Deep Learning Based Architecture Reduction on Camera-Lidar Fusion for Autonomous Vehicles

Mihreteab Negash Geletu, Thomas Josso-Laurain, Maxime Devanne, Mengesha Mamo Wogari, Jean-Philippe Lauffenburger

Deep Learning Based Architecture Reduction on Camera-Lidar Fusion for Autonomous Vehicles

Mihreteab Negash Geletu[†‡]
mihreteab.negash@aau.edu.et
mihreteab-negash.geletu@uha.fr

Thomas Josso-Laurain
IRIMAS-UR7499
Université de Haute-Alsace
Mulhouse, France
thomas.josso-laurain@uha.fr

Maxime Devanne
IRIMAS-UR7499
Université de Haute-Alsace
Mulhouse, France
maxime.devanne@uha.fr

Mengesha Mamo Wogari
AAiT-SECE[†]
Addis Ababa University
Addis Ababa, Ethipia
mengesha.mamo@aau.edu.et

Jean-Philippe Lauffenburger
IRIMAS-UR7499[‡]
Université de Haute-Alsace
Mulhouse, France
jean-philippe.lauffenburger@uha.fr

***Abstract*—Autonomous vehicles (AVs) are the dream of the present era and are close to become reality. In AVs, perception is a challenging task. It gives understanding of the driving environment. One type of such task is road detection, where the goal is to segment the road area into drivable and non-drivable using multi-modal sensors like cameras and lidars. For their ability on a road detection task, deep neural networks, with an encoder-decoder architecture, are chosen in this paper. Since deep learning models have large size and AVs have constrained computational power, model reduction is important. Therefore, architecture reduction of a convolutional neural network is proposed on a deep learning based multi-modal fusion model. This model is used as the baseline of our work, and camera and lidar are its modalities. The baseline model's weights that are used to fuse the camera processing pipeline with the lidar pipeline are analysed. The analysis shows that the strength of fusion between the two modalities changes from layer to layer. Using this result and a support from generic encoder-decoder architecture, a reduced architecture is proposed. The latter is further processed by removing some layers of the baseline to produce a lite model. The reduced architectures are validated to show comparable performance with the baseline. Furthermore, both the reduced architectures outperform the baseline on a brightness adjusted camera image. These reduced architectures can be used from the perspective of embedded system, or they can be used to boost performance by appending additional algorithm. The training and validation are done on the KITTI dataset.**

***Keywords*—architecture reduction, sensor fusion, perception, autonomous vehicle, deep learning**

I. Introduction

Autonomous Vehicles (AVs) are sought-after systems with an aim of reducing traffic accidents, mitigating traffic congestion, reducing emissions, increasing mobility, and using infrastructure efficiently. AVs have perception, path planning and control units in the modular approach [1]. Therefore, the vehicle can perceive the environment. This understanding of the driving area is used to generate path trajectory and actuate the vehicle.

Perception in AVs is a challenging task. It maps the driving environment, and gives the world knowledge for autonomy. It involves multi-modal sensors like cameras, lidars, radars or ultrasonic sensors [1]–[3]. Failure in perception propagates in all the succeeding modules, and may initiates a wrong control action. In perception systems, there are various tasks like object detection, semantic segmentation, road and lane detection, localization and mapping, and so on [1], [4]. For this tasks, deep learning becomes a standard approach, and has shown outstanding performance [1], [5]–[7]. It employs convolutional neural network (CNN) to extract features. However, sensors have their own inherent limitations, and a uni-modal perception system will have a constrained performance.

Sensor fusion is used to overcome the inherent shortcomings [1], [2]. There are different scenarios that affect sensing modalities: illumination change, rain, snow, night, entering or leaving a tunnel and so on. Cameras are rich in features, but they are affected by change in illumination levels, non visible or partially visible objects, etc. Lidars are sparse though they are not affected by illumination variation. The robustness to weather condition of radar is limited by its low accuracy. Thus, the cons of one modality need to be complemented by the pros of the other by multi-modal sensor fusion [8], [9]. The fusion can be at early, middle or late stage of the processing pipeline [3]. However, the fused architecture could become large in size because of the multiple modalities (or sensors). Therefore, it is important from the perspective of real-time execution to have a reduction in the size of fusion architectures while maintaining comparable performance [10]. Besides, reduced architectures may have different level of robustness to change in illumination level [11]. Furthermore, this reduction can also be used to extend the perception systems with uncertainty handling methods [12], [13]. In this paper two

This work is funded by Ethiopian Ministry of Education and the French Embassy in Ethiopia and to African Union under the Ethio-French PhD scholarships program in engineering.

reduced models with comparable performance, and enhanced robustness for change in level of illumination are proposed for a camera-lidar fusion architecture. The reduction has used model weight analysis to see the degree of fusion between the two modalities. This analysis together with selection of appropriate architecture resulted in the reduced models.

The rest of this paper is organized as follows. Section II discusses related works, and section III gives a background discussion on a baseline multi-modal fusion architecture that will be reduced in size. The method of reduction and the proposed reduced architectures are presented in section IV. Finally, the experimental results are reported in section V, and Section VI concludes the paper.

II. Related work

Sensor fusion architectures based on deep learning have been proposed for perception in AVs. Camera and lidar are fused using a multi-layer perceptron (MLP) in [14]. The fusion is done after camera and lidar inputs are processed using a CNN based detector called YOLO [15]. The detection from the individual modalities is re-scored by the fusion. This work has shown that up-sampled lidar maps can be used for object detection. Besides, MLP is also used for a purpose of fusion. The same modalities are also fused at decision level in [16]. In [17], YOLO is also used camera-lidar fusion by a weighted-mean of the prediction bounding boxes. The confidence score is used to weight the detection and produce better result than the single modalities. A fusion throughout a processing pipeline of camera and radar using concatenation operation is done in [18]. The fusion showed the advantage of radar in sever weather condition. Differently, authors in [19] propose to investigate fusion at different stage of the processing pipeline: early fusion, late fusion and cross fusion. Respectively, the fusion should occur at the start, at the end or throughout the pipeline. Early and late fusion use concatenation as fusion operation at early and late stage of processing, respectively. However, the fusion called cross-fusion is implemented by combining camera and lidar features at each processing layer of a fully convolutional neural network (FCN). The fusion is done using learnable parameters. Therefore, the position and extent of fusion is not fixed, rather it is determined by learning. It is reported that the cross-fusion outweighs the performance of the early and late fusion. This kind of FCN is also used in previous works on lidar-based road detection and path generation [20], [21]. Therefore, the cross-fusion technique worth to be investigated but needs to be modified for real-time execution.

III. Background

A camera-lidar fusion architecture called cross-fusion (CF) is proposed by Caltagirone et al. [19]. It is used as our baseline for a fusion architecture reduction. The implementation detail of the baseline architecture is provided by the authors allowing to be re-implemented[1]. To have an accurate re-implementation, every line on data preprocessing, architecture building, and network training and validation is strictly followed.

A. Lidar-camera cross fusion baseline model

The baseline model is structured around an encoder-decoder architecture for road detection [19]. To have context aggregation without losing resolution, a module called context module is used [22]. The module is placed in between the encoder and decoder sections. Convolution is the operation used in the encoder, and dilated convolution followed by dropout is used in the context module. The dilation factor used in the context module has exponential growth. The decoder has successive layers of deconvolution and convolution. The baseline model has two processing pipelines. This kind of pipelines have been utilized by previous works [20], [21]. The pipelines are used to process camera and lidar inputs. Each pipeline has twenty layers. A final softmax layer produces the output. The processing layers of one modality are fused with the corresponding layer of the other modality. The fusion operations are additions and weightings. A lidar layer is weighted by a learnable scalar. This parameter is called fusion weight of the lidar layer. Then it is added to the corresponding layer of the camera pipeline. The sum is used as input to the next camera layer. The same line of operation flows from camera layer to produce the next lidar layer. The weighting parameter is now called fusion weight of the camera layer.

B. Re-implementation detail

The baseline cross-fusion architecture is trained on the KITTI dataset (see section V-A) [23]. Camera images and 3D-lidar point clouds from the road detection are used. The lidar points are projected on the image plan to have a 2D representation of the depth information. The projection matrix P, the rectification matrix R and the translation matrix T are used to project a 3D lidar point x to a point y in the camera image. These matrices are provided on the KITTI benchmark suite.

$$y = P\,R\,T\,x \tag{1}$$

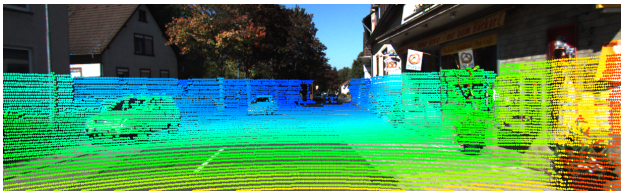
The depth in the X-axis, Y-axis and Z-axis of the projection forms three separate depth maps, similar to the three RGB channels of the camera image. Since these depth maps are sparse, up-sampling is done using the technique proposed by Premebidia et al. [24]. Fig. 1 illustrates the projection and up-sampling of 3D lidar points. Besides, zero-padding is used to have images of matching size, and no additional image pre-processing is used.

The fusion architecture is implemented in tensorflow. The network is trained using Adam optimization, and a polynomial decay is used as the learning rate. Detail of the re-implementation in alignment with the baseline cross-fusion architecture is given in table I.

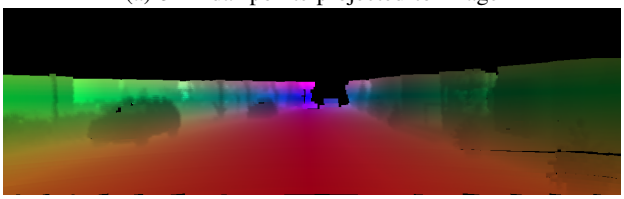
IV. Architecture reduction

The baseline cross-fusion architecture is reduced from the perspective of real-time execution in an instrumented test car, where the computational power is constrained. The reduction

[1]https://github.com/geletumn/cf_reduction



(a) 3D lidar points projected to image



(b) Up-sampled dense depth image

Fig. 1: Projection and up-sampling of 3D-lidar points

TABLE I: Baseline cross-fusion re-implementation detail

Re-implementation detail	
Inputs	RGB image lidar point cloud
Input size	384x1248
Preprocessing	Lidar projection Upsampling
Data split	Training: 239 frames Validation: 50 frames
Augmentation	Random rotation about the center in range of $[-20^o, 20^o]$
Optimization	Adam
Learning rate	Polynomial decay
Epoch	420
Batch size	1
Dataset	KITTI

is done considering fusion weights, generic encoder-decoder architecture, and context module.

A. Reduction in decoding

The baseline CF architecture has two pipelines for the two modalities. Each layer of one modality is fused with the corresponding layer of the other modality. The fusion weights are analyzed to see the strength of fusion in each layer. Since there is randomness in network initialization and training dataset split, the training of the baseline network is done multiple times. Five separate trainings having different seeds of the input are used. In all different seeds, the training-validation split ratio is maintained. There are twenty lidar and camera layers each. For each of these layers, the statistical summary of the fusion weights on the five runs is given in fig. 2.

The encoder has the first five layers. Layers six up to fourteen belongs to the context module. The last six layers are part of the decoder. For each layer, as can be seen from the statistical summary, there is a variation in the strength of fusion from layer to layer. From the box and whisker plot (see fig. 2), the median and the dispersion of the fusion weights in the decoding layers are small. This shows that there is no strong fusion in the decoding layers.

In a generic encoder-decoder architecture, reconstruction is done from the latent-space representation [25]–[27]. Once the latent representation is obtained, a single decoder will give the full resolution output. Since the fusion in the decoder section of the baseline is negligible regarding the other fusion weights, and a single decoder can be used for the reconstruction, a reduced fusion architecture is proposed (see fig. 3). This reduced architecture has a single unified decoder. It reconstructs from the joint latent representation of the two modalities. The fusion in the encoder and context module of the baseline is kept unchanged. This reduced architecture is named uni-decoder CF.

B. Reduction in the context module

Further reduction is proposed on the uni-decoder CF using the work of Yu and Kontun [22]. In their work, the context module was reduced depending on the image resolution. In their experiment on the KITTI dataset, a layer of the context module was removed because the vertical resolution of KITTI dataset images is small. Akin to this, layer 12 (L12) of the baseline is removed from the two processing pipelines. The resulting context module has eight layers. The architecture of this reduced context module is given in table II. This reduced fusion architecture is named Lite CF.

TABLE II: Architecture of the reduced context module

Context layer	1	2	3	4	5	6	7	8
Dilation	1	1	2	4	8	16	1	–
Receptive field	3×3	5×5	9×9	17×17	33×33	65×65	67×67	67×67
# feature maps	128	128	128	128	128	128	128	128
Filter size	3	3	3	3	3	3	3	1

V. Experimental results

The re-implementation in section III-B is evaluated, and its performance is compared against the baseline CF. The reduced architectures, uni-decoder CF and lite CF are also trained, and their prediction performance is compared with the baseline CF. Besides, the performance of the baseline, uni-decoder CF and lite CF are evaluated on brightness adjusted KITTI image. The KITTI metrics maximum F1 (MaxF), precision (PRE) and recall (REC) are used, and MaxF is the ranking metrics [28].

A. Dataset

In this work the KITTI dataset is used [23]. It is a dataset captured by driving around the city of Karlsruhe, Germany. It has camera images and laser scans among other measurements. For the task of road detection, the dataset provides 289 training images with their corresponding ground truth. These are collected mainly in good lighting conditions. It contains three different road categories: urban unmarked, urban marked and urban multiple marked lanes with 98, 95 and 96 samples respectively. These samples are split in to training and validation set in our experiment.

B. Re-implementation validation

The baseline is re-implemented as it is discussed in section III-B. Caltagirone et al. [19] have not reported the performance metrics value to be mean of multiple runs. However, there is randomness while training the network. Therefore, to have a

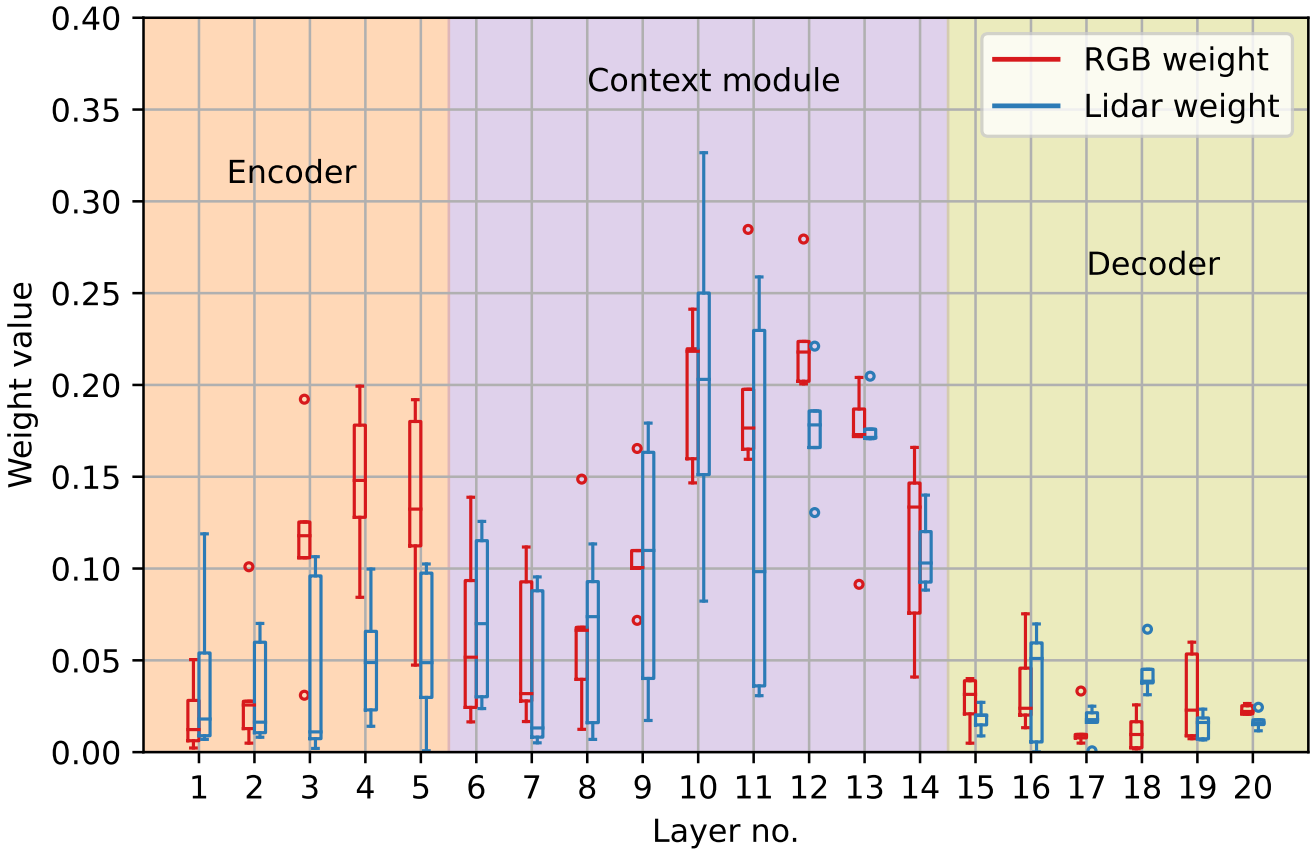


Fig. 2: Statistical summary of fusion weights

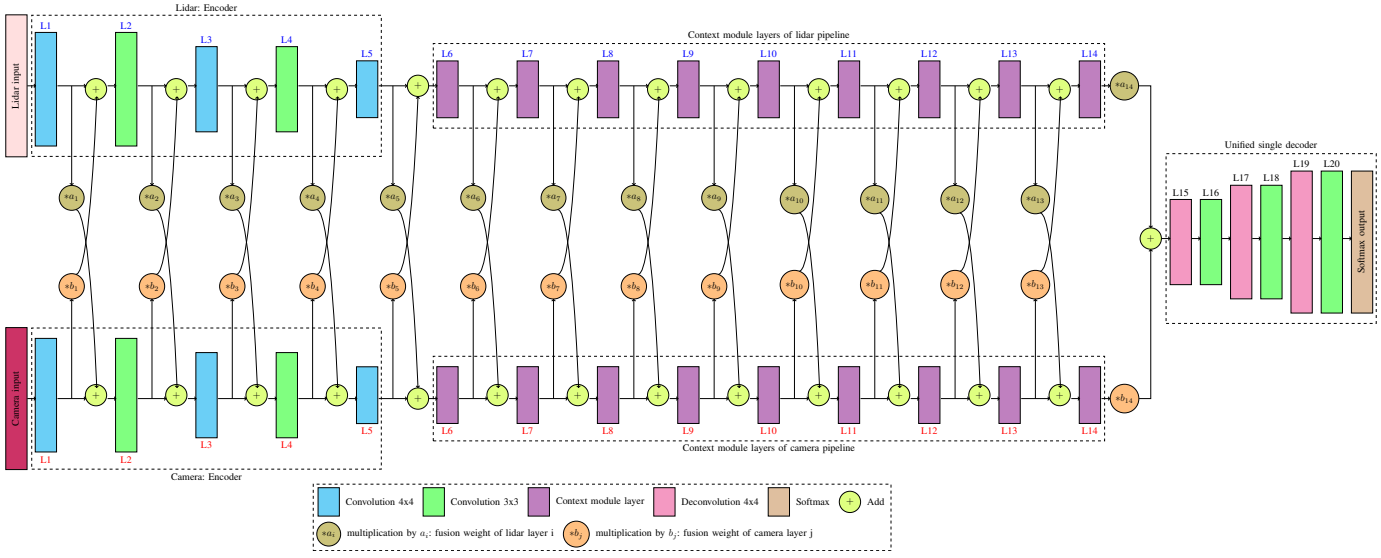


Fig. 3: Uni-decoder CF architecture

robust validation, five separate trainings are done on different seeds. The mean and the standard deviation (std), not the best, performance of the multiple trainings is reported (see table III). As can be seen from the table, the re-implementation performance is in alignment with the baseline result. Using the ranking metrics MaxF, the mean value on the re-implementation is close to the baseline. Furthermore, the re-implementation has a small statistical dispersion (small std

value). Therefore, in the absence of a public source code, the re-implementation is used as the baseline.

TABLE III: Re-implementation validation

	MaxF [%]	PRE [%]	REC [%]
Cross-fusion [19]	96.25	96.17	96.34
Re-implementation	96.52 (std=0.43)	96.70 (std=0.47)	96.34 (std=0.52)

C. Comparison of the reduced architectures

The proposed reduced architectures are evaluated, and compared with the validation of the baseline. As per the suggestion from the KITTI benchmark suite, 10-fold cross validation is used. The dataset is randomly splitted into ten folds. Each fold is made to have a proportional number of frames from each road category: urban unmarked, urban marked and urban multiple marked lanes. Then ten separate training-validation are made. Except for the data split, the training follows the procedure described in III-B. The mean and the std of the 10-fold cross validation is given in table IV.

TABLE IV: Reduced architectures validation

Fusion architecture	#model param.	MaxF [%]		PRE [%]		REC [%]	
		mean	std	mean	std	mean	std
Baseline	3,246,830	**96.25**	0.71	96.46	0.66	96.05	1.06
Uni-decoder CF	3,032,383	96.18	0.74	96.16	0.74	96.20	0.84
Lite CF	**2,737,213**	95.50	**0.52**	95.57	0.69	95.45	0.74

The validation result shows that the reduced architectures have comparable performance with the baseline while the model complexity (number of model parameters) is decreased. The performance of the reduced architectures, i.e. %MaxF, is only slightly decreased. As can be seen from the table, the uni-decoder CF has more than 6% reduction in number of model parameters. This reduction is further increased by the lite CF to more than 15% with small std on the MaxF value compared with the baseline. This may show that the baseline has learnt details of some scenarios that the dispersion on MaxF is slightly higher. Therefore, the lite CF gives comparable performance while the statistical dispersion is small.

For qualitative comparison of the proposed reduced architectures, some visuals of detection results are shown in fig. 4. As can be seen from the figure, the majority of the road segment is detected in both the baseline and the reduced architectures. However, they both make some false detections (false positive or false negative) at road edges and far end points. The false detection at far end points could be because of lack of rich representation in the camera image, and absence of laser scans. This is common to both the baseline and reduced architectures.

D. Evaluation on brightness adjusted KITTI dataset

Different fusion models can have different degree of robustness. To show this, a scenario is created by changing the brightness level of KITTI images. The original KITTI images are taken in a good sunny condition. However, when the image brightness level is changed, there are dark and foggy kind of images. This will impose difficulties on the fusion models to degrade their performance.

Evaluation is done to see the degree of robustness to change of brightness level on the baseline and the reduced architectures. The level of brightness is adjusted randomly by a little value on the KITTI images. The models that are trained on normal brightness images are evaluated on brightness adjusted ones. The 10-fold cross validation is used by changing the image brightness in the validation split. Table V gives the mean and std of the cross validation when the brightness is changed.

TABLE V: Performance comparison of baseline and reduced architecture while the image brightness is adjusted

Fusion architecture	MaxF [%]		PRE [%]		REC [%]	
	mean	std	mean	std	mean	std
Baseline	83.65	6.24	88.74	3.84	79.37	8.69
Uni-decoder CF	87.09	6.74	92.70	1.98	82.62	10.07
Lite CF	**87.42**	**3.68**	91.58	3.27	83.69	4.63

It is expected that the models' performance generally decrease. However, as can be seen from table V, the reduced architectures have higher MaxF value. Therefore, they are less affected than the baseline by the change in the level of brightness. Moreover, the lite CF model has small std value, which is statistically desirable. Similar result is also observed on the lite in section V-C. This shows that the lite CF is more robust than the baseline with less statistical dispersion. This robustness result of the lite CF together with the result in section V-C, i.e. more than 15% parameter reduction and comparable performance, makes lite CF a notable model.

Some visual results are given in Fig. 5. As can be seen from the figure, the baseline model is more affected by the change than the reduced models. When the image is darkened, the baseline model produces bad result. However, the reduced architectures are comparatively better. When the image is a foggy kind, all the models detected the major part of the road segment.

The robustness of the reduced architecture could be attributed to the removal of some layers. The baseline CF has twenty layers dedicated to process the camera input, which is rich in feature. Therefore, the model can be highly dominated by the camera input, and grabs details of the camera image features. As a result, a slight change in the brightness of camera images may result in a large loss of performance. This is also supported by the comparatively high values of std in the MaxF value (see table IV). However, the uni-decoder and lite CF have only fourteen and thirteen layers, respectively, dedicated to the camera input. Besides, decoding is done on a unified feature, i.e. there is no room to emphasize one modality over the other while decoding. As a result, every detail of the feature rich camera input may not be grabbed and processed. This can contribute to the fact that they are less affected when the brightness level in the camera images is changed slightly.

The MaxF value in table V has high value of std. Therefore, it is statistically dispersed. This can be due to the fact that the level of brightness is changed randomly. Since 10-fold cross

Fig. 4: Detection output of the baseline and reduced architectures. From left to right: baseline prediction, uni-decoder CF prediction, Lite CF prediction [Green: True positive, Red: False positive, Blue: False negative].

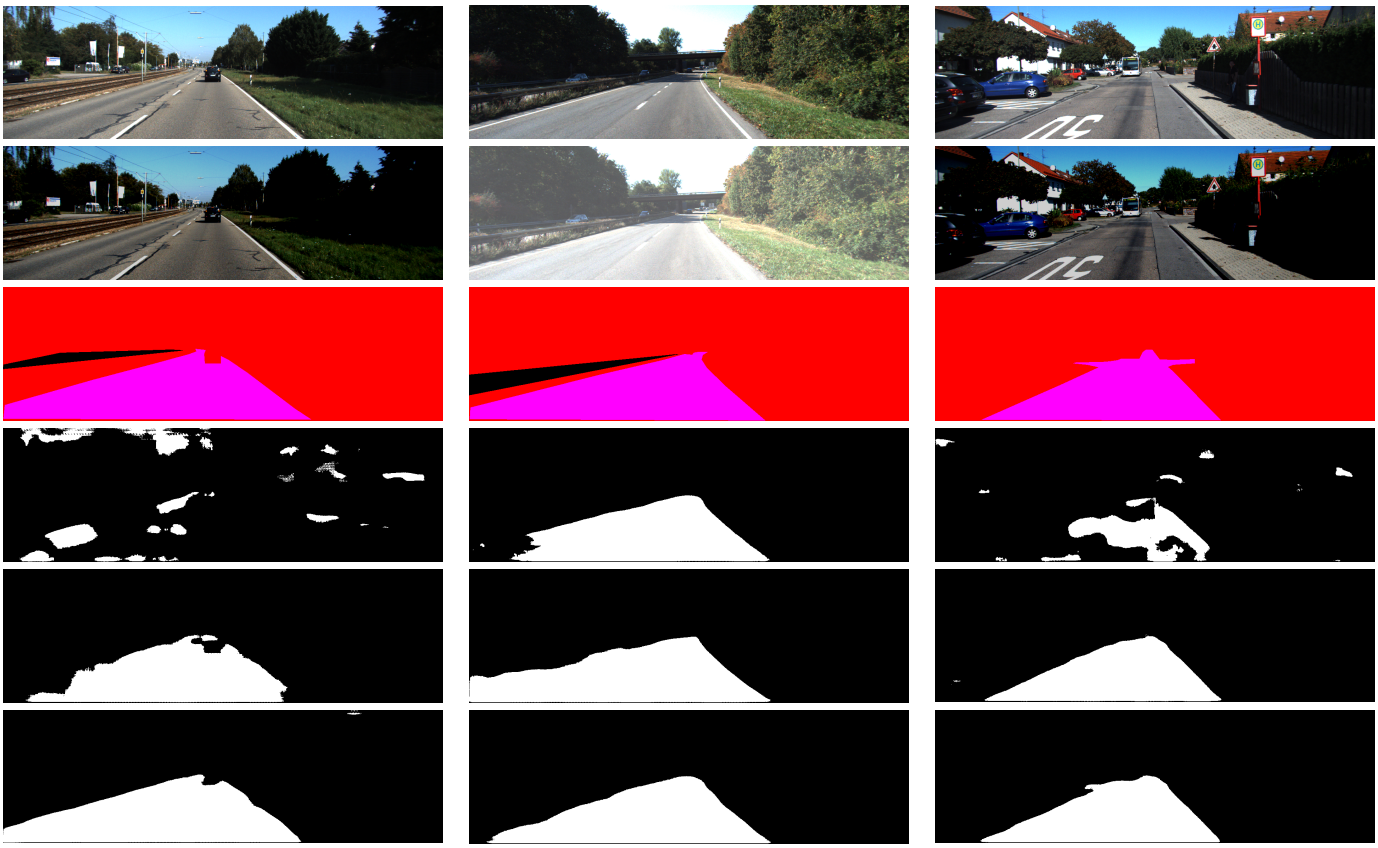


Fig. 5: Detection on brightness adjusted images. From top to bottom: KITTI image, brightness adjusted image, ground truth, baseline prediction, uni-decoder CF prediction, Lite CF prediction [Ground truth: pink-road, red-not road, black-invalid area; prediction: white-road, black-not road].

validation is used, the change in brightness in one fold could be totally different from the change in the other fold. Therefore, there is dispersion in performance from one validation fold to the other.

VI. Conclusion

In this paper, the problem of road detection for autonomous driving has been addressed. Thanks to their performance in image processing, deep neural networks have been considered. Since the main target is to embed the algorithms into industrial computers, there is a real need for architecture reduction. The paper has proposed two reduced architectures for a lidar-camera fusion. The reduced architectures have up to 15% reduction in model parameters compared with the baseline while maintaining comparable performance, only a maximum drop of 0.78% in mean value of MaxF. Moreover, the evaluation on brightness adjusted images has shown that the reduced architectures are more robust to brightness change than the baseline. Therefore, the proposed reduced architectures have practical advantage in providing robust and accurate performance at low computational cost in AVs, where computational power is constrained. Alternatively, they can be

used to boost performance by appending additional algorithm without inducing much computational complexity. Fusion architectures generally focus on building a high performance model. However, the model can be complex and demand high computational resources. Model architecture reduction should be pursued in order to get models that have comparable or better performance while having reduced size. This perspective can be applied to reduce a more complex and robust fusion architecture. It is also relevant to append additional algorithms that could improve performance, like evidential theory.

REFERENCES

[1] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A survey of autonomous driving: Common practices and emerging technologies," *IEEE access*, vol. 8, pp. 58 443–58 469, 2020.

[2] J. Van Brummelen, M. O'Brien, D. Gruyer, and H. Najjaran, "Autonomous vehicle perception: The technology of today and tomorrow," *Transportation research part C: emerging technologies*, vol. 89, pp. 384–406, 2018.

[3] D. Feng, C. Haase-Schütz, L. Rosenbaum, H. Hertlein, C. Glaeser, F. Timm, W. Wiesbeck, and K. Dietmayer, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 3, pp. 1341–1360, 2020.

[4] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *2012 IEEE conference on computer vision and pattern recognition*. IEEE, 2012, pp. 3354–3361.

[5] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen, "Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, no. 2, pp. 261–318, 2020.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[8] H. Laghmara, C. Cudel, J.-P. Lauffenburger, and M. Boumediene, "Evidential object association using heterogeneous sensor data," in *2018 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 1285–1292.

[9] H. Laghmara, T. Laurain, C. Cudel, and J.-P. Lauffenburger, "Heterogeneous sensor data fusion for multiple object association using belief functions," *Information Fusion*, vol. 57, pp. 44–58, 2020.

[10] G. Menghani, "Efficient deep learning: A survey on making deep learning models smaller, faster, and better," *arXiv preprint arXiv:2106.08962*, 2021.

[11] I. Konovalenko, P. Maruschak, H. Kozbur, J. Brezinová, J. Brezina, B. Nazarevich, and Y. Shkira, "Influence of uneven lighting on quantitative indicators of surface defects," *Machines*, vol. 10, no. 3, p. 194, 2022.

[12] T. Denœux, "Logistic regression, neural networks and dempster–shafer theory: A new perspective," *Knowledge-Based Systems*, vol. 176, pp. 54–67, 2019.

[13] E. Capellier, F. Davoine, V. Cherfaoui, and Y. Li, "Evidential deep learning for arbitrary lidar object classification in the context of autonomous driving," in *2019 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2019, pp. 1304–1311.

[14] A. Asvadi, L. Garrote, C. Premebida, P. Peixoto, and U. J. Nunes, "Multimodal vehicle detection: fusing 3d-lidar and color camera data," *Pattern Recognition Letters*, vol. 115, pp. 20–29, 2018.

[15] J. Redmon and A. Farhadi, "Yolo9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.

[16] J. Han, Y. Liao, J. Zhang, S. Wang, and S. Li, "Target fusion detection of lidar and camera based on the improved yolo algorithm," *Mathematics*, vol. 6, no. 10, p. 213, 2018.

[17] J. Kim, J. Kim, and J. Cho, "An advanced object classification strategy using yolo through camera and lidar sensor fusion," in *2019 13th International Conference on Signal Processing and Communication Systems (ICSPCS)*. IEEE, 2019, pp. 1–5.

[18] F. Nobis, M. Geisslinger, M. Weber, J. Betz, and M. Lienkamp, "A deep learning-based radar and camera sensor fusion architecture for object detection," in *2019 Sensor Data Fusion: Trends, Solutions, Applications (SDF)*. IEEE, 2019, pp. 1–7.

[19] L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, "Lidar–camera fusion for road detection using fully convolutional neural networks," *Robotics and Autonomous Systems*, vol. 111, pp. 125–131, 2019.

[20] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde, "Fast lidar-based road detection using fully convolutional neural networks," in *2017 ieee intelligent vehicles symposium (iv)*. IEEE, 2017, pp. 1019–1024.

[21] L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, "Lidar-based driving path generation using fully convolutional neural networks," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2017, pp. 1–6.

[22] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[23] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[24] C. Premebida, J. Carreira, J. Batista, and U. Nunes, "Pedestrian detection combining rgb and dense lidar data," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 4112–4117.

[25] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE transactions on pattern analysis and machine intelligence*, 2021.

[26] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[27] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[28] J. Fritsch, T. Kuehnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 1693–1700.