# Fine-grained recognition for Indian cars

Prince Patel
IIIT Delhi
Delhi
prince15046@iiitd.ac.in

## Abstract

*In this project, we will solve the problem of fine grained classification of cars, particularly Indian cars. The project aims at creating an application which can detect the cars present in the camera's view and then classify them according to its make, company and model. Three main challenges in this task are:first, is to gather significant amount of data/meta-data of Indian cars, in form of images and they will be annotated according to our work. Second, is to create a model which will do fine grained classification, giving output as make,model and company of the car. Third, is to work with images which have haziness, blurring and low lighting. We plan to use Open CV to detect cars in an image frame and use Convolutional Neural Network(CNN) in various configurations to learn features of the car parts and then classify them accordingly. The metrics to evaluate the model will be classification and verification accuracies, specifically Top-1 and Top-5 accuracies.*

## 1. Introduction

Fine grained recognition [2][6] refers to the task of finding more refined information about the objects. For example if we find that that there is a bird in a picture/video, we emphasize on detecting the type of bird. To add this level of segregation, we focus on the specific features and parts of objects that are detected. This level of recognition in machines would be a major breakthrough in various domains. A lot of work has been done in this area worldwide[3-6]. Our work proposes number of novel models to recognize models for Indian cars. There are various applications of fine grained recognition in multiple domains. For example surveillance applications for Indian agencies for car models recognition, where search is for particular model and make of car. Car agencies who want to survey the number of cars of a particular model. We will also try to push models to real time application which could result in crowd-sourcing applications.

## 2. Prior Work

We have followed the work of Yang et.al [5] and Krause et.al.[1] for our research.

### 2.1. CompCars

Yang et.al.[5] have created a large scale and detailed dataset which can capture various internal and external parts of the car. The authors have mentioned that there is a lack of high quality dataset which captures rich and detailed features of a car, and in turn restricts the exploration of computer vision challenges in this domain. In this paper, they have shown that many car related problems and applications haven't been well researched and explored.

They done the task of creating a dataset of cars of cross modal nature, which would give a diverse view of cars present in the market. The cross modality of the dataset contains images from 2 scenarios: web-nature and surveillance. Along with the various views of a car, they capture rich attributes like type of car, seat capacity and number of doors.The CompCar dataset contains 214,345 images of 1,687 models. Apart from that, the authors have studied the three applications which are fine grained classification, attribute prediction and car verification.

For fine grained experiment, there are two setups: first, is entire car images and second is using images of car parts. For both the experiments, the authors use the Overfeat model, which is fine-tuned using logistic-loss and car model labels as training data. To classify the entire car images, the CNN model is fine-tuned using different viewpoints namely, front, rear, side, front-side, rear-side and all-view. For car parts, the CNN model was trained using images of various car parts. The results showed that tail light gave best responses than any other car part. For attribute prediction, the authors fine-tune the CNN model with sum-of-square loss to incorporate the continuous attributes like max speed, displacement, but use logistic loss to predict attributes like number of doors, seat type etc which are discrete in nature. To perform car verification, they use the classification model used for fine-grained task as feature extractor of car images, and then apply Joint Bayesian to train a verification model. The dimensionality of the features extracted was reduced using PCA. These features are then used to train the Joint Bayesian Model.

The fine-grained classification of car models were evaluated using Top-1 and Top-3 predictions for each of the viewpoints in case of entire car image. The classification using car parts, also used the same rubric for testing the performance, using various internal and external parts. To evaluate the performance of attribute prediction using different viewpoints, mean guess i.e errors calculated using mean of training set and classification accuracy was used. And at last the performance of car verification using (CNN feature+Joint Bayesian), was done by plotting ROC curves. The paper makes an assumption of putting car images of different years into a single category. So the dataset considers all cars of same model but different years as one, which is contrary to what we are trying to achieve.

### 2.2. Learning Features and parts for fine grained recognition

Krause et.al. in there paper[1], solves the problem of fine grained recognition of cars.

The main idea is to learn both the features and parts to create a unified object depiction. They use CNN to learn appearance descriptors and collect part detectors with unsupervised learning. By keeping the part discovery unsupervised, it would make their algorithm scalable to range of fine-grained domains. During recognition time, the authors use their novel method called Ensemble of Localized Learned features(ELLF), to detect parts and represent their appearances using learned features.

The experiment is performed on cuda-covenet implementation of CNN. To train there final classifier(a linear SVM) using ELLF features extracted on the original images as data. The results in this paper outperformed the work done in last papers on the subject. The accuracy of the plain CNN comes out to be 70.5% which is better than experiments using other features such as SIFT. However, this model is not perfect as the speed is an issue. This can be rectified by retraining the CNN on ImageNet.

## 3. Problem Statement

The project aims to detect Indian car models and the defined attributes by studying their various details using a locally created dataset of images of such cars in diverse realistic scenarios. The strong assumption made here is that the cars having evolved on yearly basis are to be kept in the same class as those differences are hard to spot given the practical conditions. This problem can be thought as a classification problem, where the models and make of cars are various classes.

## 4. Benchmark Model

Since our test images are not evaluated on any of the standard models, therefore the benchmarking is done with one of initial model. We will choose the HOG based classification as our benchmark model. The reason to choose it as our benchmark is that every computer vision task can be solved by some image processing task's. HOG is a bag-of-words model which is widely used in initial approaches of classification.

## 5. Approach

We will create Indian car dataset on pre-defined classes(models). The number of classes will be according to the data scraped from various portals of used cars. We picked used cars portals for data-set generation as they are true to real world images. The dataset will be refined using image processing, application of noise, translation in bounding box, mirror images etc. Then we plan to apply variations of the deep convolutional models for solving the task like VGG16 model, Inception model[4]. Variation in Deep CNN will comprise of changes in dropouts, error rate, momentum, drop connect and Labelled Noise in images.

### 5.1. HOG-based classification

The basic idea behind the HOG descriptor is that local image features can well be described by the distribution pattern of intensity gradients or edge directions. The image is divided into small adjacent patches called cells, and for the pixels within each cell, a histogram of gradient directions is computed. The descriptor is the concatenation of these histograms.
The HOG descriptor has a few key advantages over other descriptors. Since it operates on local cells, it is invariant to geometric and photometric transformations, except for object orientation. Such changes would only appear in larger spatial regions. Moreover, as Dalal and Triggs discovered, coarse spatial sampling, fine orientation sampling, and strong local photometric normalization permits the individual body movement of pedestrians to be ignored so long as they maintain a roughly upright position. The HOG descriptor is thus particularly suited for human detection in images

### 5.2. Inception model

The pre-trained deep learning model that will be used is Inception-v3. It has been developed by Google and has been trained for the ImageNet Competition using the data from 2012. We chose this model because of its high classification performance and because it is easily available in TensorFlow.

### 5.3. VGG16

The VGG network architecture was introduced by Simonyan and Zisserman. This network is characterized by its simplicity, using only 33 convolutional layers stacked on top of each other in increasing depth. Reducing volume size is handled by max pooling. Two fully-connected layers, each with 4,096 nodes are then followed by a softmax classifier .

### 5.4. Fine Tuning

Fine tuning is task of taking a retrained model which is for a particular task, and make it perform a second similar task. The method to perform fine tuning task, is to take a model, which is originally trained for somewhat similar task(Image recognition in our case) and then replacing the last layer to change the number of classes. VGG16 and InceptionV3 were trained for 1.2m images. Now initial layers contain the features of images, if we change the last layer and retrained the model on some other labelled images. The model will then work according to our task.

## 6. Dataset and Inputs

The data set for the problem comprises around 5518 images of Indian cars divided into 5 classes each having approx. 1000 images to themselves. The images have been scraped from Google images. The dataset being used is smaller than that in [3] but will serve the purpose.

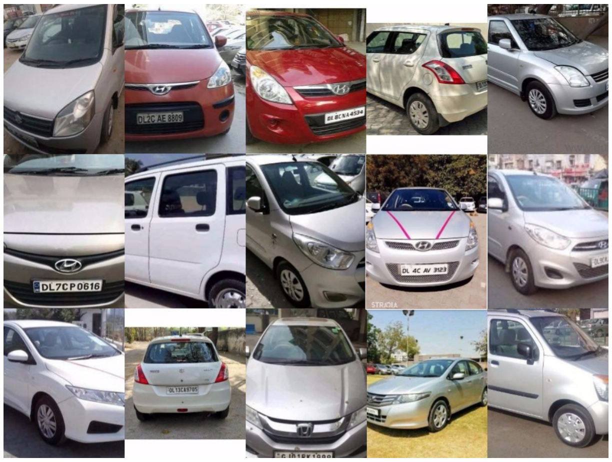


Figure 1: Data set: Total number of images are 5518, 1000 per class. Each view angle of car is covered in images.

## 7. Evaluation Metrics

The evaluation metrics that are used below are selected as they provide a clear boundary to distinguish between models of cars, which is crutial to the fined grained classification task.

- **Top-1 accuracy**: In top-1 accuracy you give yourself credit for having the right answer if the right answer appears in your first guess. For example - If model returns a model label with highest confidence number, then we denote it as top-1 prediction. Now we can test it whether it is correct or not. The predicted label was Maruti Swift but the ground truth was WagonR then it does not add to the accuracy of model.

- **Top-3 accuracy**: In top-3 accuracy you give yourself credit for having the right answer if the right answer appears in your top five guesses.For example - Now we have seen Top-1 prediction, Top-3 is same as above. Here, window of error is increased i.e. now model returns three guess(prediction). If the model is correctly returning labels(with in three prediction) then it adds to the accuracy.

- **Confusion Matrix**: It will help identifying which cars are mostly confused with one another by the model. It is calculated by forming a matrix, where each of label is mentioned. Now we have five classes for our problem. So matrix of 5x5 is formed. And predicted label and truth label are denoted as both axis. The number of matrix entries denote the total test images which were classified. If original label was WagonR, the predicted label as WagonR should have more number of images. If that is not the case, then our model is not accurate.

- **ROC curves**: It will help identifying the accuracy of model. Area under the ROC curve is accuracy of the model. It is most used evaluation metric. It is curve with True Positive Rate versus False Positive Rate. The area under the curve is denoting the accuracy of the model. More information can be found here - `https://en.wikipedia.org/wiki/Receiver_operating_characteristic`

We can see that each of the evaluation metric tells about the performance of the model in one way or another. The method by which are they calculated are explained.

## 8. Experimental Setup

Firstly computer vision techniques need to applied to our problem. The algorithm/work-flow will be following :

- **Preprocessed Images**:Images should contain the bounding boxes of cars. We considered bounding boxes as size of images. There are 5518 images in total for fine tuning.

- **Train-Test-Validation Split**: The annotated images will be divided in splits. There are 3000 images for training, validated on the image set of 551 and tested on 552 images.

- **Formulation of model**: Tensor flow model of VGG16 and Inception Model.

- **Training methodology**:For fine tuning VGG16, the batch size is 40, base learning rate is 0.0001, decay steps are 7500, decay factor is 0.10, 200 epochs and ADAM optimizer was used. And for fine tuning Inception, the batch size is 32, learning rate is 0.01, decay factor is 0.00004, 1000 epochs and RMSprop as optimizer was used.

The aforementioned model is trained on the locally created database having images of cars clicked in realistic scenarios. In addition to it , we will suggest top-k retrievals.

The demo will comprise of an application running the mentioned models on real-time camera footage to detect the cars.

## 9. Results

The models show the following accuracies on the exercised models using the synthesized dataset: Now our benchmark model was HOG. Let's see how the variation of Inception and VGG16 performed over HOG. The accuracies are better than HOG. The Top-1 precision and Top-3 precision are better than the HOG-based approach.

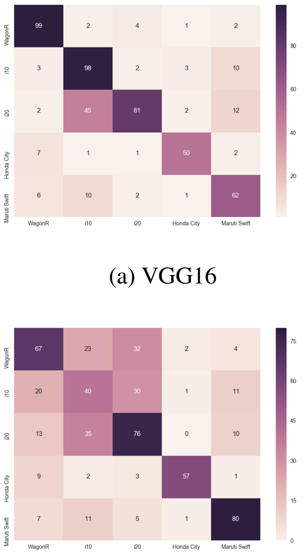
| Model | Top-1 Precision | Top-3 Precision | Evaluation Loss |
|---|---|---|---|
| HOG | 0.6033 | 0.8342 | - |
| VGG16 | 0.8384 | 0.9692 | 0.14 |
| Inception | 0.7825 | 0.9575 | 1.22 |

Table 1: Results

VGG16 performed better than the bench mark and inception model. Therefore VGG16 model is selected as final model. VGG16 model contains 33 convolutions layers. The model is robust to slight change in two models of same firm. VGG16 confusion matrix is better than Inception confusion matrix. It can be observed that classes - i10 and i20 are better classified in VGG16 than Inception. Therefore the model results can be trusted. The state of the art accuracy for such a task is 95 percentage which is for the Compcar dataset. Since the dataset is limited and neural network models requires large amount of data, so the VGG16 accuracy on task dataset and five labels is meeting the solution expectations.

## 10. Conclusion

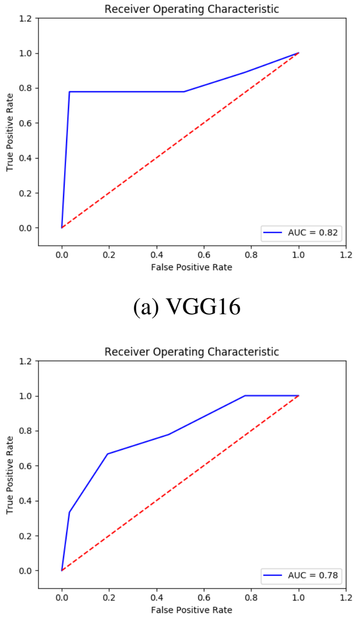
Three models were explored for the given task. All the evaluation metrics were depicting the usage and quality of the model. Here we discuss about the following -



(a) VGG16

(b) Inception

Figure 2: Confusion Matrix of VGG16 and Inception



(a) VGG16

(b) Inception

Figure 3: ROC curves of VGG16 and Inception

### 10.1. Free Form Visualization

Figure 2 contains the confusion matrix of 2 models. We can see that VGG16 is performing better than inception. VGG16 confuses with label i10 and i20. But we can observe that test images were classified accurately. Area under

ROC curves(Figure 6) depicts the accuracies of the model. All the plots and tables are properly labelled and scaled.

### 10.2. Reflection

The problem statement was to classify car up to fined grained level. Only the model of car was selected as the problem. The dataset of the cars according to the Indian models was created and formulated. It was itself an challenging task. But the google images of the cars came to rescue. Collection and refining of images took project 1/3rd time. It was also a risk that the predefined models when fined tuned according to new images would perform.
Then images were labelled. A basic approach of HOG was made as bench mark model. Retrained models of VGG16 and Inception were performing for image classification task's in other dataset. The challenging task was to fine tune the model according to the self formulated and labelled images. The accuracies of the model clearly tells that the image classification task up to fined grained level can be performed by explored approaches, VGG16 being the best.

### 10.3. Improvement

The models developed give the accuracy on the compiled dataset. The task ahead comprises extending the dataset into a structured set having more classes. And each class further divided into subclasses (year of the make). The models are then modified to cater these subclasses. This model would then be trained on this new dataset and perform the fine grain classification.

## References

[1] Jonathan Krause, Timnit Gebru, Jia Deng, Li-Jia Li, and Li Fei-Fei. Learning features and parts for fine-grained recognition. In *Pattern Recognition (ICPR), 2014 22nd International Conference on*, pages 26–33. IEEE, 2014.

[2] Jonathan Krause, Hailin Jin, Jianchao Yang, and Li Fei-Fei. Fine-grained recognition without part annotations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5546–5555, 2015.

[3] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[4] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,*, 2016.

[5] Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3973–3981, 2015.

[6] Bangpeng Yao, Gary Bradski, and Li Fei-Fei. A codebook-free and annotation-free approach for fine-grained image categorization. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3466–3473. IEEE, 2012.