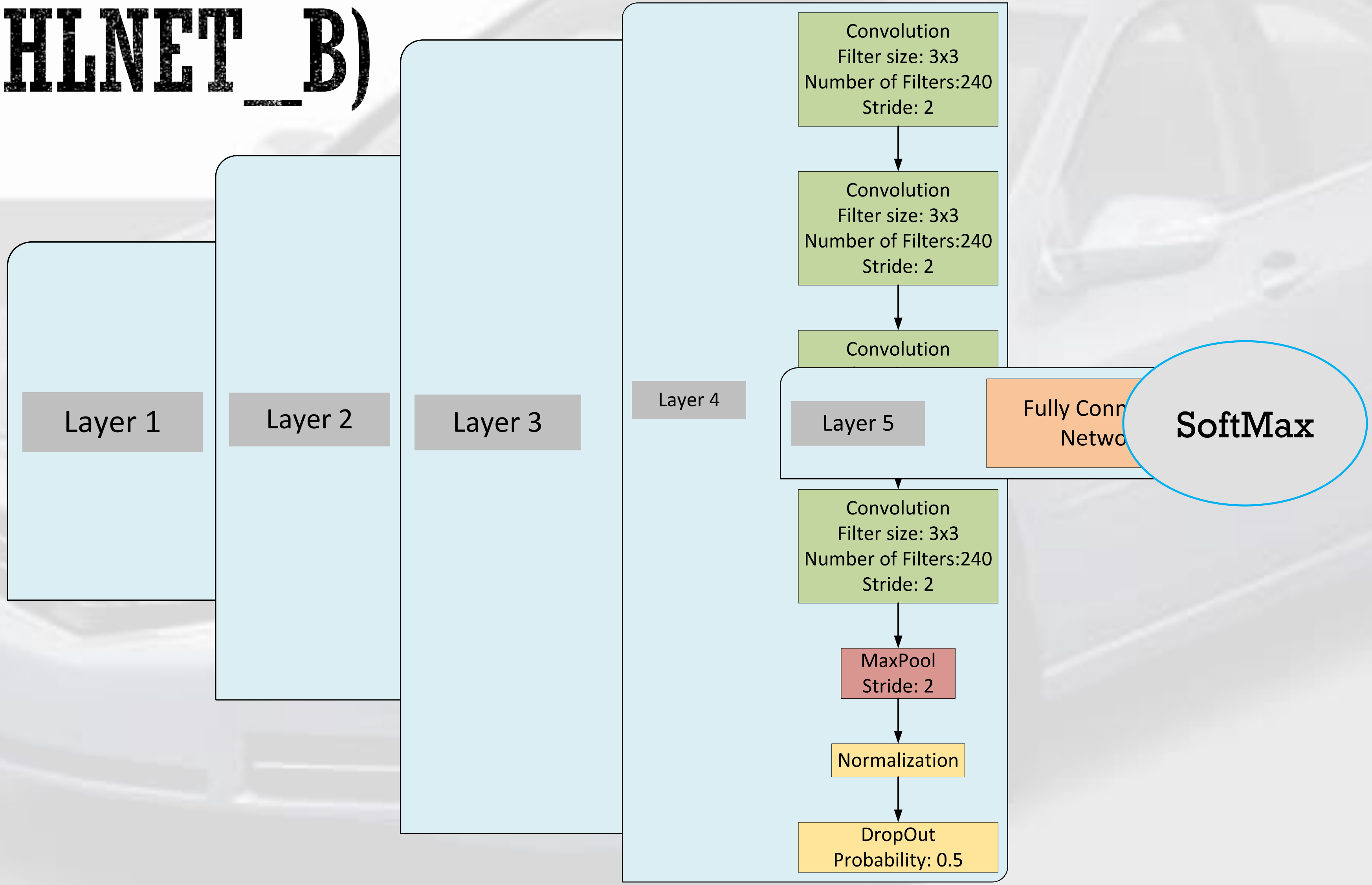
IMPLEMENTATION (NHLNET/OUR MODEL)

- Two different models
  - ✓NhLNet_A
  - ✓6 Layers
    - ➢6 Convolution Layer of size: 3X3
    - ➢2 Convolution Layer of size: 2X2
    - ➢6 Convolution Layer of size: 1X1
    - ➢MaxPool after each Convolution
    - ➢Dropout (50% rate)
    - ➢1 Fully Connected Network
  - ✓NhLNet_B
  - ✓5 layers
    - ➢10 Convolution Layer of size: 3X3
    - ➢MaxPool after each Convolution
    - ➢Dropout (50% rate)
    - ➢1 Fully Connected Network

IMPLEMENTATION (NHLNET_A)



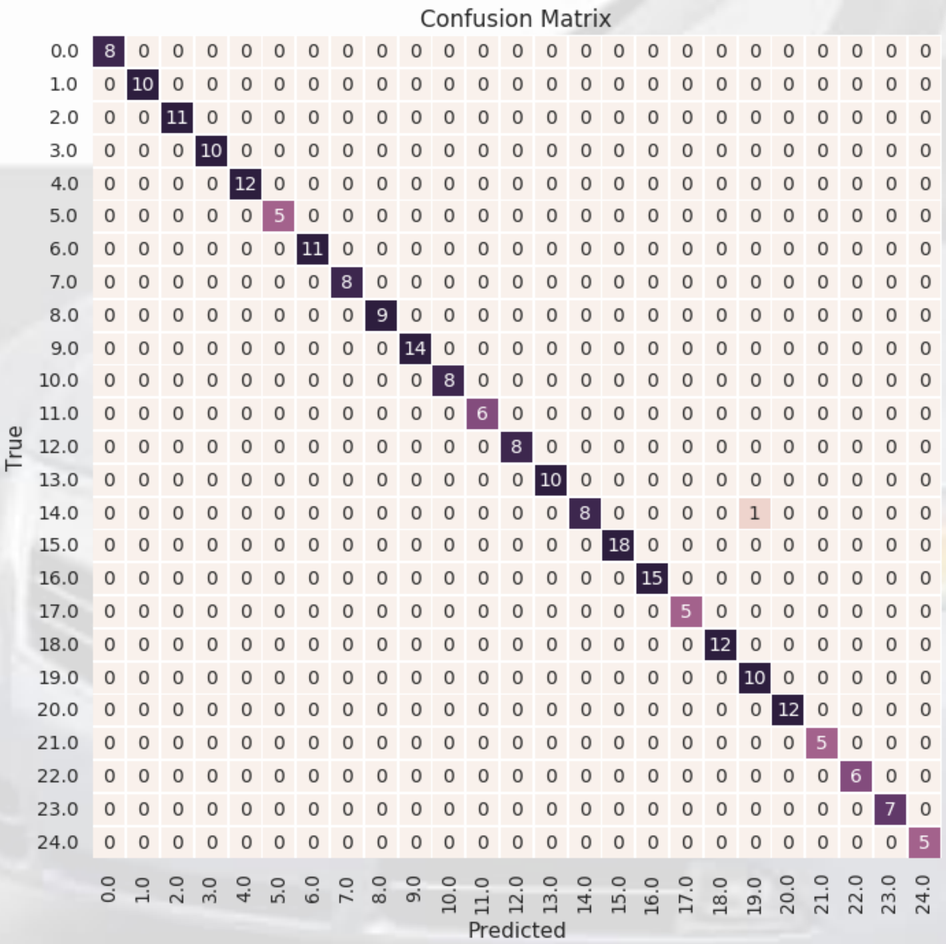
IMPLEMENTATION (NHLNET_B)



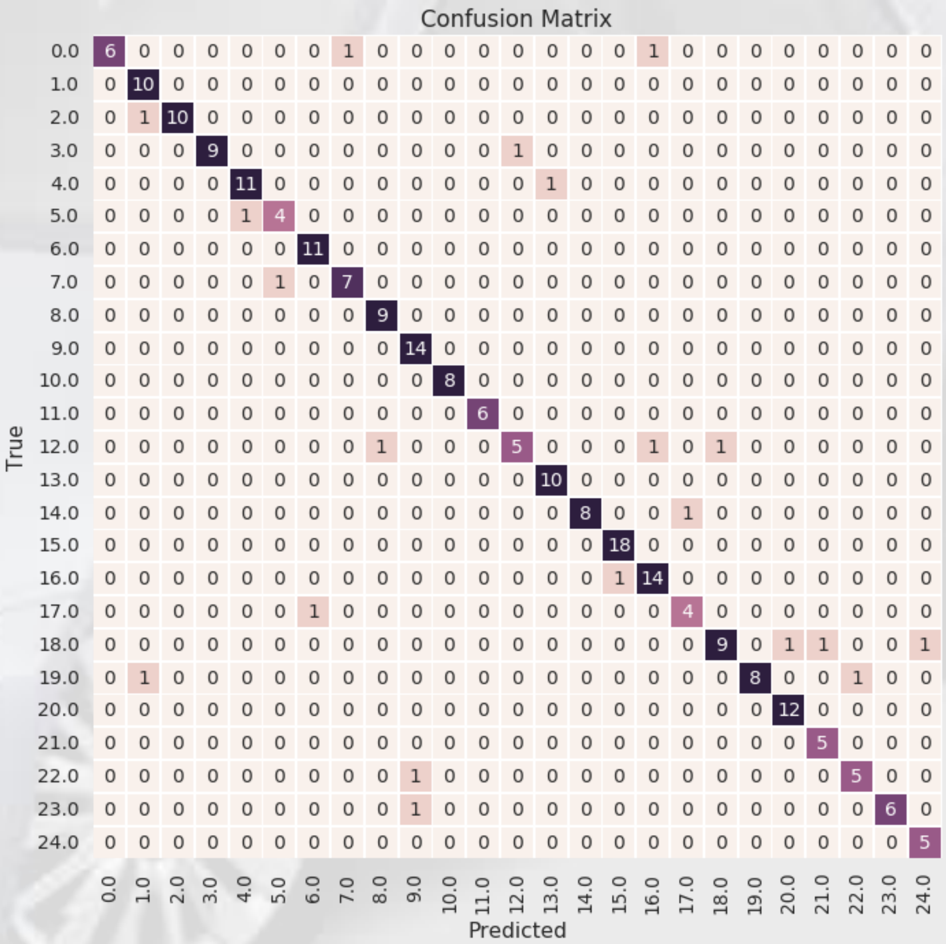
RESULTS AND ANALYSIS

Model	Classes/ Labels	Top 1 Accuracy	Initial Learning Rate
GoogleNet (InceptionNet)	196	0.773	0.001
GoogleNet (InceptionNet)	25	0.981	0.001
NhLNet_A	25	0.183	0.001
NhLNet_B	25	0.951	0.001
NhLNet_B	196	0.753	0.001

RESULTS AND ANALYSIS



GoogleNet confusion matrix for 25 classes



NhLNet_B confusion matrix for 25 classes

RESULTS AND ANALYSIS

Ford Focus sedan 2007

Mercedes Benz C class Sedan2012

GoogleNet Misclassification

RESULTS AND ANALYSIS

Chevorlet corvette ZR1 2012

Chevorlet corvette ZR1 2006

NhLNet Misclassification

CHALLENGES

- Why these misclassifications?
  - ✓Fine grain categorization → Visually Similar Features
  - ✓Size of the dataset
  - ✓Compressed images to make all of them to same size
  - ✓Most of the times, the upgraded model released the following year may have changes in the engine specifications or interior changes by maintaining the exterior same→ identifying the year of the car may become difficult
  - ✓The new model may have some of the features extracted from old model→ identifying the model may become difficult

REFERENCES

[1] Krause, Jonathan, et al. "3d object representations for fine-grained categorization." *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2013.

[2] Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

[3] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." Advances in neural information processing systems. 2012.

[4] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." European Conference on Computer Vision. Springer International Publishing, 2014.

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

[6] Zagoruyko, Sergey, and Nikos Komodakis. "Learning to compare image patches via convolutional neural networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.

[7] Liu, Derrick, and Yushi Wang. "Monza: Image Classification of Vehicle Make and Model Using Convolutional Neural Networks and Transfer Learning."

[8] Jia, Y. Q. C. "An Open Source Convolutional Architecture for Fast Feature Embedding." (2013).

[9] Krause, Jonathan, et al. "Learning Features and Parts for Fine-Grained Recognition." ICPR. Vol. 2. No. 7. 2014.

[10] Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). Vol. 1. IEEE, 2005.

REFERENCES

[11] Zhang, Zhaoxiang, et al. "Three-dimensional deformable-model-based localization and recognition of road vehicles." IEEE transactions on image processing 21.1 (2012): 1-13.

[12] Ramnath, Krishnan, et al. "Car make and model recognition using 3D curve alignment." IEEE Winter Conference on Applications of Computer Vision. IEEE, 2014.

[13] M. Stark, J. Krause, B. Pepik, D. Meger, J. J. Little,B. Schiele, and D. Koller. Fine-grained categorization for 3d scene understanding. In BMVC, 2012.

[14] TFLearn: Deep learning library featuring a higher-level API for TensorFlow.

[15] Sermanet, Pierre, et al. "Overfeat: Integrated recognition, localization and detection using convolutional networks." arXiv preprint arXiv:1312.6229 (2013).

[16] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." European Conference on Computer Vision. Springer International Publishing, 2014.

[17] Srivastava, Nitish, et al. "Dropout: a simple way to prevent neural networks from overfitting." Journal of Machine Learning Research 15.1 (2014): 1929-1958.

[18] Goodfellow, Ian J., et al. "Multi-digit number recognition from street view imagery using deep convolutional neural networks." arXiv preprint arXiv:1312.6082 (2013).

[19] Duchi, John, Elad Hazan, and Yoram Singer. "Adaptive subgradient methods for online learning and stochastic optimization." Journal of Machine Learning Research 12.Jul (2011): 2121-2159.

THANK YOU

Questions?