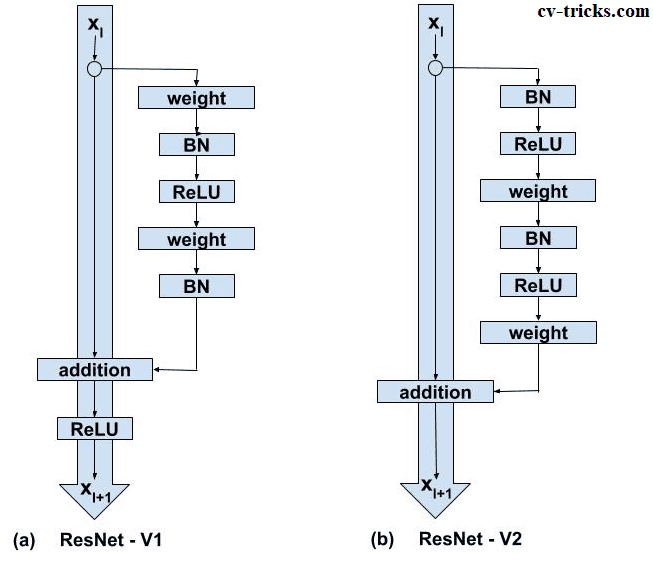
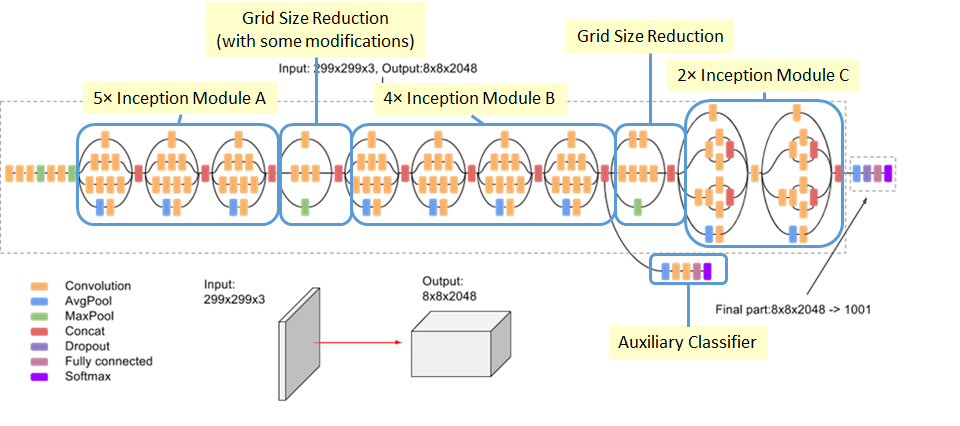
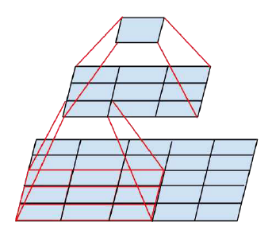
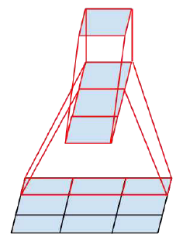
**Data:**

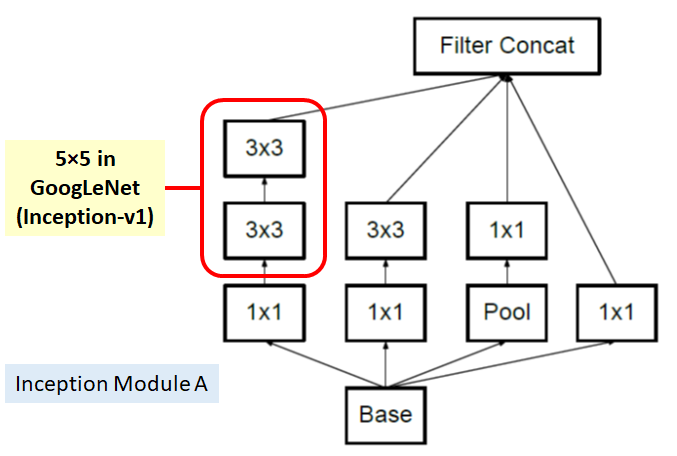
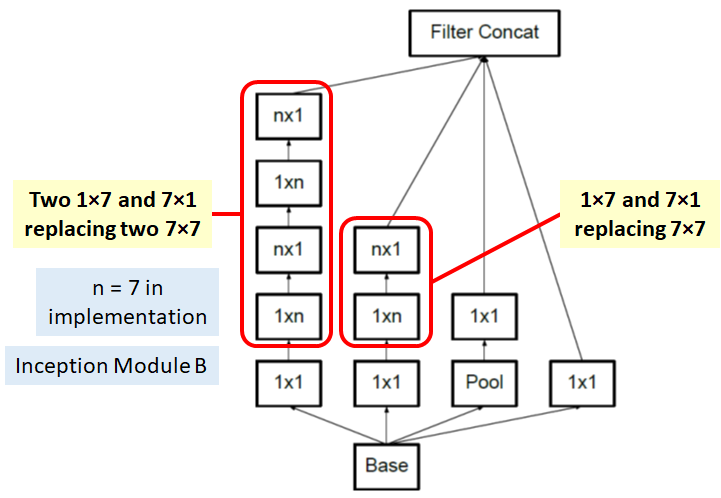
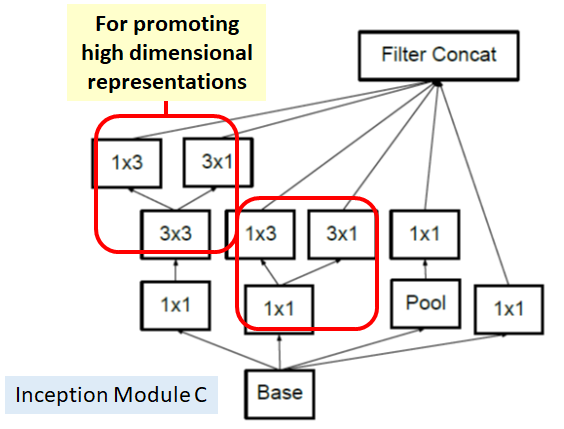
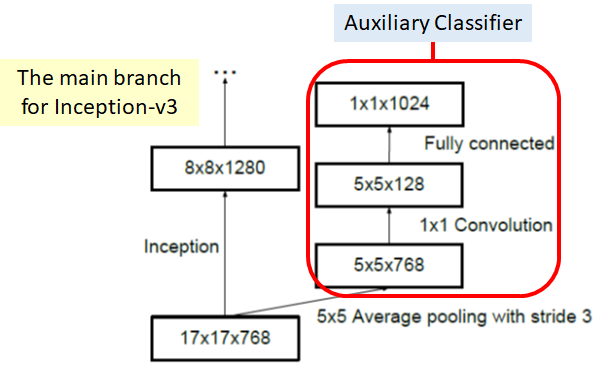
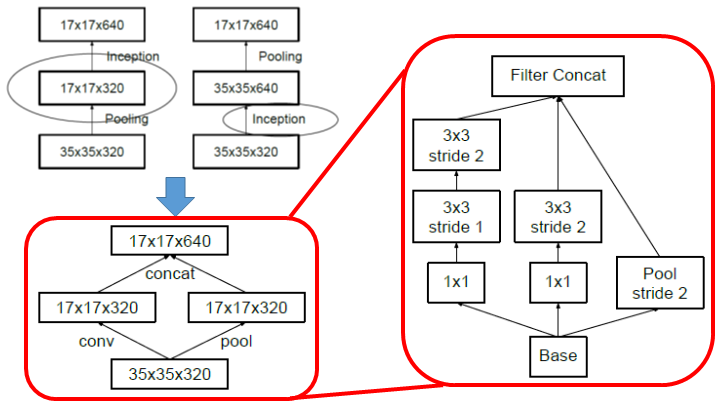
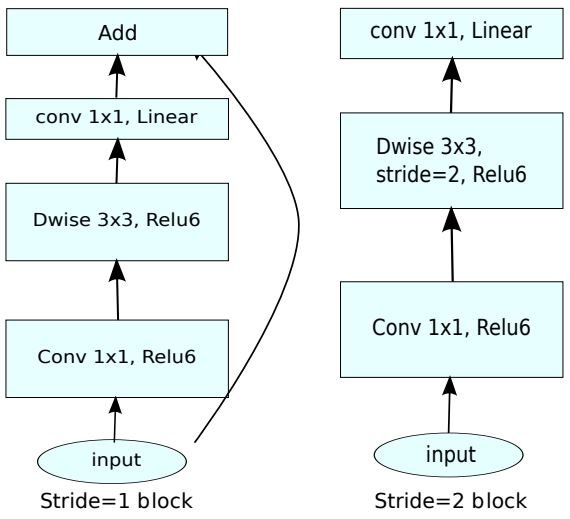
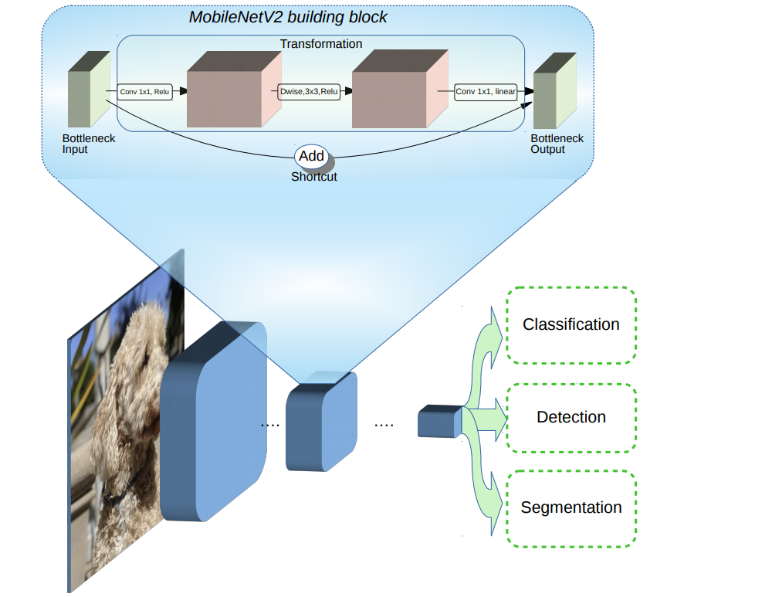
* Obtained from garasi.id
* ~7k Training images from 18 different class of cars
* ~2k Testing Images from 18 different class of cars
* 18 classes: [‘Daihatsu Ayla’, ‘Daihatsu Sigra’, ‘Daihatsu Terios’, ‘Daihatsu Xenia’, ‘Honda BR-V’, ‘Honda Brio’, ‘Honda CR-V’, ‘Honda HR-V’, ‘Mitsubishi Pajero Sport’, ‘Mitsubishi Xpander’, ‘Suzuki Ertiga’, ‘Toyota Agya’, ‘Toyota Avanza’, ‘Toyota Calya’, ‘Toyota Fortuner’, ‘Toyota Kijang Innova’, ‘Toyota Rush’, ‘Toyota Yaris’]

**Models Notes**

* ResNet50
  + Use ResNetV2
  + Utilizes skip connections between the layers to avoid vanishing/exploding gradients problem when doing back propagation.
  + <https://cv-tricks.com/keras/understand-implement-resnets/>
* VGG16
  + Less deep compared to ResNet50, but large and slow to train
  + Suffers from exploding/vanishing gradient problems
  + Utilizes ReLU for activation
  + <https://neurohive.io/en/popular-networks/vgg16/#:~:text=VGG16%20is%20a%20convolutional%20neural%20network%20model%20proposed%20by%20K.&text=Zisserman%20from%20the%20University%20of,images%20belonging%20to%201000%20classes.>
* InceptionV3
  + Utilizes small inception modules within the network. 42 layers deep
  + Factorizing Convolutions: Reduce the number of connections without decreasing network efficiency.

Using 3x1 and 1x3 filters reduce number of parameters by 33% (9->6)

Two 3x3 convolution replaces 5x5 convolution. Parameters reduced by 28% (25 -> 18)

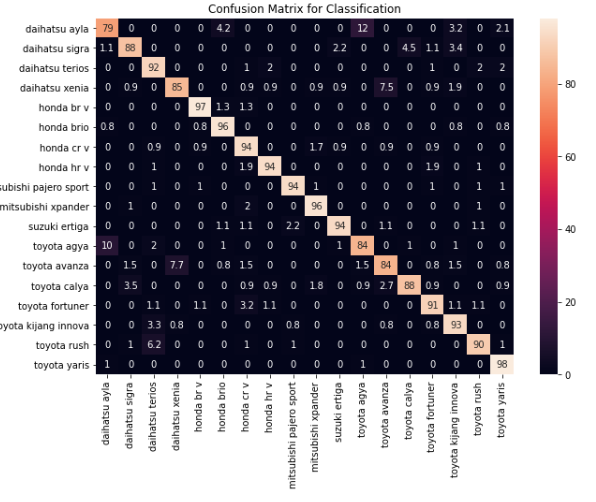
* + Inception Module A (Uses first type of factorizing convolution)
  + Inception Module B (Uses second type of factorizing convolution)
  + Inception Module C (Uses second type of factorizing convolution)
  + 3 kinds of inception modules can help reduce overfitting and reduce number of parameters.
  + **Conventionally, such as AlexNet and VGGNet, the feature map downsizing is done by max pooling. But the drawback is either too greedy by max pooling followed by conv layer, or too expensive by conv layer followed by max pooling.**
  + Auxilary classifier used to reduce overfitting (regularization) hanging on the side
  + With the efficient grid size reduction, 320 feature maps are done by conv with stride 2. 320 feature maps are obtained by max pooling. And these 2 sets of feature maps are concatenated as 640 feature maps and go to the next level of inception module.
  + <https://medium.com/@sh.tsang/review-inception-v3-1st-runner-up-image-classification-in-ilsvrc-2015-17915421f77c>
* MobilenetV2
  + Small architecture to output fast classification
  + Simple model that uses ReLU activation
  + <https://towardsdatascience.com/review-mobilenetv2-light-weight-model-image-classification-8febb490e61c>

**Model Architecture**

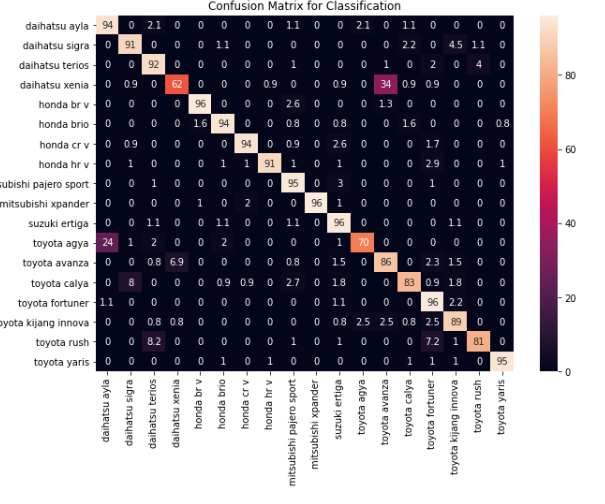
* Base Model + 3 FC layers + Last softmax layers
* Batch Normalization in FC layers
* Dropout of 20%
* Resnet50
  + ResNet50 Base + 512 Dense + BN + D + 256 Dense + BN + D + 64 Dense + BN + Softmax
  + Relu activation function used
* VGG16
* InceptionV3
  + InceptionV3 Base + 512 Dense + BN + D + 256 Dense + BN + D + 64 Dense + BN + Softmax
* MobinetV2

**Result**

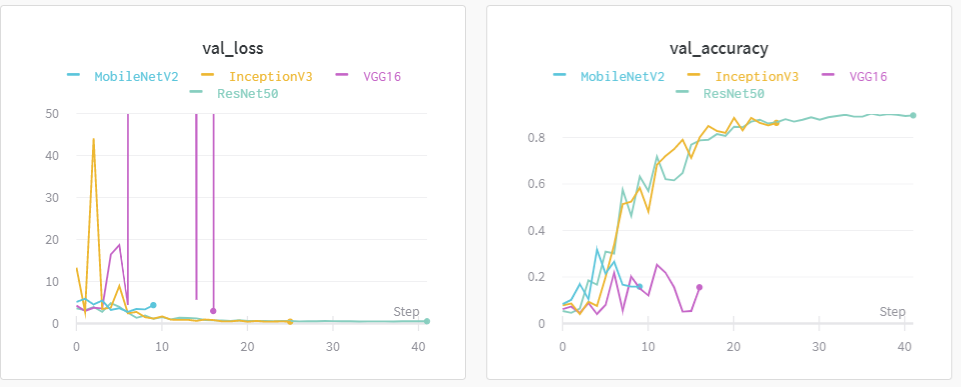
|  |  |  |
| --- | --- | --- |
| **Model** | **Result (Test Accuracy)** | **Notes** |
| ResNet50 Base | 90.85% | Best Model |
| VGG16 Base | 22.23% | Vanishing Gradients |
| InceptionV3 Base | 88.75% | Decent |
| MobileNetV2 Base | 32.40% | Vanishing Gradients |

**Confusion Matrix**

Confusion Matrix for ResNet50

 Confusion Matrix for InceptionV3

Both models have high False Positives for Toyota Avanza vs Daihatsu Xenia and Toyota Agya vs Daihatsu Ayla. They are in fact similar cars with each other where Toyota and Daihatsu collaborated to build the two models.

**Loss and Training Accuracy**



**Conclusion**

Best Model utilizes Resnet50 model as base