Cooperative Object Classification for Driving Applications

Eduardo Arnold[1], Omar Y. Al-Jarrah[1], Mehrdad Dianati[1], Saber Fallah[2], David Oxtoby[3] and Alex Mouzakitis[3]

***Abstract*— 3D object classification can be realised by rendering views of the same object from different angles and aggregating all the views to build a classifier. Although this approach has been previously proposed for general objects classification, most existing works did not consider visual impairments. In contrast, this paper considers the problem of 3D object classification for driving applications under impairments (*e.g.* occlusion and sensor noise) by generating an application-specific dataset. We present a cooperative object classification method where multiple images of the same object seen from different perspectives (agents) are exploited to generate more accurate classification. We consider model generalisation capability and its resilience to impairments. We introduce an occlusion model with higher resemblance to real-world occlusion and use a simplified sensor noise model. The experimental results show that the cooperative model, relying on multiple views, significantly outperforms single-view methods and is effective in mitigating the effects of occlusion and sensor noise.**

I. INTRODUCTION

Several applications including augmented reality [1], indoor object localization [2] and autonomous robots navigation [3] require object classification. In general, 3D object classification has been addressed using either models that act upon the 3D shape representation or on multiple 2D images from different perspectives, which are then aggregated to generate the final classification output. The spatial resolution in 3D representations must be reduced to maintain computational performance, resulting in loss of details. This can explain why models using multiple 2D images from different perspectives outperform models using an explicit 3D representation [4] and motivates the usage of multiple view images for object classification.

Multiple views of an object can be obtained by rendering 3D models from different perspectives [5], [6]. Similarly, we conceive a cooperative object classification system where multiple agents share their sensor information, *i.e.* camera images, to improve the overall classification result. Note that although we provide the algorithm for such cooperative system, the details of real implementation, such as communication protocols, are out of the scope of this paper. Specifically, we target driving scenarios where autonomous vehicles need to identify the class of objects surrounding them. This is a requirement to comply with specific driving rules, such as the minimal distance to pedestrians, cyclists, *etc.* These real world driving scenarios introduce impairments that are not traditionally explored by most classification methods and can significantly degrade their performance [6]. Among these impairments we investigate object occlusion and sensor noise, particularly for camera sensors. Although occlusion in object classification has been a topic of interest for indoor applications [2], we differentiate our research by considering multiple views of occluded objects in a driving scenario.

This paper presents a "concatenation" method for cooperative object classification in driving applications. Particularly, we address the problem of model generalization and resilience with respect to occlusion and sensor noise impairments. We focus specifically in driving applications by adopting a relevant dataset with a realistic occlusion model. Our contributions are:

- Creation of a 3D object dataset with relevant classes for driving applications.
- Introduction of a realistic occlusion model with higher resemblance to real-world occlusion cases.
- Proposal of a concatenation model for cooperative object classification.
- Experimental assessment of different cooperative architectures for object classification.

This paper is structured as follows. Section II reviews relevant works in the context of object classification and occlusion handling, while Section III describes the proposed model and how it compares to previous research. Section IV describes the dataset generation, impairment models and evaluation metrics, then presents results and discussion regarding model generalisation and resilience to impairments. Finally, Section V concludes this work.

II. RELATED WORKS

3D object classification methods can be categorized into two groups. The first one considers 3D shape descriptors acting directly on the 3D shape. In this group, the descriptors can be "hand-engineered" [7], [8], [9] or automatically learned from data using voxel [10], [11] or point cloud [12], [13] representations. In contrast, the second group renders the 3D shape into multiple images from different perspectives and then extract features from these images to perform classification. The feature extraction can be either based on hand-engineered features, for example using Scale Invariant Feature Transform (SIFT) [14] and Fisher vectors [15], or learned with a deep-learning based approach [16].

In the first group, *3D ShapeNet* [10] uses a convolutional deep belief network to learn the joint distribution of volu-

This work was supported by Jaguar Land Rover and the U.K.-EPSRC as part of the jointly funded Towards Autonomy: Smart and Connected Control (TASCC) Programme under Grant EP/N01300X/1.

[1]Warwick Manufacturing Group (WMG), University of Warwick, Coventry, U.K. e.arnold@warwick.ac.uk

[2]Centre for Automotive Engineering, University of Surrey, Guildford, U.K.

[3]Jaguar Land Rover Ltd., Coventry, U.K.

metric representation (voxels) of 3D objects and their class labels. In contrast, *PointNet* [12] uses point cloud data, *i.e.* 3D points, to perform object classification and segmentation. Despite improvements in this group, the dimensionality and low resolution of voxelized 3D inputs still undermines the classification performance of these methods [4].

In the second group, Su *et al.* [5] render each 3D model in 12 different views and then propose a convolutional neural network architecture to generate a 3D descriptor of the object based on the rendered views. This descriptor is used both for object classification and retrieval. Further work by Kanezaki *et al.* [6] explored the connection between pose estimation and classification: when the pose of the object is known, the classification task becomes easier, likewise if the class is known. Their *RotationNet* model takes multi-view inputs and predicts both pose and class for each image, selecting the object class that maximizes the overall class likelihood. Furthermore, the object pose is treated as a latent variable, allowing unsupervised training on the pose with an unaligned dataset.

Considering occlusion, Meger *et al.* [2] train image classifiers on full objects and sub-parts to detect occluded 3D objects in an indoor scenario. They formulate the presence of an object under a Bayesian framework considering size priors, depth and structure-from-motion posteriors. Yilmaz *et al.* [17] explore recurrent connections on convolutional networks to overcome occlusion on image classification tasks. Despite classification performance improvement, a naive occlusion model is used, where black rectangles are drawn upon original images. Chandler *et al.* [18] generate an occluded dataset using more diverse occlusion models, but limited in the number of classes and samples. Their method uses an in-painting technique to overcome occlusion, but in turn requires segmentation and annotation of the occlusion degree to be effective.

Our research uses elements of [5] to investigate the effect of occlusion and sensor noise into object classification in the scope of autonomous driving. We differentiate our research from previous works considering object occlusion by exploring multiple views and employing a more realistic occlusion model along a sensor noise model. In doing so, we propose a new concatenation model that overcome occlusion by using information from multiple views. Contrasting to previous occlusion-aware methods, the proposed method does not require occlusion labelling or segmentation. Furthermore we use a dataset with relevant classes for driving applications.

III. METHODS

In this section we first describe our proposed concatenation model as well as two comparative models available in the literature. We then present the training procedure used for evaluation. We call multi-view methods interchangeably as cooperative methods.

A. Model

As a baseline model we adopt a Convolutional Neural Network (CNN) model based on the VGG-11 architecture [19]. This architecture consists of 11 weight layers, 8 convolutional (CNN) and 3 fully connected layers (FCN). The convolutional layers extract features, generating a feature map, with dimensions 7 x 7 x 512, that represents the whole image. On the other hand, the fully connected layers transform this feature map into a class distribution. The network input consists of a 224 x 224 RGB image, while the output represents a distribution over the classes and is normalized using the *soft-max* activation function. This architecture can be easily adjusted for single-view object classification by simply changing the number of output units in the last fully connected layer to fit the current dataset.

We extend this single-view classifier to a cooperative approach through a concatenation model, as in Figure 1c. In this model, the images from n different views are processed independently by the same convolutional stage of the baseline model, which generates a set of n feature maps. Following feature extraction, all n feature maps are concatenated into a single one, which is then fed to a fully connected network (FCN) composed of three layers. This allows to exploit all the information available in each view, which will be highly significant in cases of occlusion and accentuated sensor noise. Similarly to the baseline model, the last layer represents a class distribution.

We implement two different models based in previous literature [5] used for cooperative object classification to provide a comparison against our proposed concatenation model. The first is a voting based model and the second a view-pooling model. Both are described below.

An ensemble model consists of a set of classifiers whose individual results are averaged to produce more accurate results [20]. The voting scheme borrows output averaging from model ensembles, however considers a single classifier, averaging the different views' output instead. Specifically, the voting method independently classifies all the views using the same network (architecture and weights), then averages the resulting class distributions to get an overall object distribution. Note the assumption that all the views contain the same amount of information, thus the output average has equal weight for each view.

Contrasting to the voting method which employs late fusion, the view-pooling method [5] fuses information from multiple views at the feature level. The procedure is the same as in our concatenation model, however after extracting the features there is a sampling operation in the n feature maps (dimension 7x7x512), that results in a single feature map with same dimensionality. The sampling is achieved through the element-wise *max* operation along the feature maps view's axis, and inherently causes loss of information.

B. Training

We train each model independently on the same dataset, described in Section IV-A. For model comparison we train all three cooperative models and n single-view models, where n is the number of views. Each single-view model is trained on a specific view, which allows to compare classification performance among different views.

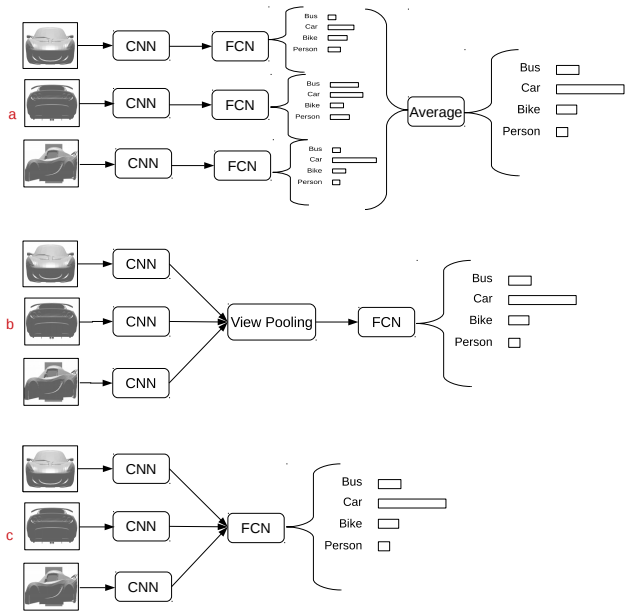

Fig. 1. Cooperative models for object classification with number of views $n = 3$. (a) voting, (b) view-pooling and (c) concatenation (proposed) models.

The size of our dataset limits the training of such deep learning models since training large capacity models without enough data results in over-fitting. Transfer learning is used to overcome this challenge. We use the weights of a VGG-11 model trained on the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 dataset [21], which contains 1.3M images for training distributed over 1000 classes. The last fully connected layer cannot be used since our dataset has different number of classes, so we replace it with a random initialized weight layer with a unit for each class in the dataset.

During the training procedure we assume that the convolutional layers can generate discriminative features and, thus, do not optimize these layers' weights. On the other hand, the fully connected layers are optimized during training, allowing fine-tuning to our dataset. We tried optimizing the parameters from the last convolutional layer but did not obtain any significant performance gain.

We employ standard Stochastic Gradient Descent (SGD) optimization with 10^{-3} learning rate, 0.9 momentum and batch sizes of 64 for all methods, except for concatenation which uses 32 samples for batch. Each model is trained for 10 epochs. We use dropout regularization with $p = 0.5$ after each fully connected layer. We use a weighed cross-entropy loss function, where the weight of a class is the inverse of the number of training samples. This allows the model to generalize for all classes, despite the imbalanced dataset.

IV. EXPERIMENTS AND RESULTS

The presented methods are evaluated using a dataset with relevant classes for driving applications. The dataset generation process is described, as well as the impairment models and evaluation metrics used. Finally, the results comparing cooperative vs single-view methods and a comparative analysis of cooperative methods regarding resilience to impairments are presented.

A. Dataset

Despite a few datasets of general objects present multiple views of objects [22], [10], they use a particular camera configuration and limit the introduction of occlusion, since the images are real-photographs or already rendered 3D models. For this reason we use a standard 3D objects model dataset to render our own multi-view object dataset. Although there are a few options of 3D model datasets [23], [24], the ShapeNet dataset [10] offers the widest collection of 3D models. The original core dataset has 51,300 models distributed over 55 classes, while the segmentation dataset has 12,000 models over 270 classes, which are densely labelled. We select a subset of relevant classes for the driving context from the ShapeNet core and segmentation datasets as our collection of 3D models.

The resulting set has 3268 object models distributed among nine classes. We randomly divide the resulting set of models into training and test sets with ratio 0.7 and 0.3, respectively. The classes have very unbalanced frequency of occurrence, as observed in the histogram presented in Figure 2. Note that this histogram represents the 3D models and not image samples, which will be a multiple of the number of model samples. The final dataset corresponds to the rendered images of the set of 3D models. Although some annotation regarding the pose of object is available we had to manually rotate some of the models to obtain a canonical upright pose.

We now describe the render process using a computer graphics renderer. The models are imported without texture information, which increases the classification complexity, since colour information is a discriminant factor. Then the objects sizes are normalized to unit size along the longest dimension, as real size information is not available for all models. This implies that objects such as animals may seem to have the same size as cars, for example. However, considering a preliminary step of object detection where the object of interest was cropped and scaled, this is a valid assumption. We use a smaller number of views to simulate a cooperative perception system where only a limited number of agents can share their sensor information. Three cameras are placed in a circle centred in the target object, the angle between each adjacent camera is 90 degrees. The cameras face the target object and are tilted by 10 degrees around the x axis. Figure 3 illustrates the rendering settings with the corresponding rendered images.

B. Impairment models

We model two visual impairments that are common in driving scenarios: object occlusion and sensor noise. The occlusion model consists of introducing an occluding object to mask part of the target object. The adopted occluding object is a cube which is placed on a circle of radius 0.6

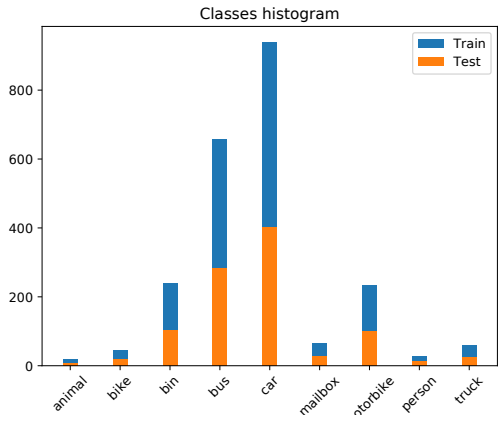

Fig. 2. Class histogram for 3D models in our dataset, divided in training and test sets. The 3D object models sum to 3268 samples.

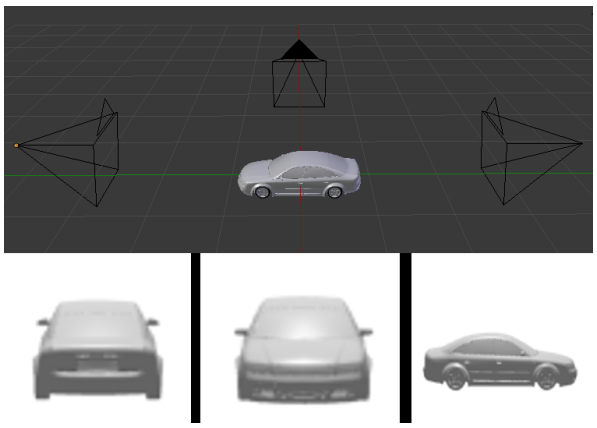

Fig. 3. Top: render cameras settings. The three cameras are placed in a circle and tilted by 10 degrees around the X-axis. Bottom: rendering results of the cameras on a car sample.

(relative to the object's maximum dimension of 1) around the target object, in a random angle that changes for each sample object. The random angle is sampled from a uniform distribution between -90 and 90 degrees to ensure that the target object is occluded in at least one of the views. The cube size modulates the level of occlusion and is used as a parameter to verify different occlusion degrees.

Noise models for image sensors can be categorized in fixed-pattern, banding and random noise [25]. The source of random sensor noise in digital cameras can be photon emissions, photoelectric effects and thermal noise. This noise can be modelled as Additive White Gaussian Noise (AWGN) to pixel intensities [25]. Although most models use an AWGN with a signal dependent variance, we simplify the model by considering a fixed variance, which controls the intensity of the noise. Contrasting to the occlusion model, the AWGN model can be introduced after rendering the images, not requiring to synthesize the dataset again for each noise realisation.

C. Evaluation metrics

We use confusion matrices to evaluate the performance of a multi-class classifier. A confusion matrix element C_{ij} represents the number of elements of ground-truth label i that were classified as class j. Considering the unbalanced nature of the dataset we normalize the matrix along the rows to ease visualisation.

Although the confusion matrix can give insightful information about the classifier, it is useful to have a single metric to compare between classifiers. Metrics such as precision and recall consider the effects of false negatives and false positives respectively. We use the F-measure metric, which is the harmonic mean between precision and recall and is sensitive to both precision and recall. Note that these metrics are class-specific, and the generalized score is obtained through the weighted mean of individual classes.

D. Cooperative vs single-view comparison

We compared cooperative vs non-cooperative (*i.e.* single-view) methods by evaluating their generalisation to occlusion and noise impairments. The models were trained on an impairment free dataset, then evaluated on two experiments with impairments. The first introduced occlusion with cubes sized 0.3, which corresponds to 30% of the object's largest dimension. The second experiment used the same occlusion model and included AWGN to the pixel intensities with standard deviation $\sigma = 0.05$. The results of both experiments are presented in Table I, where the metric used is the class-weighted F1 score. We also detail the confusion matrices of multi-view methods on experiment 2 in Figure 4.

Noise significantly degraded classification performance, more intensely for classes that have ambiguous appearance such as car and bus, as evidenced by the confusion matrices of experiment 2. Particularly to the voting scheme, the bus class receives many false positives. This is also observed for non-cooperative classifiers on views 1 and 2, where trucks and cars show many miss-classifications. This phenomena happens due to the car, bus and truck classes having similar characteristics and the fine distinction between them being corrupted by noise. Overall, the models presented good generalisation to small occlusions (0.3 size), but classification performance degraded drastically with sensor noise.

Cooperative classifiers showed better generalisation capabilities and can handle occlusion to a better degree when compared to single-view schemes. Under occlusion only (Experiment 1), cooperative schemes outperform all single-view classifiers, except for the voting scheme. Considering occlusion and sensor noise, the concatenation scheme is the only able to outperform all individual single-view classifiers under these conditions. This is due to the higher capacity of the concatenation model, allowing to weight the contributions from specific views without information loss caused by sampling.

E. Resilience to Impairments on Cooperative methods

The previous results indicated an advantage in using cooperative methods for object classification regarding gen-

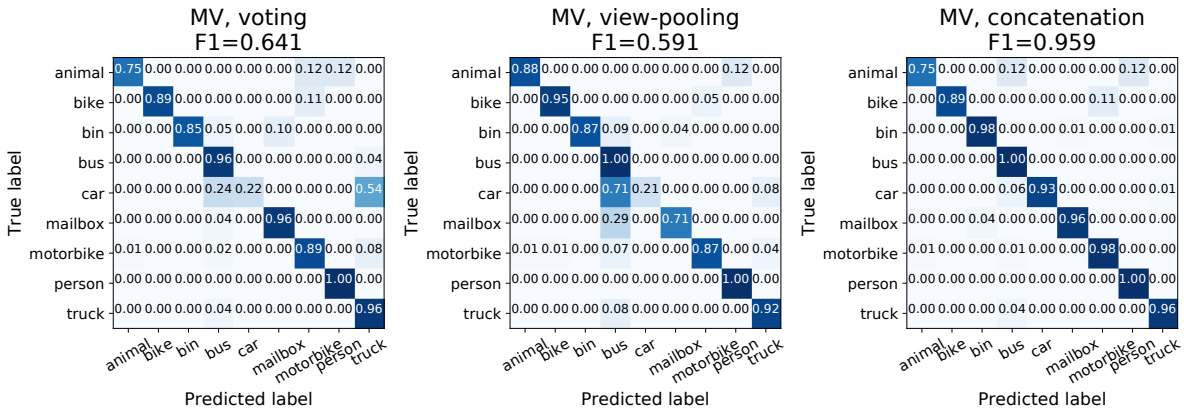

Fig. 4. Results from experiment 2: models trained on impairment free dataset evaluated with occlusion cubes sized 0.3 and AWGN $\sigma = 0.05$.

TABLE I

F1-SCORE FOR DIFFERENT MODELS IN EXPERIMENTS 1 AND 2.

MODEL	EXPERIMENT 1	EXPERIMENT 2
SV0	0.887	0.773
SV1	0.898	0.640
SV2	0.845	0.759
MV voting	0.891	0.641
MV view-pooling	0.899	0.591
MV concatenation (proposed)	0.979	0.959

eralisation to scenarios with fixed occlusion and noise power. Next we evaluate the resilience of these cooperative methods to varying degrees of noise and occlusion, that is, how classification performance degrades as impairments increase. First we trained the classifiers on a fixed impaired dataset containing both occlusion cubes sized 0.3 and AWGN with standard deviation $\sigma = 0.05$. Then we evaluated these classifiers on two experiments. Experiment 3 verified the classification performance considering varying levels of occlusion, parametrized by the occlusion cube size varying between 0 (no occlusion) to 0.5 (half of the object's largest dimension) in steps of 0.05. Analogously, experiment 4 measured the performance of classifiers on an occlusion free scenario with varying sensor noise power, parametrized by the Gaussian standard deviation σ varying between 0 and 0.15 in steps of 0.01. The results are presented graphically on Figures 5 and 6, respectively.

Firstly, cooperative methods performed even better when trained with occlusion, since it forced the network to adapt to the missing information, as noted comparing results from Experiments 3 to 1 in Table I. View-pooling and voting schemes had similar performance, despite the superiority of the former in most cases. However, the proposed concatenation scheme outperformed both of them for all degrees of occlusion and sensor noise. This is expected since no information is lost on the concatenation scheme, contrasting to pooling which discards information and voting which only fuses high level class distributions. We would expect some performance drop as the occlusion level increases.

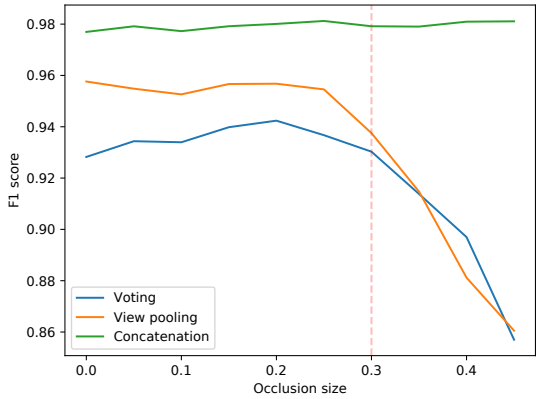

Fig. 5. Results from experiment 3: models trained on impaired dataset evaluated with varying occlusion levels. The highlighted vertical line indicates the occlusion level used for training.

Nonetheless, the concatenation model showed invariance to all degrees of occlusion and sensor noise. This suggests that the model has learnt to use cues from different views to overcome occluded parts, demonstrating the capability of overcoming impairments up to occlusion boxes of size 0.4.

Despite surpassing previous methods classification performance, the concatenation model poses an important limitation for practical usage. Due to the fully connected layer after concatenating all the feature maps, the model requires a fixed number of input views, the same used during training. Not only the number of views has to be the same, but the views should have the same order (pose of the object relative to each camera), since the weights of the fully connected network will fit to each particular view. For example, if we shuffle the input frames to the concatenation model the performance is expected to drop.

V. CONCLUSION

This paper presented a novel method of cooperative object classification, where multiple images of the same object seen from different perspectives (agents) are exploited to generate

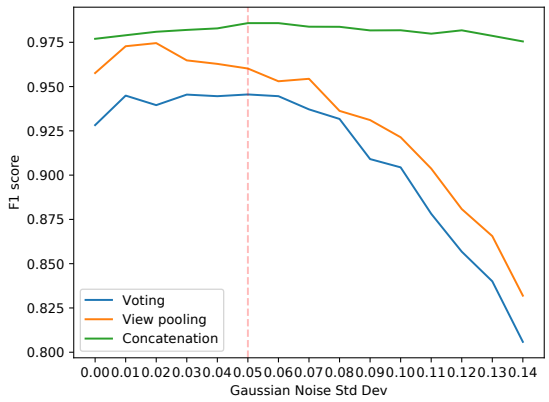

Fig. 6. Results from experiment 4: models trained on impaired dataset evaluated with varying sensor noise levels. The highlighted vertical line indicates the noise standard deviation used for training.

more accurate class predictions. Particularly, our experimental assessment took into account model generalisation and resilience with respect to impairments. We introduced an occlusion model with higher resemblance to real-world occlusion and a simplified sensor noise model. The results showed that the proposed method can significantly improve classification performance, especially in the presence of severe occlusion, when compared to single-view methods and previous approaches.

Our work limitations is the assumed object detection step, which is non-trivial in practice, specially considering object pose detection. Nonetheless, the model architecture should be general to be used in a multi-view, real-world dataset. In such a real-world setting the main constraint would be the alignment of camera poses. That is, the proposed model assumes a particular order of objects poses, and since vehicles will be moving, the object images would have to be sorted in such a way that the objects poses is presented in the order used for training. Another limitation is that our proposed concatenation method relies upon a fixed number of views, which limits the application to scenarios with a fixed number of agents. Future work should generalise the classification model into a 3D object detection model, where the whole scene is scanned for objects, this will allow to obtain object's position and size, other than class.

References

[1] J. Rao, Y. Qiao, F. Ren, J. Wang, and Q. Du, "A mobile outdoor augmented reality method combining deep learning object detection and spatial relationships for geovisualization," *Sensors*, vol. 17, no. 9, p. 1951, 2017.

[2] D. Meger, C. Wojek, J. Little, and B. Schiele, "Explicit Occlusion Reasoning for 3d Object Detection," in *Procedings of the British Machine Vision Conference 2011*. Dundee: British Machine Vision Association, 2011, pp. 113.1–113.11.

[3] A. Teichman, J. Levinson, and S. Thrun, "Towards 3d object recognition via classification of arbitrary object tracks," in *2011 IEEE International Conference on Robotics and Automation*, May 2011, pp. 4034–4041.

[4] C. R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, and L. J. Guibas, "Volumetric and Multi-view CNNs for Object Classification on 3d Data," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 5648–5656.

[5] H. Su, S. Maji, E. Kalogerakis, and E. Learned-Miller, "Multi-view Convolutional Neural Networks for 3d Shape Recognition," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 945–953.

[6] A. Kanezaki, Y. Matsushita, and Y. Nishida, "RotationNet: Joint Object Categorization and Pose Estimation Using Multiviews from Unsupervised Viewpoints," p. 24, 2018.

[7] B. K. P. Horn, "Extended gaussian images," *Proceedings of the IEEE*, vol. 72, no. 12, pp. 1671–1686, 1984.

[8] M. Kazhdan, T. Funkhouser, and S. Rusinkiewicz, "Rotation invariant spherical harmonic representation of 3 d shape descriptors," 2003.

[9] R. Osada, T. Funkhouser, B. Chazelle, and D. Dobkin, "Shape distributions," *ACM Transactions on Graphics (TOG)*, vol. 21, no. 4, pp. 807–832, 2002.

[10] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An Information-Rich 3D Model Repository," no. arXiv:1512.03012 [cs.GR], 2015.

[11] D. Maturana and S. Scherer, "VoxNet: A 3d Convolutional Neural Network for real-time object recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2015, pp. 922–928.

[12] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3d Classification and Segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 77–85.

[13] X. Chen, Y. Chen, and H. Najjaran, "3d object classification with point convolution network," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sept. 2017.

[14] D. G. Lowe, "Object recognition from local scale-invariant features," in *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, vol. 2. Ieee, 1999, pp. 1150–1157.

[15] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the fisher kernel for large-scale image classification," in *European conference on computer vision*. Springer, 2010, pp. 143–156.

[16] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, vol. 2. IEEE, 2004, pp. II–104.

[17] O. Yilmaz, "Classification of occluded objects using fast recurrent processing," in *Machine Learning and Applications (ICMLA), 2015 IEEE 14th International Conference on*. IEEE, 2015, pp. 805–812.

[18] B. Chandler and E. Mingolla, "Mitigation of effects of occlusion on object recognition with deep neural networks through low-level image completion," *Computational intelligence and neuroscience*, vol. 2016, 2016.

[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[20] D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *Journal of artificial intelligence research*, vol. 11, pp. 169–198, 1999.

[21] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[22] K. Lai, L. Bo, X. Ren, and D. Fox, "A large-scale hierarchical multi-view rgb-d object dataset," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1817–1824.

[23] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1912–1920.

[24] Y. Xiang, W. Kim, W. Chen, J. Ji, C. Choy, H. Su, R. Mottaghi, L. Guibas, and S. Savarese, "Objectnet3d: A large scale database for 3d object recognition," in *European Conference Computer Vision (ECCV)*, 2016.

[25] X. Jin and K. Hirakawa, "Approximations to camera sensor noise," in *Image Processing: Algorithms and Systems XI*, vol. 8655. International Society for Optics and Photonics, 2013, p. 86550H.