Hybrid Deep Learning Ensemble Model for Improved Large-Scale Car Recognition

Abhishek Verma and Yu Liu
Department of Computer Science
California State University
Fullerton, California 92831, USA
Email: averma@fullerton.edu, liuyuxisu@csu.fullerton.edu

***Abstract*— Smart video based traffic monitoring and surveillance systems for improved security rely upon sophisticated deep learning based computer vision algorithms. In this paper we propose a novel deep learning model to automatically recognize cars on large-scale grand challenge CompCars dataset. Such a task for a computer is difficult due to its fine-grained nature and achieving recognition accuracy close to human experts remains a challenge due to lack of big dataset and machine learning model to detect delicate nuances. Our proposed deep CNN hybrid architecture outperforms previously published classification accuracy by 2.52%, which is a significant improvement considering the challenging nature of the dataset.**

Our method suggests several novelties and advantages over existing methods: First, it uses the GoogLeNet's key architecture - inception modules to efficiently exploit the inception's dimension reduction power and to lower the network cost. Second, inspired by the VGG's uniform and powerful architecture, the method replaces GoogLeNet's auxiliary classifiers into deeper networks with 3x3 convolution components to increase its recognition capability. Third, it is a powerful and efficient network in the way that it represents the ensembles of multiple short and medium depth networks. We believe our method could be useful in other domains that require fine-grained recognition.

Keywords— deep learning; GoogLeNet; VGG Net; relay backpropagation; transfer learning; fine-grained recognition

I. INTRODUCTION

In recent years deep learning has achieved tremendous success in various computer vision tasks. Convolutional neural networks (CNNs) can extract rich features from images and has been instrumental in driving progress in image recognition. In the past few years, training deep convolutional neural networks on large scale dataset such as ImageNet [1] has shown to drastically improve computers' visual detection and recognition capabilities.

Deeper neural networks are expected to result in better performance. However, simply adding more layers can lead to performance degradation. The increase in the size of a neural network results in several challenges. The problems with deeper neural networks are: 1) increasing the number of layers results in growth of parameter size and neural network training faces divergence or slow convergence, and is prone to overfitting [2]; 2) issue of vanishing or exploding gradients relates to gradients becoming either very large or small after the backpropagation step as the error signal propagates back across many layers. This leads to poor adjustment of weights and degradation in performance [2].

Prior attempts made in order to address the aforementioned issues either apply optimization techniques such as refined initialization scheme and batch normalization [18] or modify the network architecture by adding auxiliary classifiers to effectively reinforce information at various points in the network [23]. Several recently proposed deep networks such as VGG Net [3], GoogLeNet [4], ResNet [5], CNDS [23], Dense Net [24], Residual-CNDS [25], VNXK and CKML [26] address the aforementioned issues and achieve significant progresses in image recognition and classification. R-CNN [6] showed great improvements in object detection by adopting two streams of CNNs, one for objects proposal and another for classifications.

Object classification at first level of class hierarchy with highly dissimilar object classes has proven to be successful. With advances in CNN models, recently many researchers gained interest in exploring fine-grained classification, or fine-grained visual recognition (FGVR) [7], where the object classes are closer to each other in terms of visual features [26]. This is a novel and challenging task of great significance. Furthermore, it is much more difficult to distinguish between classes at levels lower down in the hierarchy, i.e., sub-classes compared to classes at the first level. The difference across sub-classes is usually subtle and requires expert knowledge.

Fueled by recent advances in the Convolutional Neural Networks, majority of previous work adopts end-to-end CNN scheme [7]. Popular fine-grained classification datasets are relatively small: Flowers dataset [8] with 102 different types of flowers common in United Kingdom, Caltech-UCSD birds-200-2011 [9] dataset contains 11, 788 images spanning 200 sub-species, Stanford dogs [10] consists of 20,580 images, FGVC aircraft dataset [11] contains 10,000 images from 100 classes of aircraft. Saliency-based sampling for fine-tuning upon VGG-VD network is used in [22]. It combines fully connected layer and spatially weighted fisher vector to achieve 84.54% accuracy on Birds and 71.96% on Dogs datasets. Vehicle model dataset [12] comprises of 3,210 vehicle images

with 107 vehicle models, and 30 images of various colors and illuminations are captured for each model. The authors in [12] construct local tiled CNN based Histogram of Gradients (HOG) [13] features of the frontal view of the car images. However, the recognition task using frontal car image contains the car logo and is easier task compared with using the entire car outlook under various viewpoints, and thereby it is of limited practical use.

Fine-grained classification is in-depth computer recognition and it is challenging for two reasons. First comes from the task itself that the classes are similar and their differences are subtle, thus it is necessary to extract highly distinctive features to achieve good sub-class categorization. Typically, there are two sub-problems that need to be addressed: localization and feature representation of the distinctive parts. Second reason is it is known that not having enough training data leads to overfitting. Very large scale fine grained datasets are not publicly available, which could match the size of non-fine grained datasets such as MS Coco and ImageNet that are extremely large in terms of total number of images and classes.

The overarching goal of our work is to model a high-performing network that can effectively and readily form a concept and capture fine-grained details. Such a network should learn smoothly with minimal loss. We propose a deep neural network with modified auxiliary modules that encourages more expressive local representation, ensuring effective gradients relay and increases the multiplicity. Our network adds auxiliary modules to the GoogLeNet. Our proposed model achieves 2% higher top-1 accuracy on ImageNet when compared with GoogLeNet. Since, ImageNet is considered one of the most challenging computer vision datasets currently available, a 2% improvement is considered as a very strong contribution. Next, we use our model as pretrained from ImageNet and apply it to CompCars dataset [14]. Our model achieves 2.54% top-1 accuracy gain compare with the GoogLeNet [4] and this surpasses all previously published results on this dataset.

The rest of this paper is organized as follows. In section II, we give a brief background work. We discuss the details of large-scale CompCars dataset in section III. In section IV, we present our proposed novel network. Section V presents our experimental approach and discussion of the results. We conclude the paper and suggest future work in section VI.

II. Background Work

A. GoogLeNet

GoogLeNet [4] was introduced along with the inception module in 2014. GoogLeNet is primarily used to improve the performance by reducing computation cost and authors in [4] claim to reduce 2/3 of parameters. It comprises 22 parameterized layers. The authors in [17] explain the central idea of inception module, which is to reduce grid size while expanding the filter banks. It is accomplished by using factorization to replace 3x3 filter to fully connected layers to reduce feature dimensionality and thereby leads to less parameter entanglement.

The reason for the success of GoogLeNet inception model comes from the observation that the input features are correlated, and thus redundancy can be removed by combining them appropriately with a 1x1 convolutional filter branch. Additionally, the network uses various other filter size branches, concatenates the output features from all branches, and thereby provides a meaningful combination for the next layer [17].

B. Relay Backpropagation

Relay backpropagation [2] addresses the issues related to regular backpropagation in deep networks. As network goes deep, parameter size and model complexity grows, that poses great challenge for optimization. We encounter vanishing and exploding gradients phenomenon as the gradients might be prone to either being very large or too small as error is propagated across many layers. This could lead to degradation in the network. Such degradation amplifies as network becomes deeper [2]. The key idea of relay backpropagation is to encourage the propagation of error information in a manner that it goes back up to a certain layer. Authors in [2] introduce one or more interim output modules (including loss layer) to selectively manage the back propagation of error in intermediate segments of the network, such segments comprise of fewer layers and are able to reduce the issue of degradation by minimizing the ensembles of losses.

C. VGG Network

The 16 or 19 layers VGG [3] network has a uniform architecture and yields powerful performance. The entire network is composed of a series of small (3x3) convolutional layers with stride one. The advantage of the architecture it to increase the discriminative power of the rectified linear activation by having more layers (two or three layers as opposed to one, as in the case of 7x7 receptive field) followed by three fully-connected layers. VGG net is a computational heavyweight but provides high classification accuracy.

D. Residual Network

Residual network (ResNet) [5] employs shortcut connections to skip some layers and performs identify mapping to achieve desired output. The basic building block of the ResNet architecture is designed to learn residual functions F(x), where the residual function is related to the functions $F(x) = H(x)$, $F(x) = H(x) - x$, and $F(x) = H(x) + x$. One motivation for introducing these residual functions is that the authors believe that the ideal H(x) learned by any model is closer to the identity function x. Furthermore, instead of having a network learn H(x) from randomly initialized weights, we save training time by learning F(x), the difference of H(x) and x [15] suggests that the ResNet is an exponential ensemble across several layers and introduces the third dimension of network size, which is multiplicity, i.e., the number of ensembles.

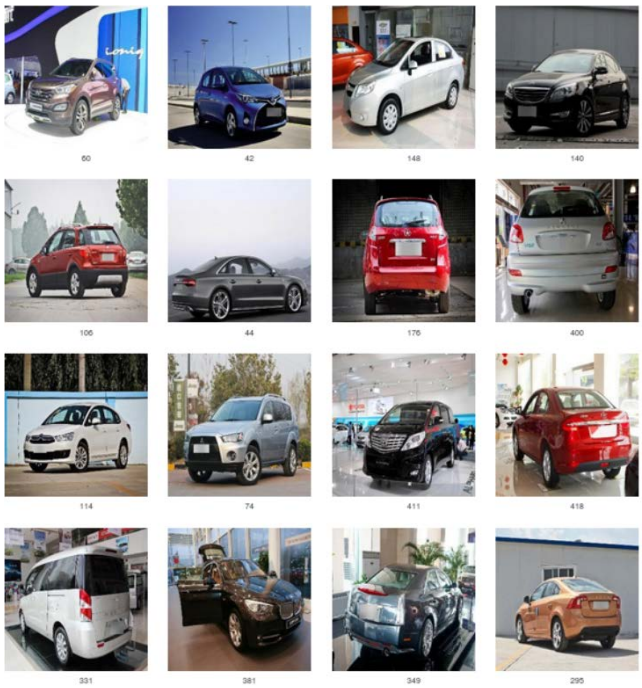

Fig. 1. Sample images from CompCars dataset [14].

E. Transfer Learning

Convolutional neural network features trained from large scale datasets such as the ImageNet or MS Coco datasets can be used as generic image descriptors. Transfer learning refers to the process of using these generic features from a pretrained network such as GoogLeNet and ResNet [14] [27] as a baseline, to bootstrap the learning process on new datasets and further fine-tuning. Initializing with transferred features can improve generalization performance even for features from distant tasks and after further fine-tuning on a new task from the new dataset; it could be a useful technique for improving deep neural network performance. Therefore, we pretrain our proposed network on ImageNet and then fine tune it on CompCars [14] to improve performance.

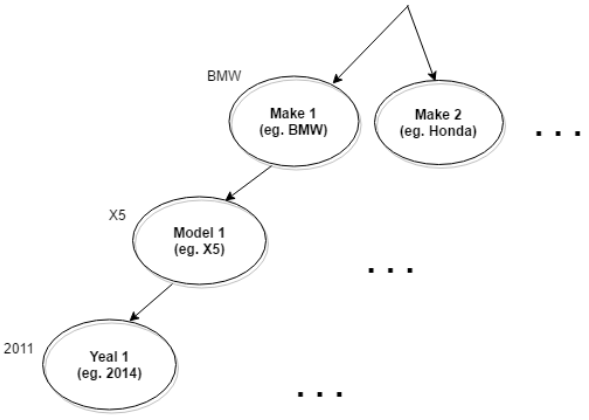

Fig. 2. Hierarchical tree structure of Make->Model->Year :: BMW->X5->2014 for the CompCars dataset [14].

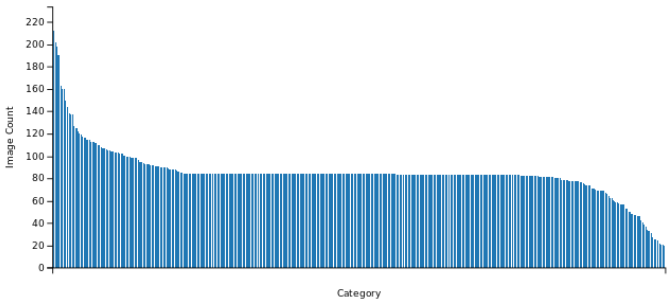

Fig. 3. Train image counts across 431 catagories of CompCars dataset [14].

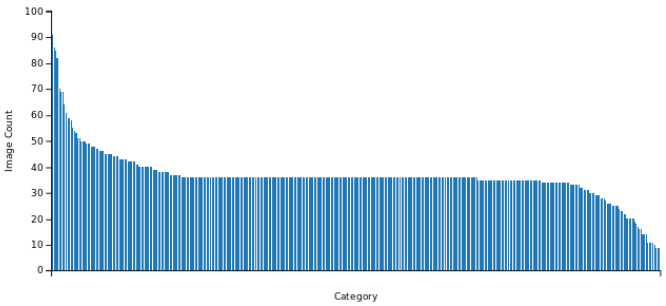

Fig. 4. Test image counts across 431 catagories of CompCars dataset [14].

III. Description of Dataset

Comprehensive cars (CompCars) [14] dataset is a large-scale car image dataset that contains 214,345 images and 1,687 car models. Besides the larger scale, compared with other car datasets, CompCars comprises of images taken from various viewpoints, images from both car interior and exterior car parts.

CompCars contains web and traffic surveillance scenarios and in this paper we focus on the web-nature data. The web-nature car dataset contains 163 car makes and 1,716 car models. See fig. 1 for sample images. These images are organized hierarchically in three levels as make, model and year. There are 12 types of car categories: MPV, SUV, hatchback, sedan, minibus, fastback, estate, pickup, sports, crossover, convertible, hardtop convertible. See fig. 2 for the hierarchical tree structure of a particular BMW instance.

For the car make and model recognition task, [14] selects a subset of CompCars and it contains 431 car models (labeled as 0 – 430 as shown in fig. 3 and 4). Each class contains on an average 100 pictures. The dataset is partitioned into train and verification sets. In [14] authors uses GoogLeNet on car make and model recognition task and achieve top-1 recognition accuracy of 91.2%. For the sake of comparison with earlier published results, we perform our experiments on the same subset of train and verification set as used in [14]. Fig. 3 and 4 shows the train and test image counts across various car categories.

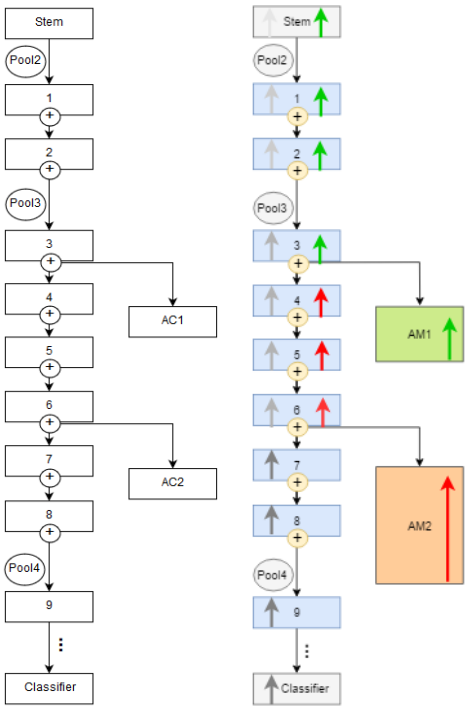

Fig. 5. On left GoogLeNet with 9 inception modules. Mid-level segment consists of modules 3 – 6 (between pool layers 3 and 4). On the right is our proposed model, two enhanced auxiliary modules (AM1 and AM2) along with relay backpropagation maximize the usage of mid-latter features to ensure gradients propagate effectively and encourages expressive local representation.

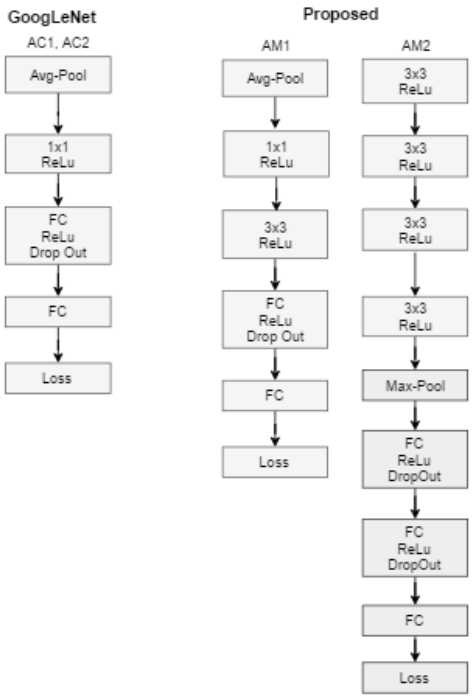

Fig. 6. Comparison of auxiliary modules. On left GoogLeNet's auxiliary components and on the right proposed enhanced auxiliary modules (AM1 and AM2)

CompCars is the largest car image dataset we are aware of to this date. The challenge of this dataset is its large variation of the viewpoint of the same car under various illumination conditions and with little difference between various models of the same car make.

IV. Proposed Methodology

Mid-to-latter features contain distinctive and important information and hence ways to efficiently represent and utilize them are worthy of exploration. An auxiliary part of a network is an additional branch (sometimes with classifier) that comes out from an intermediate layer.

With auxiliary output modules, we provide an elegant way to effectively preserve relevant information by shortening the path from output layer to lower layers, and meanwhile restrain the effect of less relevant information, which would have otherwise propagated through too many layers [2].

Inspired by GoogLeNet's resource utilization efficiency and VGG's feature extraction high performance, we propose a hybrid convolutional neural network that takes advantage of both networks along with the idea of relay backpropagation and improves classification performance. The hybrid model not only provides early supervision and relay of gradients, but also works as ensemble network and offers expressive local representation. In order to further extract high level features, we embed the auxiliary module into mid-latter part of the network.

A. *Auxiliary Components*

GoogLeNet [4] is the first network to propose auxiliary classifier. Relay backpropagation [2] exploits the idea from the perspective of information flow and calls it auxiliary loss. The aforementioned two concepts both address solving the issue of vanishing gradients; however, each holds slightly different interpretation and implementation. Auxiliary module in a network is an additional branch with classifier that comes out from intermediate layers. GoogLeNet recognizes it more as a regularizer and thus only uses 1x1 filter to reduce size, followed by fully-connected layers, and loss layer. Relay backpropagation exploits the idea from the perspective of information flow, and uses it to help propagate the supervision information to shallower layers via intermediate shortcuts. Relay backpropagation uses VGG network and inserts auxiliary modules in the mid-latter region of the network. It uses max pooling layer, 3x3 convolutional layers and two fully connected layers.

We believe the auxiliary component is not only a classifier but also capable of extracting more in-depth features, and thus it could be expanded into a mini network of its own. In this paper, we call our proposed component as auxiliary module. At mid-latter stage, rich and distinctive features are formed and thus this is considered as a mature stage to derive

features. Hence, we insert the auxiliary modules at the mid-level. It summarizes the current sophisticated features and provides a proactive yet mature feedback. In our proposed auxiliary module, we use average pooling, 1x1 filter to reduce dimensionality, followed by 3x3 filter to further extract detailed features, then two fully connected layers, ReLu and drop out is added to regularize the module.

However, we need to carefully choose exactly where to add the auxiliary components. Through experiments, we find premature auxiliary classifier, i.e., adding it too soon in the network slows down the performance because low level features are generic and leads to false classification. Therefore, we stick to the same locations of the auxiliary components as in original GoogLeNet.

Middle level features are considered distinctive and thus it is important to effectively utilize this rich information. We compare GoogLeNet architecture with our model that utilizes enhanced auxiliary components. While the stem and the final classifier remained unchanged, we focus on the middle sections, which comprises of 9 inception modules as segment 1 – 9 as shown in fig. 5. Region between pool 3 and pool 4 is at the middle level and comprises of segments 3-6.

B. Proposed Novel Deep CNN Architecture with Auxiliary Components

GoogeLeNet adopts the auxiliary classifier structure, as showed in figure 6 left and comprises of: one average pooling layer, one 1x1 conv layer and followed by two fully connected (FC) layers and finally loss layer. After having explored several different implementations we conclude having different structures for each of the two auxiliary modules (AM1 and AM2) yields better results. Fig. 6 right shows, in the early stages of the network we implement a shallower module (AM1 in fig. 6 right) by incorporating additional 3x3 convolutional layer. A 1x1 convolution was first introduced in Network in Network [19], where the author's goal was to generate a deeper network without simply stacking more layers. It replaces few filters with a smaller perceptron layer with mixture of 1x1 and 3x3 convolutions.

For extracting features in the latter stage of the network where abstract distinctive features reside we propose deeper AM2 as seen in fig. 6 right. We add four 3x3 convolutional layers.

C. Benefits of Proposed Architecture

1. *Ensures gradients propagate effectively:* Increasing the depth of network introduces the vanishing and exploding gradients and thus simply stacking layers results in poor performance. Although this issue has been partially addressed with by techniques such as rectifier neuron, refined initialization scheme and batch normalization, it remains a challenge in training [2]. The auxilary classifer provides early additional supervision for the network and provides an effective error propagation mechanism. GoogLeNet and relay backpropagation use auxiliary classifier to allievate vanishing gradients.

2. *Encourages expressive local representation by leveraging mature mid features:* Mid-to-latter features contain distinctive information and hence efficiently representing and utilizing them is important. Feature extraction gets more intricate as we go through layers and hence 3x3 filter layer is used to further extract features before classification. Inspired from VGG's high performing architecture, we use a series of 3x3 convolutional layers and followed by three fully connected layers.

3. *Efficiently increase the size of network through ensemble:* The state-of-the-art neural networks such as ResNet [5], GoogLeNet [4] are essentially large number of convolutional network ensembles [15]. In In [15] authors propose the third dimension of neural network size: multiplicity, which is the number of ensembles. Auxiliary modules add to the dimensions of ensembles.

V. Experimental Results and Discussion

A. Setup and Implementation

We ran the experiments on the individual model using Caffe [20], which is an open source deep learning software framework developed by the Berkley Vision and Learning Center. Caffe plugs in into the NVIDIA DIGITS platform [21], which is a Deep Learning GPU Training System. NVIDIA DIGITS [21] is an open source project that enables the users to design and test their neural networks for image category classification and object detection with real-time visualization. The specifications of the system used to perform all the experiments are as follows: Four NVIDIA GeForce GTX TITAN X GPU each with 12GB of VRAM, and two Intel Xeon processors E5-2690 v3 2.60GHz with a total of 48/24 logical/physical cores and 256 GB of main memory

We first train our model on ImageNet and then use pertained model on CompCars dataset. The two auxiliary classifier modules' losses are added up to the total loss at the training stage, auxiliary classifiers are not used during test stage.

B. Experimental Results

The first phase is to train the neural network model on ImageNet. We compare the GoogLeNet model with our proposed model. The result is shown in Table I. Our model improves upon GoogLeNet by 2.07% top-1 and 1.37% top-5 accuracy, which is considered significant considering the hugely challenging nature of ImageNet dataset. Our best

TABLE I
COMPARISON OF THE TOP 1 & 5 CLASSIFICATION ACCURACY (%) OF PROPOSED METHOD WITH GOOGLENET ON THE IMAGENET DATASET AT 50 EPOCHS OF TRAINING

	GoogLeNet	Proposed Method
Top-1	65.83	**67.90**
Top-5	86.83	**88.19**

TABLE II
COMPARISON OF THE TOP 1 & 5 CLASSIFICATION ACCURACY (%) OF PROPOSED METHOD WITH OTHER METHODS ON THE COMPCARS DATASET AT 30 EPOCHS OF TRAINING

	GoogLeNet	GoogLeNet[14]	ResNet	Proposed Method
Top-1	91.32	91.2	91.19	**93.84**
Top-5	98.47	98.1	-	**98.93**

Note: Data not available is marked as '-'. Pretrained ResNet model did not implement Top-5 accuracy.

results were achieved using stochastic gradient descent with decay of 0.0001, base learning rate of 0.01 and momentum of 0.9. The results of GoogLeNet model and our proposed model trained on ImageNet are shown in fig. 7 and fig. 8, at 50 or more epochs of training our proposed model continues to outperform GoogLeNet.

The second phase is transfer learning with pretrained models. We continue training the pretrained models on CompCars. This allows the network performance to mature further as it picks up the car-specific features. Our best results are achieved using stochastic gradient descent with decay of 0.0001, base learning rate of 0.01 and momentum of 0.9.

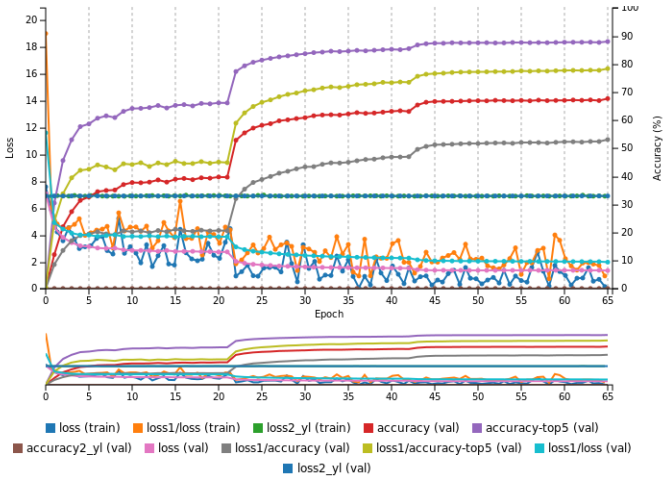

Fig. 8. Training and validation loss. Top 1 and 5 validation accuracy on ImageNet dataset using proposed method.

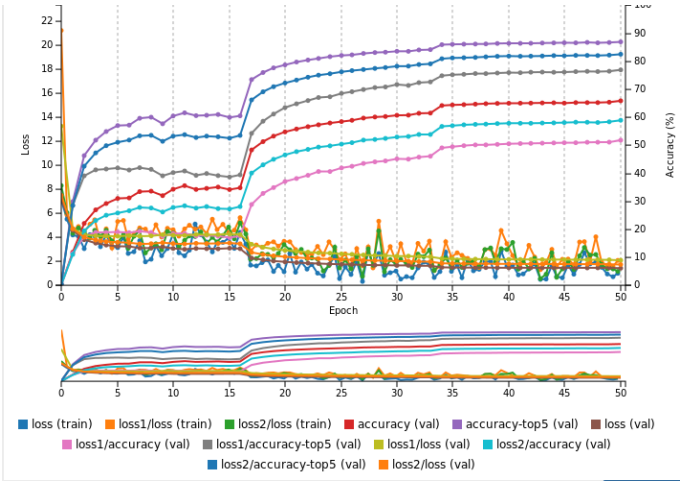

Fig. 7. Training and validation loss. Top 1 and 5 validation accuracy on ImageNet dataset using GoogLeNet.

In order to compare the performance of start-of-the art neural networks with our proposed architecture, we choose to benchmark on pretrained GoogLeNet and ResNet. For GoogLeNet we train with learning rate as 0.001 and batch size as 32. For ResNet, we train with learning rate as 0.001 and batch size as 10. The results are shown in Table II. Authors in [14] used GoogLeNet for car classification task and reported 91.2%. In order to verify authors' results we ran experiments on GoogLeNet (see Table II column 1) and obtained 91.32% accuracy. Our proposed novel hybrid method proves to be about 2.5% higher than GoogLeNet and 2.7% higher than ResNet on the top-1 accuracy. We consider this to be a significant contribution considering the hugely challenging nature of the dataset. The results of GoogLeNet and our proposed model trained on ImageNet are shown in fig. 9 and 10, at 30 or more epochs of training our proposed model continues to outperform GoogLeNet.

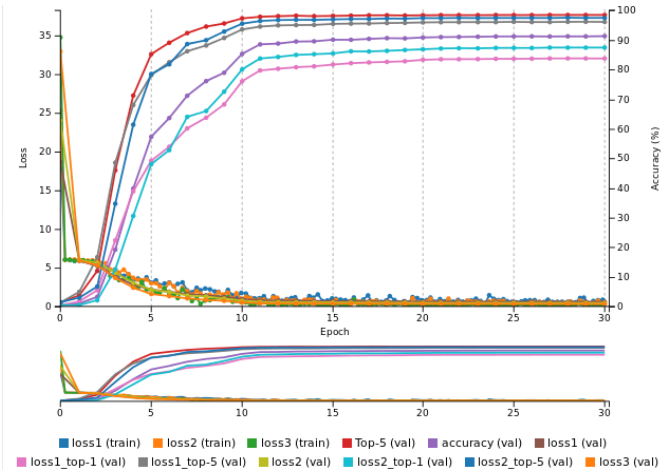

Fig. 9. Training and validation loss. Top 1 and 5 validation accuracy on CompCars dataset using GoogLeNet.

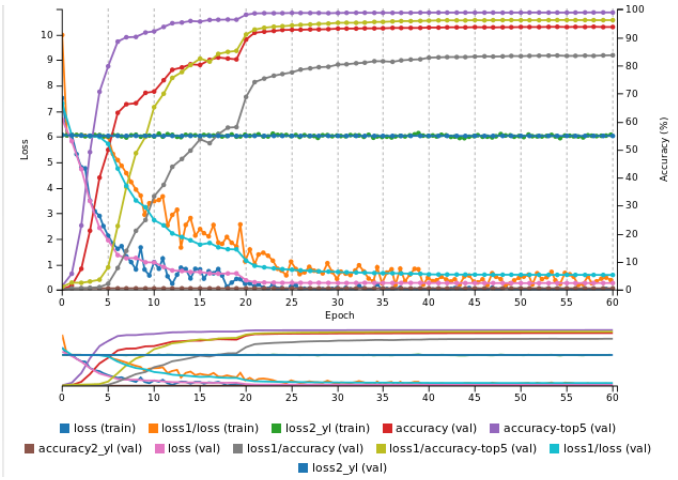

Fig. 10. Training and validation loss. Top 1 and 5 validation accuracy on CompCars dataset using proposed method.

VI. Conclusion And Future Work

We proposed new deep CNN architecture that combines the merits of GoogLeNet's time and space efficiency, VGG's intensive convolution ability and high classification accuracy, and successful relay backpropagation concept to tackle network degradation. Our hybrid network achieved better performance than other published results. Our method suggests several novelties and advantages over existing methods: First, it uses the GoogLeNet's key architecture - inception modules to efficiently exploit the inception's dimension reduction power and to lower the network cost. Second, inspired by the VGG's uniform and powerful architecture, the method replaced GoogLeNet's auxiliary classifiers into deeper networks with 3x3 convolution components to increase its recognition capability. Third the network is a powerful and efficient network in the way that it represents the ensembles of multiple short and medium depth networks. Further exploration could still be done in the direction of firstly adding depth to the auxiliary modules and secondly fusing features from multiple networks.

References

[1] O. Russakovsky et al., "ImageNet large scale visual recognition challenge" *Int. J. on Computer Vision*, vol. 115 , no. 3, pp. 211-252, 2015.

[2] L. Shen, Z. Lin, and Q. Huang, "Relay backpropagation for effective learning of deep convolutional neural networks," *European Conf. on Comp. Vision (ECCV)*, Amsterdam, Netherlands, Oct. 8-16, 2016.

[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *Int. Conf. on Learning Reprsentation (ICLR)*, San Diego, CA, May 7-9, 2015.

[4] C. Szegedy et al., "Going deeper with convolutions," *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Boston, MA, June 7-12, 2015.

[5] K. He et al., "Deep residual learning for image recognition," *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, June 26 – July 1, 2016.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation,". *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Columbus, OH, June 23-28, 2014.

[7] C. Yin and Z. Feng, "Fine-grained categorization and dataset bootstrapping using deep metric learning with humans in the loop," *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, June 26 – July 1, 2016.

[8] M-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," *Indian Conf. on Comp. Vision, Graphics and Image Processing (ICVGIP)*, Bhubaneswar, India, Dec. 16-19, 2008.

[9] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 dataset," *Technical Report CNS-TR-2011-001*, California Institute of Technology, 2011.

[10] A. Khosla, N Jayadevaprakash, B. Yao, and Fei-Fei Li, "Novel dataset for fine-grained image categorization," *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Colorado Springs, CO, June 20-25, 2011.

[11] S. Maji, E. Rahtu, J. Kannala, M. B. Blaschko, and A. Vedaldi, "Fine-grained visual classification of aircraft," arXiv preprint arXiv:1306.5151, 2013.

[12] Y. Gao and H. Lee, "Local tiled deep networks for recogntion of vehicle make and model," *Molecular Diversity Preservation International (MDPI)*, vol. 16, no. 2, pp. 226, 2016.

[13] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Boston, MA, June 7-12, 2015.

[14] L. Yang, P. Luo, C. C. Loy, and X. Tang, "A large-scale car dataset for fine-grained categorization and verification," *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Boston, MA, June 7-12, 2015.

[15] A. Veit, M. Wilber, and S. Belongie, "Residual networks are exponential ensembles of relatively shallow networks," *Neural Information Processing Systems (NIPS)*, Barcelona, Spain, Dec. 5-10, 2016.

[16] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," *Neural Information Processing Systems (NIPS)*, Montreal, Canada, Dec. 8-13, 2014.

[17] C. Szegedy et al., "Rethinking the inception architecture for computer vision", *arXiv preprint arXiv: 1512.00567*, 2015.

[18] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," *Int. Conf. on Machine Learning*, Lille, France, July 6-11, 2015.

[19] M. Lin et al., "Network in network," *Int. Conf. on Learning Representations (ICLR)*, Banff, Canada, April 14-16, 2014.

[20] Y. Jia et al., "Caffe: convolutional architecture for fast feature embedding, " arXiv preprint arXiv:1408.5093, 2014.

[21] NVIDIA DIGITS Software. (2016). Retrieved April 23, 2016, from https: //developer.nvidia.com/digits.

[22] X. Zhang, H. Xiong, W. Zhang, W. Lin, and Q. Tian, "Picking deep filter responses for fine-grained image recognition," *Int. Conf. on Comp. Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, June 26 – July 1, 2016.

[23] L. Wang et al., "Training deeper convolutional networks with deep supervision," arXiv preprint arXiv:1505.02496, 2015.

[24] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," arXiv preprint arXiv:1608.06993, 2016.

[25] H. Al-Barazanchi, H. Qassim, and A. Verma, "Novel CNN architecture with residual learning and deep supervision for large-scale scene image categorization," *IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, New York, NY, Oct. 20-22, 2016.

[26] H. Vo and A. Verma, "New deep neural nets for fine-grained diabetic retinopathy recognition on hybrid color space," *IEEE Int. Symposium on Multimedia*, San Jose, CA, Dec. 11-13, 2016.

[27] Y. Lavinia, H. Vo and A. Verma, "Fusion based deep CNN architecture for improved large-scale image action recognition," *IEEE Int. Workshop on Multimedia Information Processing and Retrieval (MIPR)*, San Jose, CA, Dec. 11-13, 2016.