Fine-Grained Vehicle Classification with Unsupervised Parts Co-occurrence Learning

Sara Elkerdawy, Nilanjan Ray, and Hong Zhang

Computing Science department, Alberta University, Canada
{elkerdaw,nray1,hzhang}@ualberta.edu

Abstract. Vehicle fine-grained classification is a challenging research problem with little attention in the field. In this paper, we propose a deep network architecture for vehicles fine-grained classification without the need of parts or 3D bounding boxes annotation. Co-occurrence layer (COOC) layer is exploited for unsupervised parts discovery. In addition, a two-step procedure with transfer learning and fine-tuning is utilized. This enables us to better fine-tune models with pre-trained weights on ImageNet in some layers while having random initialization in some others. Our model achieves 86.5% accuracy outperforming the state of the art methods in BoxCars116K by 4%. In addition, we achieve 95.5% and 93.19% on CompCars on both train-test splits, 70-30 and 50-50, outperforming the other methods by 4.5% and 8% respectively.

1 Introduction

Vehicle fine-grained classification is a challenging problem in computer vision for multiple reasons. First, in contrast to the classification problem as in ImageNet [1], fine-grained classification deals with different classes within the same category. Secondly, fine-grained classification suffers from scarcity of datasets. Few public datasets for vehicle fine-grained classification exist, such as Cars [2], BoxCars116K [3], and CompCars [4]. Lastly, class hierarchy can be illustrated in three different levels: make, model, and year. Difficulty increases with deeper class definition as the number of samples per class becomes smaller and the visual cues become more challenging to detect.

Multiple methods rely on extra annotation either parts annotation or 3D CAD models such as [5], [2]. These annotations require extensive laborious work and not feasible for large datasets. Ya-Fang Shih et al. [6] proposed a co-occurrence layer evaluated on fine-grained bird-species recognition. COOC layer makes use of the semantic learned features in CNN models that jointly co-occur for a class. On the other hand, Jakub Sochor et al. [3] proposed an unpacking algorithm for vehicle view normalization based on 3D bounding box estimation. They used two networks as a preprocessing for 3D bounding box estimation. Then, they apply fine grained recognition with a third classification network on the unpacked images. Not to mention the unpacking distortion (Fig. 2), the use of three deep networks is computationally expensive for real-time applications such as traffic monitoring and surveillance.

2 Our approach

In this section, we provide a detailed description of the architecture and the two-step fine-tuning procedure.

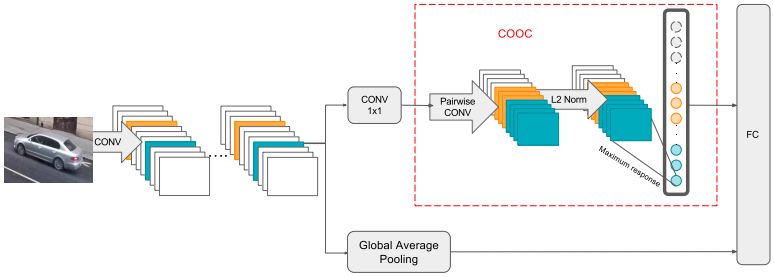

Fig. 1. Overall architecture. Applying Co-occurrence (COOC) layer [6] on the last convolution layers concatenated with Global Average Pooling (GAP).

2.1 Two-step Fine-tuning

Fine-tuning ImageNet pre-trained models as widely shown in practice has better initial weights for the task at hand than random initialization. The network up to the last convolution layers is initialized with the weights trained on a large dataset (e.g. ImageNet). Subsequently, the whole network is fine-tuned including the new randomly initialized last fully connected layers for the new classification problem. However, recent work [7] highlighted that the two-step fine-tuning achieves better results than one-step fine-tuning. The reason for this is that these random weights have high gradient in the first few epochs and it is possible to wreck up the last few learned convolution features. In our paper, a transfer learning first is trained by freezing all the pre-trained initialized weights and updating only the newly added layers for few epochs. This step prevents the high gradient to back-prob into the already learned initial features. Then, after converging, a proper fine-tuning with good initial weights is applied on the whole network.

2.2 Co-occurrence Layer

In fine-grained categorization, the collection of parts detected and recognized is what count to decide on the final categorization. To make use of the part localization learned by deep CNN networks, we exploit co-occurrence (COOC) layer [6]. Co-occurrence (COOC) layer is a trainable end-to-end layer without additional learned weights into the network. It encodes the relationship between the parts learned by the network instead of only a small set of pre-specified

manually annotated parts. Full architecture is shown in Figure 1 where only one COOC block is added after the last convolution layer. In general, COOC layer treats each feature map $F_i \in \mathbb{R}^{m \times m}$ as a filter and calculates the correlation between the feature map F_i and each other feature map F_j. This implicitly enforces learning the co-occurrence of the different visual parts detected by the ith filter and the jth filter, i.e.

$$c_{ij} = \max_{o_{ij}} \sum_{p \in [1,m] \times [1,m]} F_p^i F_{p+o_{ij}}^j \tag{1}$$

where o_{ij} is all possible spatial offsets in the correlation operator, c_{ij} is the maximal response. Finally, for each pair of feature maps F_i and F_j, the maximal correlation response c_{ij} only is used for the final COOC vector for F_i.

Following the baseline ResNet architecture [8], global average pooling (GAP) features and the COOC features are concatenated before feeding into the fully connected layer. A Normalization is applied on COOC features to handle the different range of values from both layers and ensure similar weighting per feature. In addition, 1x1 convolution layer is added to reduce the dimensionality of the COOC layer and also increase correlation between the features. Given an input with N channels, COOC layer output has a size of N^2. Without the 1x1 convolution layer, the high dimensional COOC vector is highly sparse with weak relations between neurons and thus performing useless additional computations.

3 Experimental Results

We did our experiments on BoxCars116k [3] and CompCars [4] datasets. None of these datasets have parts annotation, so we compare only with methods that rely on labels and/or 3D bounding boxes annotation if available. BoxCars116k is a surveillance only fine-grained classification while CompCars has both web-based collected and surveillance nature images. However, the surveillance data in CompCars is far less in size compared to BoxCars116k and contain frontal data only. For this reason, we evaluate on BoxCars116k and the web-collected images in CompCars to show the model in both scenarios with different views. On the training side, we apply data transformation at each epoch to introduce diversity to all images. We use transformations such as color alternation, image drop, random cuts and image flip. We used the same setup for all the models with Adam optimizer, initial learning rate 0.001, batch size 8 for BoxCars116k and 32 for CompCars. In two-step fine-tuning setup, layers initialized with random weights are first trained for 10 epochs before training the whole network for 30 epochs.

3.1 Evaluation

BoxCars116k: The dataset is divided into easy, medium, and hard subsets, based on the fine categorization in the make-model-year hierarchy. Evaluation

Table 1. Classification accuracy in percentage on BoxCars116K. The best accuracy is shown in bold for each split.

Method	Medium	Hard
VGG19 [3]	75.40	76.74
VGG19 + UNPACK [3]	83.98	84.12
ResNet50 [3]	75.07	75.48
ResNet50 + UNPACK [3]	82.28	82.27
ResNet152 [3]	78.44	76.46
ResNet152 + UNPACK [3]	83.80	83.74
ResNet50 + two-step	78.50	79.94
ResNet152 + two-step	81.43	79.76
ResNet50 (Ours)	86.00	**86.38**
ResNet152 (Ours)	**86.57**	85.31

Table 2. Classification accuracy in percentage on CompCars. The best accuracy in 70-30 split (top section) and 50-50 split (bottom section) is shown in bold.

Method	Top-1	Top-5
AlexNet [4]	81.9	94.0
GoogLeNet [4]	91.2	98.1
Overfeat [4]	87.9	96.9
ResNet50 (Ours)	**95.58**	**99.23**
Overfeat [4]	76.7	91.7
BoxCars [9]	84.8	95.4
ResNet50 [4]	90.80	98.16
ResNet50 + two-step	92.42	98.43
ResNet50 (Ours)	**93.19**	**98.98**

is done on the medium and hard subset of the dataset containing 79 and 107 class respectively. We use the provided training-test splits in both datasets for fair comparison with the other methods. Table 1 summarizes the results on BoxCars116k with different architectures compared with [3], baseline CNN models and our method's additional experiments. As can be seen, two-step fine-tuning achieves better results by up to 3.4% in accuracy than one shot fine-tuning. Still models with unpacking outperform two-step fine-tuned baseline models in accuracy by around 3%. However, this is achieved without the 3D estimation and contour finding preprocessing needed for the unpack. In addition, adding the relationship between the last feature maps via cooc layer boosts the performance further by 4% compared to the unpacking method. Also, our network with ResNet50 outperforms deeper networks like ResNet152 with unpacking by 2.2%.

CompCars: There is two training-test split provided in CompCars, one is 50-50 split and the other is 70-30 split respectively. We evaluated on both splits for further comparisons consistency. Results summary are shown in Table 2. Our model outperforms GoogLeNet, the best model, by 4.5% margin. It is also worth noting, that even with less data used in training our 50-50 model outperforms the best 70-30 achieved model by 2%. In addition, outperforming BoxCars that is using the same split by more than 8%. The accuracy gain (1.5%) holds in CompCars as well when applying two-step fine-tuning compared with its counterpart model with one-step fine-tuning.

3.2 Explanatory Analysis

Two-Step Tuning Analysis: To show the effect of the two step fine-tuning on vehicle categorization, visualization with class-activation map (CAM) [10] is

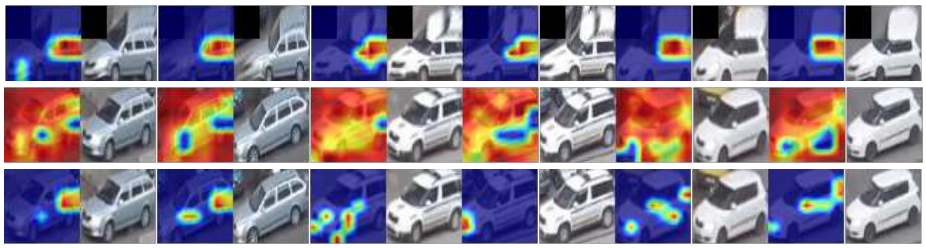

Fig. 2. CAM visualization for the ResNet50 networks trained on BoxCars116. The three rows shows heat map for 3D unboxing[3], one-step fine-tuning, and two-step fine-tuning from top to bottom respectively. Each pair of columns belong to the same vehicle with slight camera rotation but has similar heat maps in two-step fine-tuning.

performed on the last learned features. In CAM, the last layer in the network should be a global average pooling layer (GAP) after the last convolution. This GAP layer is then connected with the fully connected layers and the weights are learned. By doing this, we can know the weight of each feature map j before the GAP layer for each class i by examining the weight W_{ij}. In Figure 2, the heat map for BoxCars [3], one-step fine-tuning, and two-step fine-tuning are shown. BoxCars method, due to 3D unboxing, attends mostly to the side view parts only regardless the category. On the extreme side, the one step fine-tuning with random weights initialization in the last layers gives a heat map that is scattered all over the image. The network did not learn to attend to particular parts of the image although there can be some negatively attended parts (blue). However in the two-step fine-tuning, the network's heat map is more similar to unboxing output with focused attention on certain parts of the vehicle for each category. It is worth noting that the network attends to the same areas/parts for vehicles of the same category even with slight rotations.

Co-Occurrence Analysis: As CompCars has finer high resolution, we visualize the learned features in COOC layer. Figure 3 shows three different categories defined by their make and model. Visualization is done by inspecting the pair of features corresponding to the most activated COOC neuron in a category and displaying the corresponding F_i and F_j maps. As can be seen, the most activated pair of features that jointly occur are recurring within the category. This highlights the importance of COOC layer to capture the relations between the detected features.

4 Conclusion

We have proposed an architecture for fine-grained vehicle classification without part annotation or 3D information. Our approach achieves the best results compared to the state-of-the art methods by a margin 4% on BoxCars116K and CompCars datasets. We utilize the learned high-level features in deep networks with co-occurrence layer to obtain unsupervised part information. In addition,

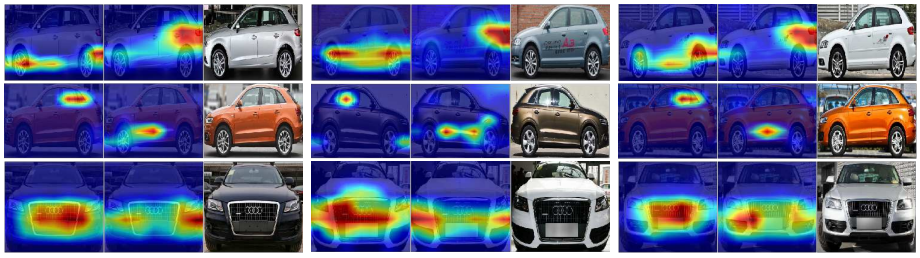

Fig. 3. Co-occurence heat map. Each row is a different vehicle class where each triplet of images represent the two highly jointly activated features and the input image respectively. The pair of features are consistently activated within the same category.

we fine-tune with two steps 1) transfer, and 2) fine-tune for better weights transfer with existent random weights.

References

1. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database. In: CVPR09. (2009)
2. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3d object representations for fine-grained categorization. In: 4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13), Sydney, Australia (2013)
3. Sochor, J., Špaňhel, J., Herout, A.: Boxcars: Improving vehicle fine-grained recognition using 3d bounding boxes in traffic surveillance. arXiv preprint arXiv:1703.00686 (2017)
4. Yang, L., Luo, P., Change Loy, C., Tang, X.: A large-scale car dataset for fine-grained categorization and verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3973–3981
5. Lin, Y.L., Morariu, V.I., Hsu, W., Davis, L.S.: Jointly optimizing 3d model fitting and fine-grained classification. (2014)
6. Shih, Y.F., Yeh, Y.M., Lin, Y.Y., Weng, M.F., Lu, Y.C., Chuang, Y.Y.: Deep co-occurrence feature learning for visual object recognition. In: Proc. Conf. Computer Vision and Pattern Recognition. (2017)
7. Branson, S., Van Horn, G., Belongie, S., Perona, P.: Bird species categorization using pose normalized deep convolutional nets. arXiv preprint arXiv:1406.2952 (2014)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2016) 770–778
9. Sochor, J., Herout, A., Havel, J.: Boxcars: 3d boxes as cnn input for improved fine-grained vehicle recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 3006–3015
10. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2016) 2921–2929