A picture containing text

Description automatically generated

**CZ4042: Neural Networks and Deep Learning**

**Group Project Report**

Prem Adithya Suresh (U1820740B)

Shearman Chua Wei Jie (U1820058D)

Sam Jian Shen (U1821296L)

TUTORIAL GROUP: CS4

[**Introduction**](#_heading=h.1r4r60e3bzvd) **3**

[**Review of Existing Techniques**](#_heading=h.fpocsoew5gmw) **3**

[H. Liu et al. 2016 “Deep relative distance learning: tell the difference between similar vehicles”](#_heading=h.bekflssnj7oh) 3

[**Dataset Analysis and Preparation**](#_heading=h.cq3lwrroyhwm) **3**

[Exploratory Analysis](#_heading=h.b7i48ceuchii) 3

[**Dataset Preparation**](#_heading=h.hp2lkpy22xsd) **4**

[**Methods Used**](#_heading=h.2xqhuw51ko58) **5**

[**Experiments and Results**](#_heading=h.yczlaxw40i3a) **5**

[**Discussions**](#_heading=h.wu64vg6ug10y) **5**

# Introduction

Automated car model analysis, especially fine-grained car categorization and verification, can be used for multitudinous purposes in intelligent transportation systems, including vehicle regulation, description, and indexing. For instance, fine-grained car categorization can be exploited to inexpensively automate and expedite paying tolls from the lanes, based on different rates for different types of vehicles.

For this project, we will be working on the Comprehensive Cars (CompCars) Dataset from <http://mmlab.ie.cuhk.edu.hk/datasets/comp_cars/index.html>. The Comprehensive Cars (CompCars) dataset contains data from two scenarios, including images from web-nature and surveillance-nature. The web-nature data contains 163 car makes with 1,716 car models. There are a total of 136,726 images capturing the entire cars and 27,618 images capturing the car parts. The full car images are labelled with bounding boxes and viewpoints. Each car model is labelled with five attributes, including maximum speed, displacement, number of doors, number of seats, and type of car. The surveillance-nature data contains 50,000 car images captured in the front view.

The goal of this project is to design a neural network to classify a given image into one of the 163 car makes. After which, consider designing a multi-task learning framework that classifies not only car makes, but also car models and the corresponding attributes. Some of the other sub-tasks include varying the network depth, tuning the network parameters to improve validation accuracy and validation loss. We will then observe the effects of utilizing a more advanced loss function on the neural network.

# Review of Existing Techniques

## H. Liu et al. 2016 “Deep relative distance learning: tell the difference between similar vehicles”

In the paper “H. Liu, Y. Tian, Y. Wang, L. Pang, T. Huang, “Deep relative distance learning: tell the difference between similar vehicles,” in Computer Vision and Pattern Recognition (CVPR), 2016”, H. Liu and his colleagues proposed a Deep Relative Distance Learning (DRDL) method which exploits a two-branch deep convolutional network to project raw vehicle images into an Euclidean space where distance can be directly used to measure the similarity of arbitrary two vehicles.

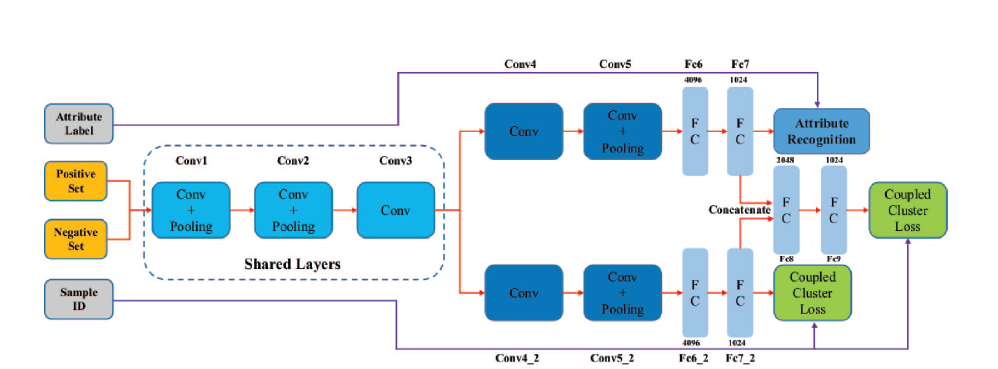
DRDL is an end-to-end framework specifically designed for vehicle re-identification. It aims to learn a deep convolutional network that can project raw vehicle images into an Euclidean space where the L2 distance can thus be used directly to measure the similarity of arbitrary two vehicles. The basic idea of DRDL is to minimize the distances of the same vehicle images and maximize those of other vehicles.

Prior to the paper, many research made use of the triplet loss deep convolutional network architecture for a multitude of recognition tasks. The architecture connects a deep convolutional network for feature extraction and a special triplet loss to achieve good performance in both person re-identification and face recognition problems. It is assumed that samples of the same identity should be closer from each other compared to samples of different identities. By optimizing the specifically designed triplet loss function, the network will gradually learn a harmonic embedding of each input image in Euclidean space that tends to maximize the relative distance between the matched pair and the mismatched pair. However, generating all possible triplets would result in numerous triplets and most of them are too easy to distinguish that would not make any contribution to the loss convergence in the training phase.

The paper proposed an end-to-end framework DRDL that is suited for both vehicle retrieval and vehicle re-identification tasks. Two different vehicles(with different license plates) could be almost the same regarding their appearance if they belong to the same model. The research aimed to capture both the inter-model difference and intra-model difference between different vehicles. In order to make the training phase more stable and accelerate the convergence speed, the paper proposed a new loss function to replace the triplet loss, namely, coupled clusters loss(CCL). Similarly, a convolutional network is used to extract features for each image here. But the triplet input is replaced by two different image sets: one positive set and one negative set.

The paper made use of a base VGG CNN M 1024 network structure, containing 5 convolutional layers and 2 fully-connected layers, for their experiments. The dimension of the network’s last fully-connected layer “fc7” is 1024 neurons. The last fully connected layer “fc8” is a mixed feature of both the vehicle’s model information and the feature representation learned from single triplet loss or coupled clusters loss. The dimension of “fc8” is set to 1024 in accordance with the output dimension of standard VGG CNN M 1024 network to eliminate the influence of feature dimensional difference when performing evaluation experiments. “fc7 2” in the mixed difference network is just the same as the output feature of a standard VGG CNN M 1024 network while “fc8” is an enhanced one suitable for both inter-model difference and intra-model difference metric. The networks are all fine

tuned on VGG CNN M 1024 which is pre-trained with the ImageNet dataset in ILSVRC-2012. They used a momentum of μ = 0.9 and weight decay λ = 2 × 10−4.



**Fig.1 Mixed difference network structure based on VGG CNN M 1024 used in paper**

### Results

The network was trained on the “CompCars” dataset, using the Part-I subset(derived from L. Yang, P. Luo, C. C. Loy, X. Tang, “A large-scale car dataset for fine-grained categorization and verification,” in IEEE Computer Vision and Pattern Recognition (CVPR), 2015) which contains 431 car models with a total of 30955 images capturing the entire car and tested on vehicle verification task on the Part-III subset which contains 22236 images of 1145 models. The test data is organized into three sets, each of which has different difficulty, i. e. easy, medium, and hard. Each set contains 20,000 pairs of images, including 10,000 positive pairs and 10,000 negative pairs. Each image pair in the "easy set" is selected from the same viewpoint, while each pair in the "medium set" is selected from a pair of random viewpoints. Each negative pair in the "hard set" is chosen from the same car make.

Three other methods are introduced to perform the comparison experiments. The experimental results of the first two methods, “Deep Feature+SVM or Joint Bayesian”, are referred from “L. Yang, P. Luo, C. C. Loy, X. Tang, “A large-scale car dataset for fine-grained categorization and verification,” in IEEE Computer Vision and Pattern Recognition (CVPR), 2015”. They first utilize a deep convolutional network to train a vehicle model classification model on Part-I data of “CompCars”. Then, Joint Bayesian or SVM is applied to train a verification model on Part-II data with the classification network in step 1 as a feature extractor. The third algorithm, “VGG CNN M 1024” network with triplet loss function is trained with Part-I data of “CompCars” and the corresponding vehicle model labels. The results are given below:

|  |  |  |  |
| --- | --- | --- | --- |
| Accuracy | Easy | Medium | Hard |
| GoogleNet+SVM | 0.700 | 0.690 | 0.659 |
| GoogleNet+Joint Bayesian | 0.833 | 0.824 | 0.761 |
| Mixed Diff+CCL | 0.833 | 0.788 | 0.703 |

The “VGG CNN M 1024+Triplet Loss” got no results because its loss function failed to converge during the training phase.

# Resources Used for Project

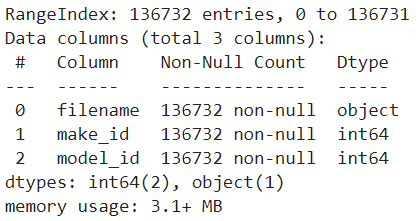
For this Project, our group will be building and training the neural network on the Google Colaboratory platform. The neural network will be coded in Python programming language. The reason behind this is that Python has many libraries that can be used to easily build neural networks. Also, by using the Google Colaboratory platform, we are able to utilize Tensor Processing Units (TPUs) developed by Google to allow our neural network to be trained faster.

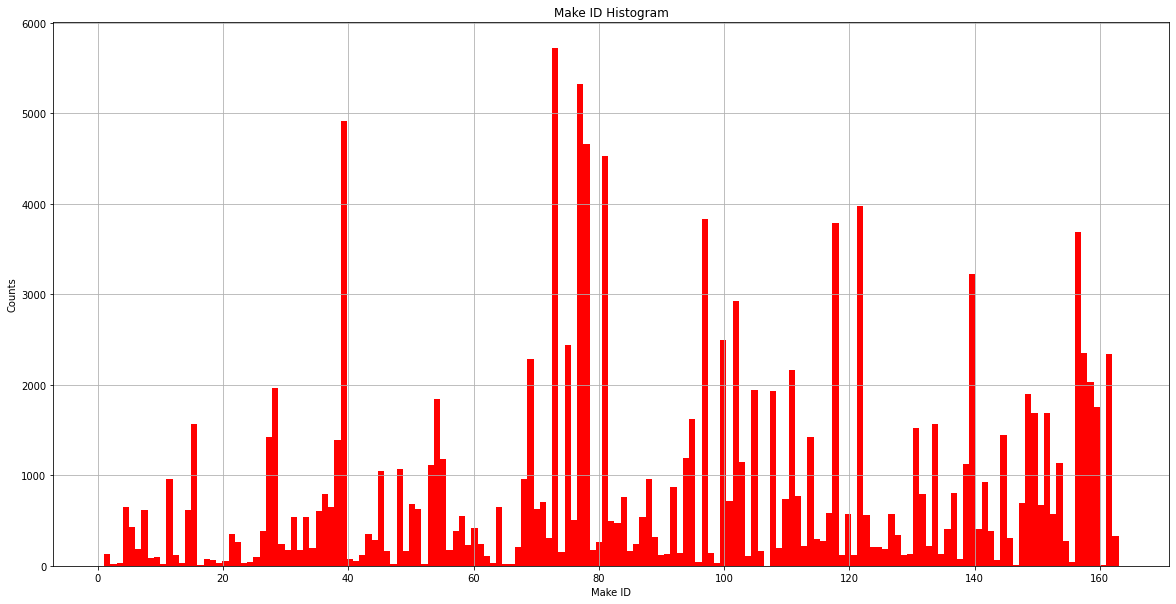
## 3.1 Python Libraries

# Dataset Analysis and Preparation

## 

## Overview





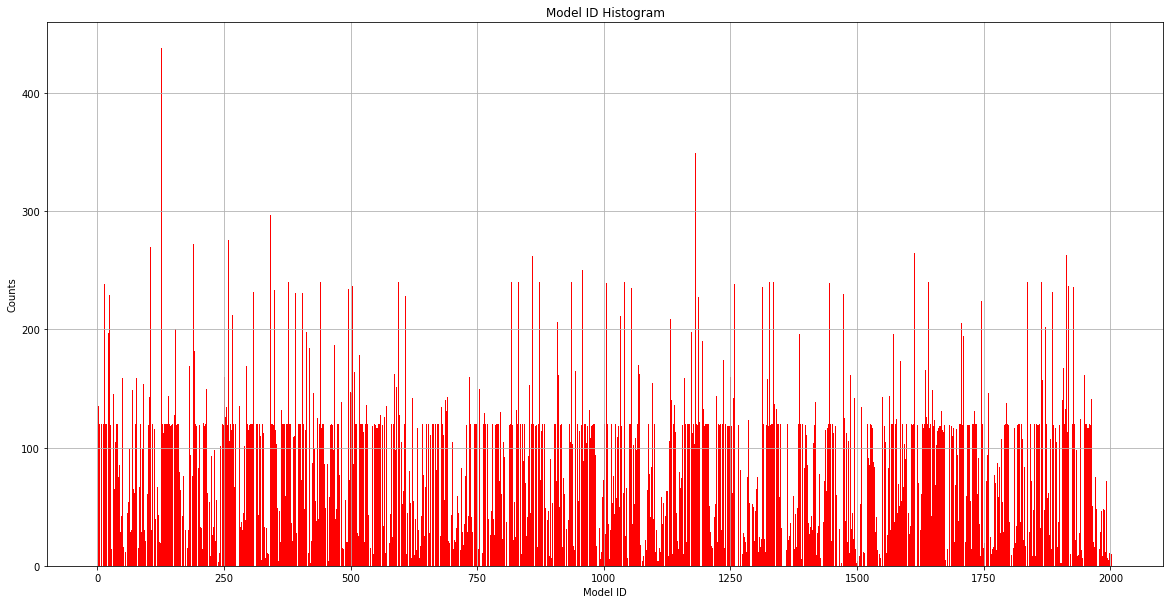
Maximum Value of Make ID: 163

Minimum Value of Make ID: 1

Total Number Of Make Unique IDs: 163

Observation:

* We can see that all ID are ranging from 1 to 163 as there was a total of 163 unique IDs
* The counts were largely deviated from one another. There are counts as high as 5k over and as low as 2 digits.



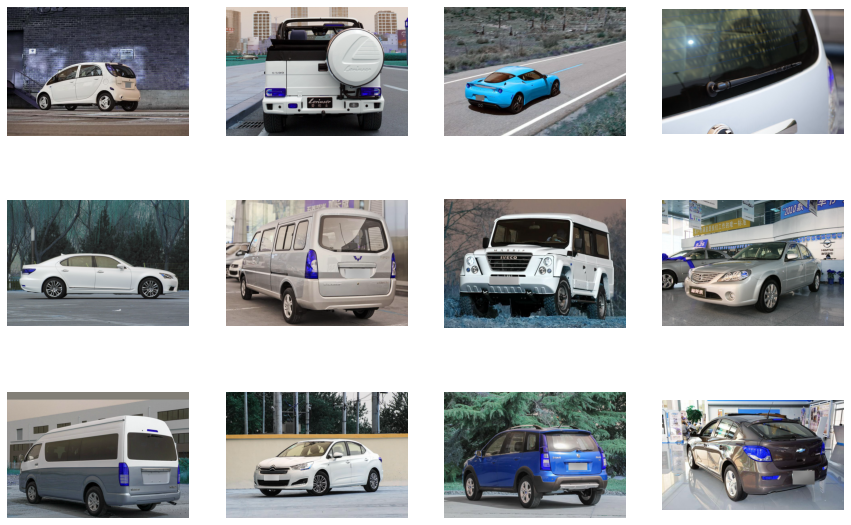
Maximum Value of Model ID: 2004

Minimum Value of Model ID: 1

Total Number Of Model Unique IDs: 1716

Observation:

* Unlike Make ID, Model ID is not uniformly allocated, there exist values that do not belong to Model ID.
* We can see the mean values

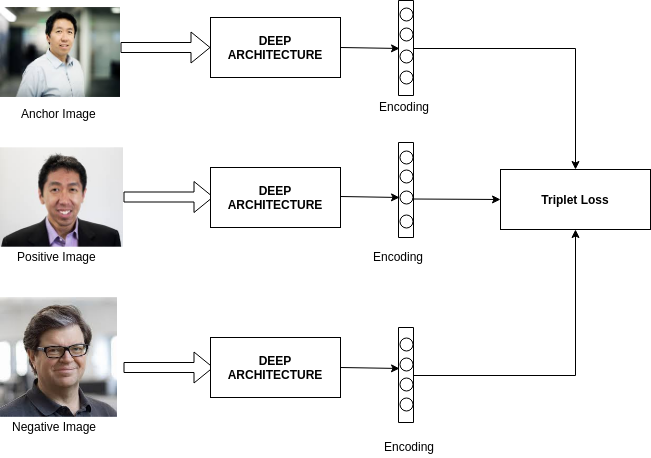


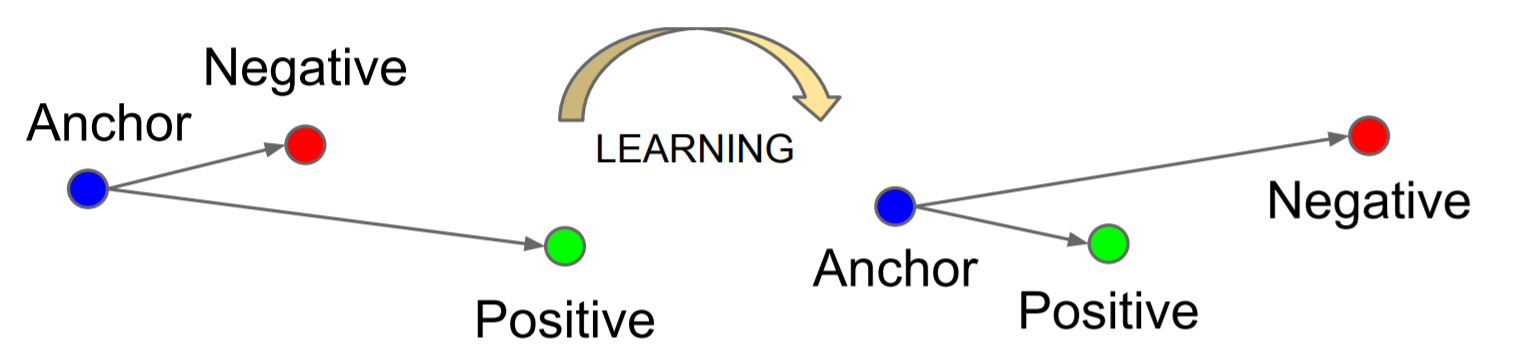
## Dataset Preparation

# 

# Methods Used

* Multi task learning framework (2 heads)
* Hyperparameter Adjustment
  + Number of Hidden Layers
    - EfficientNetB5 instead of EfficientNetB7 due to shorter training time
  + Added dropout layers to reduce overfitting
  + Image augmentation to increase accuracy by increasing model generalisation
* Loss Function
  + Triplet Loss
    - Concept





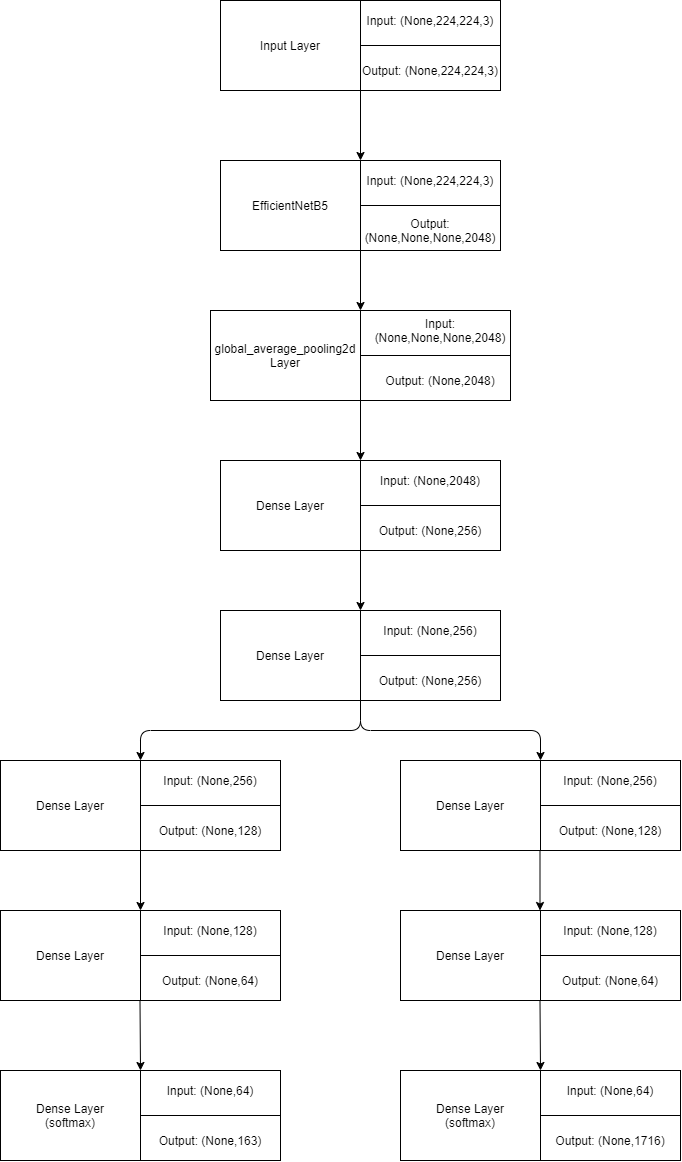
* + - Purpose
      * Minimizes the distance between an anchor and a positive, both of which have the same identity, and maximizes the distance between the anchor and negative of a different identity.

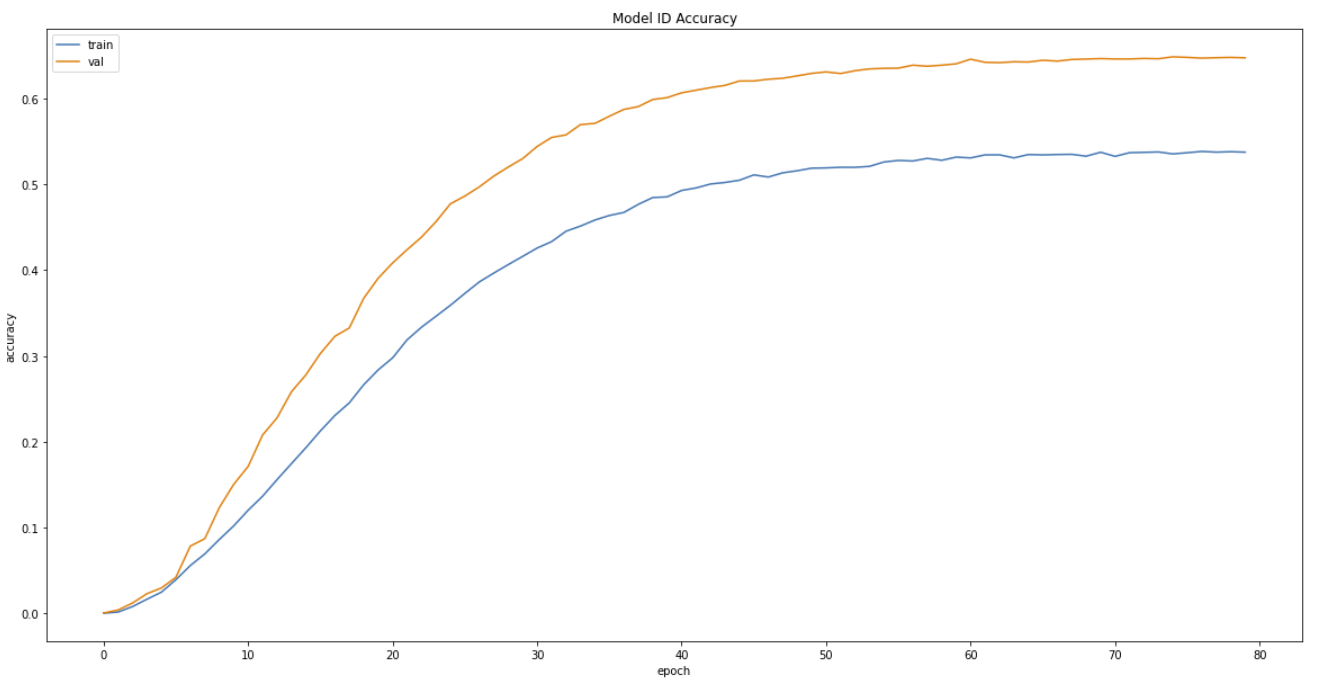
# Experiments and Results

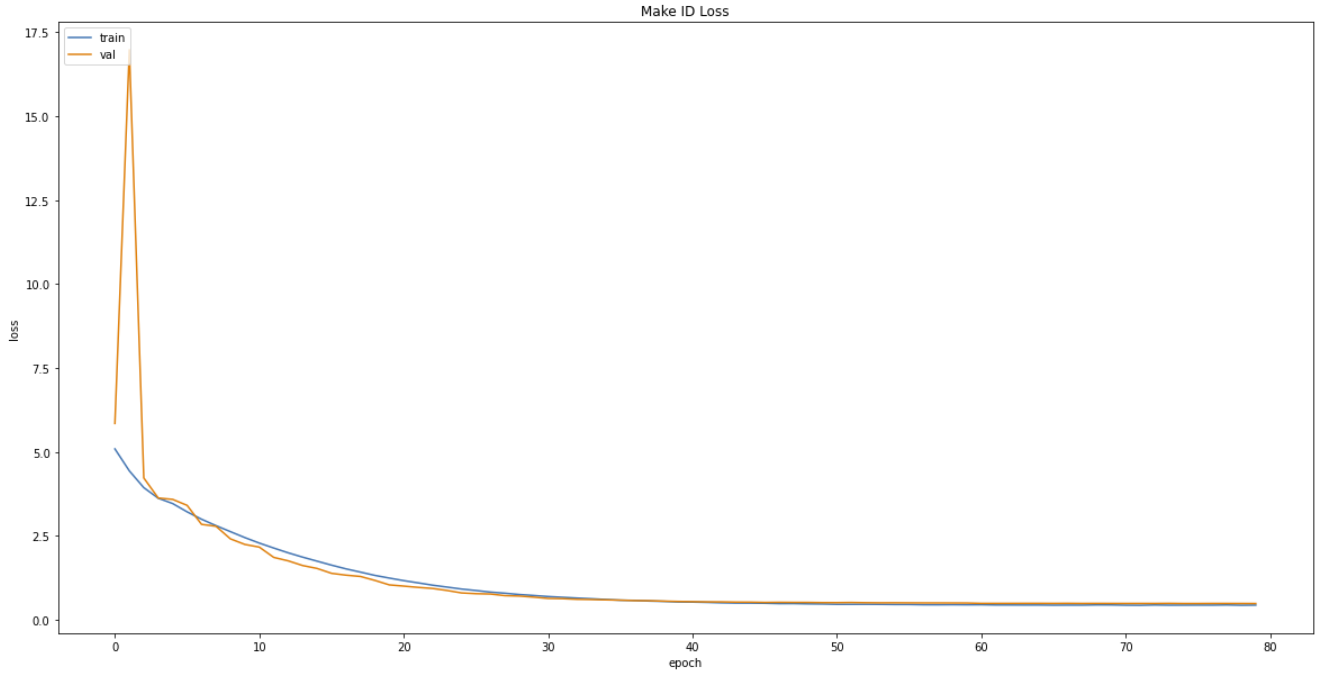
## 7.1 Baseline Deep Neural Network

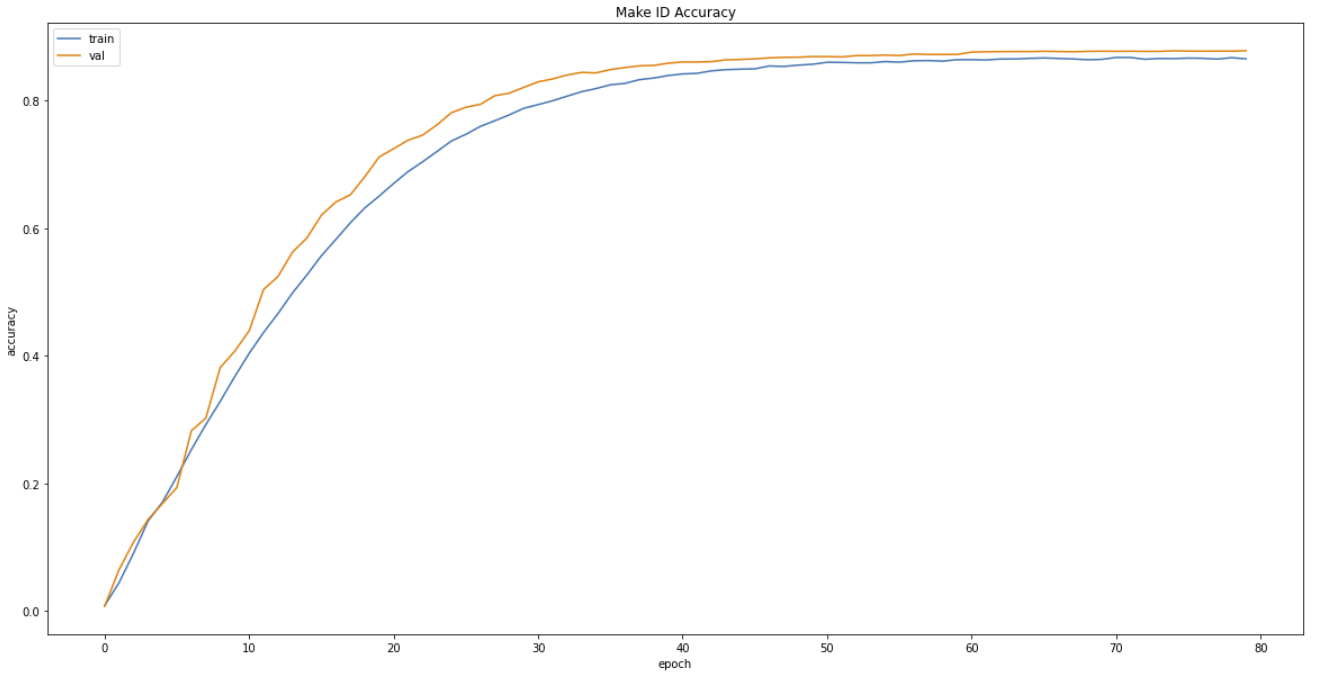
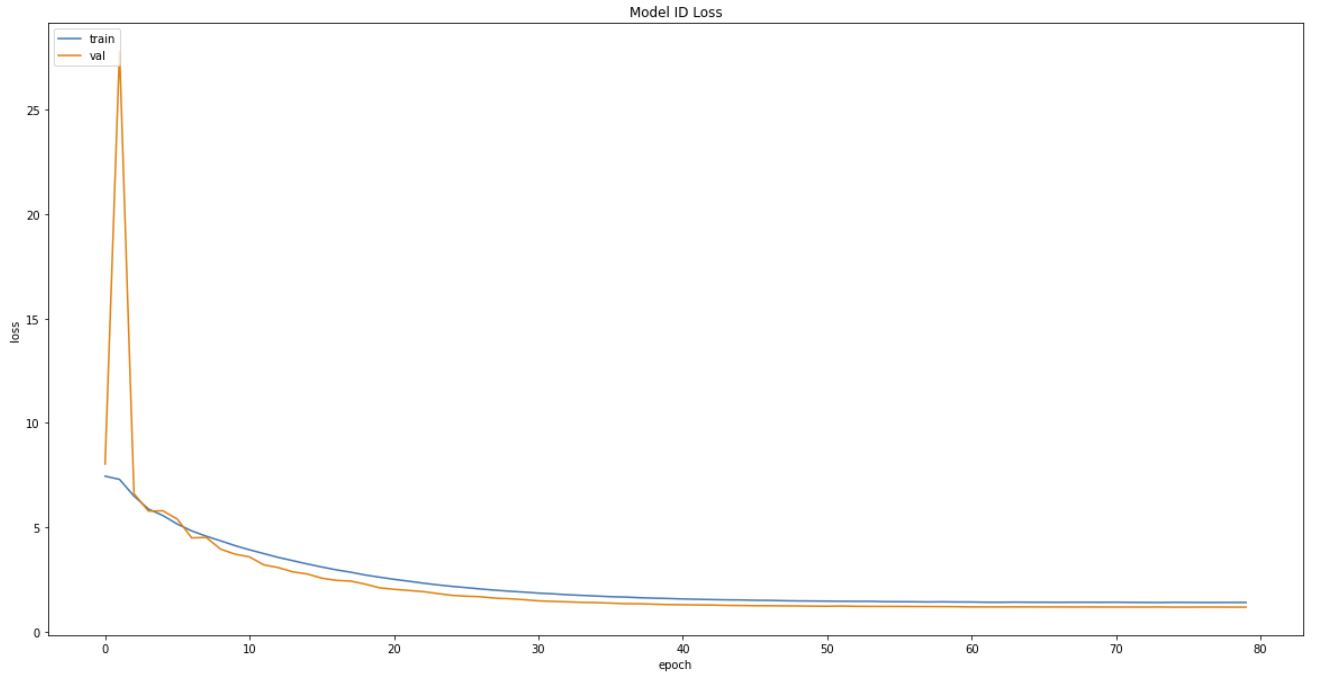
We first implemented a baseline Deep Neural Network to classify a given image into one of the 163 car makes as well as the 1716 car models. The Deep Neural Network will have a Multi task learning framework with 2 outputs, one for classifying the image into one of the car make and the other for classifying the image into one of the car models.

At the core of our network will be the EfficientNetB5 architecture which is a Convolutional Neural Network released in June 2019 by Google AI and is the new state-of-the-art on ImageNet. It introduces a systematic way to scale CNN (Convolutional Neural Networks) in a nearly optimal way. We then add a few more dense layers after the EfficientNetB5 architecture, followed by the 2 output layers to form our Neural Network to classify images into the car makes as well as the car models. Below shows the diagram of the Neural Network architecture:









# Summary

# References