ISSN 2007-9737

Deep Learning Approaches to Bird's-Eye View Transformation and RGB-Depth Fusion in Autonomous Vehicles

Daniel A. Martinez-Barba[1], Luis M. Valentín-Coronado[1,3,*], Israel Becerra[2,3], Sebastían Salazar-Colores[1], Carlos Paredes-Orta[1,3]

[1] Centro de Investigaciones en Óptica A.C., Mexico

[2] Centro de Investigación en Matemáticas A.C., Mexico

[3] Consejo Nacional de Humanidades, Ciencias y Tecnologías, Mexico

{danielmtz, luismvc, sebastian.salazar, cparedes}@cio.mx, israelb@cimat.mx

Abstract. Autonomous vehicles depend on accurate and efficient environment representations such as semantically segmented Bird's Eye View (BEV) for path planning and decision-making to achieve safe navigation. Implementing deep learning techniques to generate front-view to bird's-eye view transformations with depth information and RGB images is often complex due to the absence of real-world BEV datasets for training. Additionally, model's performance is often affected by the semantic class imbalance of the BEV maps at the pixel level. On this study, we propose a sensor fusion block to integrate RGB and depth features to improve perspective transformation performance. Furthermore, we implement a layer-based data augmentation to address the class imbalance challenge. Experiments to demonstrate that the proposed sensor fusion block and the layer based data augmentation method improve perspective transformation performance on state of the art deep learning architectures.

Keywords. Sensor fusion, bird's eye view, perspective transform, deep learning, autonomous vehicles.

1 Introduction

Efficient and safe autonomous navigation systems for self-driving cars depend on the careful design of three key components: perception, planning, and control. The perception of the environment is particularly important, as it directly impacts the effectiveness of the following stages [8]. Nowadays, self-driving cars have access to a variety of information sources from diverse sensors, including cameras, which capture rich semantic information, LiDARs and radars, which provide accurate spatial information and velocity estimation respectively [7].

Thus, the environmental perception stage necessitates of algorithms capable of fusing data from different sources, creating a unified view to perform object detection [20], semantic segmentation [12, 16], tracking [18], predicting pedestrian intentions [15], among other computer vision tasks.

At present, deep learning (DL) based methods are frequently required to effectively implement the perception stage. These artificial intelligence (AI) algorithms enable the fusion of sensor data and the creation of environment representations, thereby enhancing decision-making accuracy and system efficiency [17].

In particular, vision-based bird's eye view (BEV) techniques are extensively used in applications like surveillance, urban planning, and autonomous navigation, providing a valuable tool for understanding large areas and optimizing

processes that benefit from a comprehensive perspective [8].

Li *et al.* [8] distinguishes three main approaches to perform front to BEV perspective transformation:

i) BEV Camera, that utilizes single or multiple camera setups to transform front perspective to BEV;

ii) BEV LiDAR, which uses point cloud inputs to perform detection or segmentation tasks, and

iii) BEV Fusion, including any fusion mechanisms that transform different input sensors into a unified BEV.

Monocular front camera to BEV perspective transformation is an interesting approach to the environmental perception stage, as it tries to capture both semantic and spatial information from a single camera.

Several strategies have been proposed for this task, such as using classic geometric operations to perform front-to-BEV transformations based on the camera's intrinsic and extrinsic parameters [5, 3]. However, recent advances in deep learning algorithms have surpassed these geometric transformation methods [8]. Notable examples of these advanced techniques include Generative Adversarial Networks (GANs) [19], Variational Autoencoders (VAEs) [12], and Vision Transformer-based architectures [14, 9]. However, researchers argue that sensor fusion would ideally improve the perception system's performance, yet fusing data from different modalities remains a challenging problem to solve [8]. In particular, BEV Fusion strategies focused on fusing camera and LiDAR features.

For example, Florea *et al.* [2] developed a methodology to fuse semantically segmented front perspective images into point clouds to perform 3D object detection. Liu *et al.* proposed the transformer-based architecture BEVFusion [11], described as an efficient and generic multi-task multi-sensor fusion framework, and demonstrated its capabilities to perform tasks such as BEV map segmentation and 3D object detection.

Another example of a transformer-based sensor fusion algorithm is the works of Gunn *et al.* [4]. These authors proposed a mechanism to fuse the specific image plane features of the image into the projected horizon of the LiDAR features. Regardless of BEV Camera or BEV Fusion, implementing DL-based front-to-BEV perspective transformations in autonomous driving presents various challenges, with one of the most notable being the accurate generation of ground truth for model training. While datasets like Kitty, Cityscapes, Argoverse, and NuScenes provide labeled sensor data for tasks such as object detection, lane detection, and semantic segmentation, they lack ground truth for perspective transformation [10]. As a result, some researchers have created or estimated their own BEV maps from sensor data.

For instance, Roddick and Cipolla [16] developed an algorithm to integrate and transform features (such as LiDAR data and object masks from annotated images) from NuScenes and Argoverse toolkits to generate 2D BEV representations for training their pyramid occupancy network. Conversely, Zhou *et al.* [19] employed paired (front-top) images from a video game to train a GAN-based perspective transformation model.

Another approach to obtaining objective ground truth for perspective transformation models is to use simulation tools like Gazebo, Unity, or Unreal Engine, where every aspect of the driving environment can be precisely controlled.

In particular, in this work, to ensure an unbiased comparison, we have built and used our synthetic dataset, which includes the ground truth from the generated top-view images. This strategy ensures a fair evaluation of any network's performance, as all are assessed using the same data type. The only difference in training arises from the application of data augmentation techniques or the inclusion of depth information encoded as an image (more detail in Section 2.1).

In this work, we explore the intricacies of transforming monocular front camera perspective images (RGB images) into semantically segmented bird eye's view representations. Additionally, we propose two methodologies to fuse RGB images and depth information. Furthermore, to address the issue of limited data, we have employed a data augmentation approach utilizing the CARLA open-source simulator for autonomous vehicles research [1]. The paper

ISSN 2007-9737

also discusses experiments on how deep learning models, data augmentation techniques, and depth integration methods interact and their impact.

2 Methodology

This section provides an overview of the description of the data set and the proposed deep learning-based methodology implemented.

2.1 Dataset

We created a dataset of 6,500 images from 65 scenes across four maps, which were used to train and test DL-based models. Traffic scenes encompass a wide range of scenarios, including challenging cases such as making unprotected left turns, navigating roundabouts, handling road curves, and approaching four-way stops, among others. These scenes also feature diverse scenery, including various architectural styles and lighting conditions. The scenes were generated along specific map routes, with a varying number of vehicles and pedestrians placed on the roads and walkways.

A wide range of 3D models were used, differing in vehicle type, size, and color, as well as pedestrian appearance, size, and pose. This diversity is intended to enhance the robustness of the proposed methodology. Each dataset sample consists of three images: two front-view images split into RGB and depth channels, and one top-view image in semantic format. All images are uniformly sized at 1024×1024 pixels. The top-view images are labeled according to five semantic classes: non-drivable space, drivable space, sidewalk, vehicle, and pedestrian. These specific classes were selected because we believe they provide a stable detection baseline for navigation. Fig. 1 illustrates a representative example of the dataset elements, including the color coding for the semantic classes.

After labeling the top-view images, a digital image processing step is systematically performed, generating the ground truth BEV maps, as illustrated in Fig. 2. The process begins with resizing the image, though this step can cause unwanted effects like vehicle deformation and loss of pixel information related to pedestrians. To mitigate these issues, an opening morphological operation (which involves erosion followed by dilation) is applied.

2.2 Data Augmentation Layers-based Approach

We have implemented the layer-based data augmentation (LbDA) reported in our previous work [13]. Unlike traditional data augmentation techniques that generate synthetic data through various transformations, the layer-based Data Augmentation (LbDA) selectively includes or excludes different object layers such as buildings, pedestrians, and vehicles from traffic scenes. This process results in a dataset composed of three distinct types of layers. The first type, named *layers-none*, contains only roads. The second type, *layers-all*, incorporates static objects such as buildings, poles, and fences. The third type, *traffic*, further adds dynamic objects like pedestrians and vehicles to the *layers-all* configuration. This method is designed to introduce new features that enhance the performance of front perspective to bird´s eye view mapping. Examples of the augmented samples are presented in Fig. 3.

2.2.1 Variational Encoder-Decoder

The Variational Encoder-Decoder (VED) is a DL architecture that combines the structure of autoencoders with the variational inference [12]. The VED architecture has two main components: the encoder and the decoder. The encoder maps input data into a probabilistic distribution over latent variables, typically characterized by a mean and variance. The latent space is a compressed, lower-dimensional representation where similar data points are clustered together.

In contrast, the decoder then reconstructs the original input data from samples drawn from the latent distribution, aiming to match the original input closely. Fig. 4a depicts the VED architecture. In this work, the VED implementation accepts a $256 \times 512 \times 3$ front camera image as input and generates a 200×196 semantic map as output.

(a) RGB image **(b)** Depth image **(c)** Semantic masks

Fig. 1. Dataset representative sample. (a) Front perspective view. (b) Depth image of perspective view. (c) Top segmented map and its class identifier and corresponding color

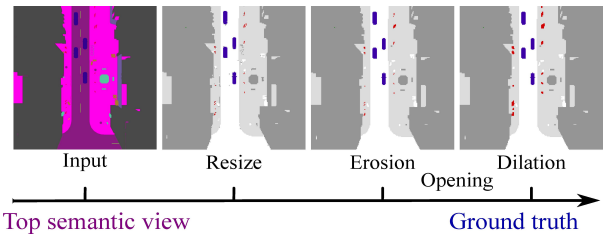


Fig. 2. Ground truth generation pipeline

(a) **(b)** **(c)**

Fig. 3. Layers-based data augmentation examples, where (a), (b), and (c) show the layers *traffic*, *layers-all*, and *layers-none*, respectively

2.3 Depth Integration

Transforming monocular front perspective RGB images to bird eye's view is one of the most common methods to create semantic BEV maps. However, as model's performance plateaus due to the architecture's limitations and dataset complexity, more advanced deep learning architectures or additional information is needed to further improve performance.

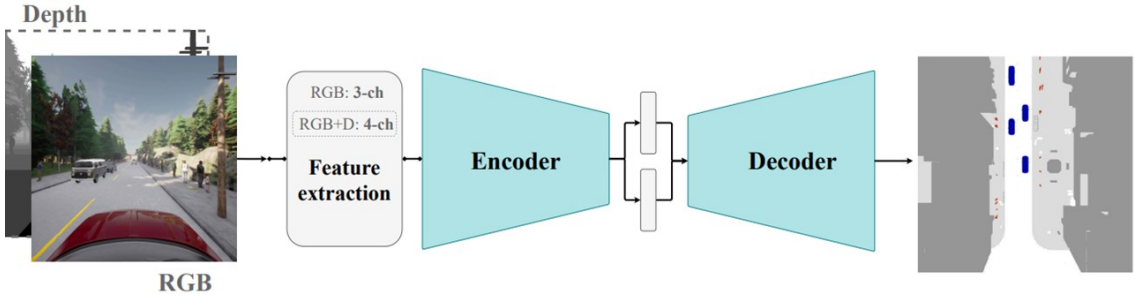
For this work, we analyze if depth integration substantially improves performance of perspective transformation. Then, we have integrated two methods for incorporating depth information into the deep-learning-based perspective transformation models.

2.3.1 Four-Channel Approach

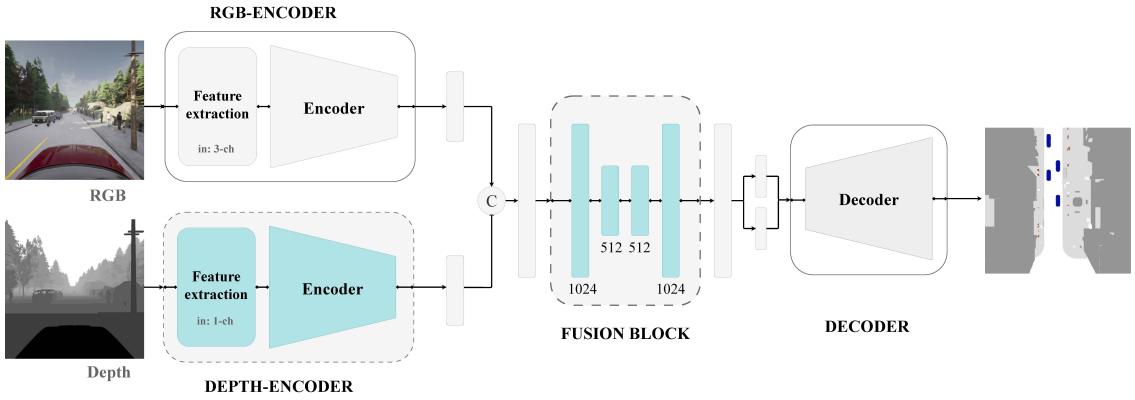
One of the simplest approaches to include depth images into monocular front perspective to BEV deep learning models is to concatenate the depth image to the RGB image in the channel dimension. Depth information encoded as an image allows seamless integration with the Convolutional Neural Network's (CNNs) ability to process image-like data. By representing depth as pixel values, CNNs may detect spatial patterns and relationships across the depth map, as has been extensively demonstrated with standard RGB images.

This capability enables the network to capture crucial 3D spatial features, such as relative positioning, which are essential for tasks like scene understanding, planning, or navigation. Moreover, depth data might complement the RGB data by providing geometric cues that improve the network's ability to disambiguate objects with similar visual appearances but different spatial locations. Encoding depth as an image also leverages the computational efficiency of CNNs, avoiding the need for specialized 3D data processing while enhancing performance in tasks requiring an understanding of 3D environments.

This approach allows processing an RGB + Depth image with almost no architecture modifications, i.e., it is only needed to change the input channels (three for RGB and four for RGBD) of the first layer of the feature extraction backbones of the models (Fig. 4a).



(a) Variational Encoder-Decoder's architecture



(b) VED-Fusion architecture

Fig. 4. (a) VED's architecture. (b) VED-Fusion architecture

With this approach, the RGB-Depth features are extracted by the backbone and then processed and combined through the encoder block of the model's architecture.

2.3.2 Sensor Fusion Block

To enhance the variational encoder-decoder's performance on perspective transformation, we propose a sensor fusion block.

The objective of this block is to generate an augmented latent vector using both RGB and depth images, with the aim of achieving superior performance in perspective transformation compared to approaches reliant solely on RGB images.

As depicted in Fig. 4b, RGB and depth features are extracted and then processed in different encoding heads. RGB and Depth encoder blocks only differ on the number of input channels of the feature extraction architecture (one and three input channels for the depth and color images, respectively). Encoding heads output each a 512 one-dimensional vector containing the representative features of the input images.

Color and depth encoded vectors are then concatenated and processed by the fusion block. The structure of the fusion block downsamples and upsamples the feature vector, allowing the color and depth perspective view features to interact with each other, enhancing the encoded representation.

The fusion block is integrated by four fully-connected layers. The input layer takes a 1024 1d-vector and further encodes into a 512 1d-vector. Subsequently, the second and third layers transform the features without changing the vector dimension, and finally, the output layer upsamples the vector into a 1024 1d-vector.

The first three fully-connect layers are followed by ReLU activation functions, then, dropout regularization to avoid overfitting. The output layer of the fusion block is followed by a 1d-batch normalization. The normalized 1024 1d-vector is then utilized to calculate the mean and log variance and compute the latent representation (z), which is then fed to the decoder block to produce the semantic BEV map.

2.4 Training

The dataset was divided into three subsets for the purpose of training and testing the models: training, validation, and test subsets. The training set comprises 6,500 images, with 90% designated for training and 10% reserved for validation. Then, 500 images, representing five scenes from an entirely different and previously unseen map, were set aside for testing. The models were trained for 50 epochs using the Adam optimizer [6], with the square root of the inverse frequency cross-entropy ($SqrtInvCE$) as the loss function, as defined in Eq. 1:

$$SqrtInvCE = \sqrt{-\sum_{c=1}^{C} N_c \log(P^c)}. \qquad (1)$$

3 Results and Discussion

The following experiments were conducted by training the following models with our dataset: Original VED architecture trained with RGB images (VED-RGB), a modified version of the VED's backbone input channels to extract features from RGB-Depth images (VED-RGBD), and the proposed VED modification to fuse RGB and depth images on separate encoding heads (VED-Fusion). The results were compared with the original Pyramid Occupancy Network (PON) architecture (PON-RGB) and a modified PON designed for handling four input channels (RGB-D) data (PON-RGBD).

3.1 Effects on Perspective Transformation of Classic and Layer-based Data Augmentation

Training deep learning models for complex problems is often affected by issues such as data scarcity either because data is limited, expensive or challenging to acquire. Acquiring ground truth to train front-to-bird-eye view perspective transformation algorithms is both a complicated and impractical task. Data augmentation is widely recognized as an effective technique to prevent model overfitting and overall improve generalization performance.

Table 1. Model performance using data augmentation. Mean intersection over union (mIoU) scores (non_aug: Non-augmented dataset, aug_cl: Classic data augmentation, LbDA: Layer-based data augmentation) * Previous work results on smaller and less complex dataset

Model	Augmentation		
	non_aug	aug_cl	LbDA
VED-RGB* [13]	0.5691	0.6330	**0.6356**
PON-RGB* [13]	0.6051	0.6487	**0.6586**
VED-RGB	0.2997	**0.4145**	0.4104
PON-RGB	0.2683	0.3297	**0.3926**

In this experiment, we analyze the effects of data augmentation on front to BEV perspective transformation models by contrasting classic methods against our reported layer-based data augmentation [13] (LbDA).

The LbDA method aims to improve semantic segmentation and perspective transform performance by augmenting map layers, as objects present or absent from the original scenes could improve the model's understanding of the driving environment. To assess the effectiveness of the LbDA technique, three distinct training approaches were utilized. The first, referred to as the "Non-augmented" method, used the original nine scenes with the *traffic* layer configuration. The second approach, known as the "classic" method, included traditional data augmentation techniques like Gaussian blur and vertical flipping within the *traffic* layer configuration. Finally, the

third approach involved the LbDA method, which applied the "layers-none", "layers-all", and "traffic" map layers as defined in Section 2.2.

Note that the classic and layer-based augmented datasets had three times more training samples.

In Table 1, two sets of results are shown. In contrast to our previous work [13], it is noteworthy that we have significantly augmented the dataset for this current work. This extension integrated more elaborated scenes including pedestrian crowds and more vehicles, which increased the the variability of the perspective transformation dataset, and therefore, its complexity.

Table 2. Sensor fusion performance (mIoU). Best performing model result is in bold. Best augmentation approach for each trained model is underlined (non_aug: Non-augmented dataset, aug_cl: Classic data augmentation, LbDA: Layer-based data augmentation)

	Sensor Fusion		
Model	non_aug	aug_cl	LbDA
VED-RGB	0.2997	**0.4145**	0.4104
PON-RGB	0.2683	0.3297	0.3926
VED-RGBD	0.4295	0.4032	0.4324
PON-RGBD	0.3843	0.3902	0.4733
VED-Fusion	**0.4602**	0.3761	**0.4921**

Nonetheless, classic and LbDA data augmentation approaches exhibit substantial improvements in mIoU metrics compared to the non-augmented approach, regardless of the model used. In particular, the Classic and LbDA models have improved around 11% for the VED and up to 13% for the PON with respect to the non-augmented approach. On the other hand, and opposed to the previous results, the PON model is more affected by the augmentation methods than VED. In summary, augmentation methods improved overall performance, classic exhibited a slight improvement over the LbDA on VED. However on PON, LbDA outperformed the classic data augmentation approach, at least on the validation split, composed of unseen scenes from the same map during the training process.

3.2 Effects on Perspective Transformation by Adding Depth Information and Using Fusion Block

To evaluate sensor fusion, three approaches were contrasted. The first approach is the *RGB* baseline, which does not include depth information. Then, the second approach is the *RGBD*, for this approach, the feature extraction backbone of the implemented models is modified so it can take a 4-channel image as input (RGB+Depth). Finally, in the third approach, the proposed *VED-Fusion* model is evaluated.

As presented in Table 2, the models that included depth information overall outperformed the *RGB* models on the mIoU score. The VED and PON models trained with the RGB-Depth images (*RGBD*) improved from 11% and up to 13% on the non-augmented approach respectively. Likewise, when trained with the data augmentation methods, the performance also improved, reaching up to 0.4733 mIoU for the PON model. Similar to the data augmentation experiment, our proposed LbDA method achieved better performance than the classic data augmentation on all models, with the exception of the VED model trained with the *RGB* approach.

Furtermore, the proposed *VED-Fusion* model achieved the best performance both in the non augmented approach and among all the trained models with data augmentation, achieving 0.4921 on the mIoU evaluation metric. Even though the *PON-RGBD* model trained with the four channel image feature extraction achieved the second highest score, the baseline and average score of the *VED-Fusion* model is higher, suggesting that the proposed fusion module better exploits the depth features, surpassing the performance of a newer and more advanced model as the PON.

3.3 Generalization of the Models on Unseen Scenarios

As mentioned in Section 2.4, a subset of 500 images from previously unseen scenarios were utilized to assess the models' generalization capabilities. These novel scenarios present increased complexity, requiring the models to predict perspective transformations based

Table 3. Performance results of all models in the generalization test. Best performing model is bolded. Best augmentation approach for each trained model is underlined (non_aug: Non-augmented dataset, aug_cl: Classic data augmentation, LbDA: Layer-based data augmentation)

	Generalization test		
Model	non_aug	aug_cl	LbDA
VED-RGB	0.2173	0.2610	0.2782
PON-RGB	0.2576	0.2613	0.2805
VED-RGBD	**0.2858**	**0.2855**	0.2896
PON-RGBD	0.2643	0.2518	0.2771
VED-Fusion	0.2725	0.2399	**0.3177**

on images derived from entirely different environments. The variations in lighting, weather, and architectural conditions within this new cities further compound the challenges, posing significant difficulties for accurate model predictions.

Tests have been conducted using the trained models on the color images (*VED-RGB* and *PON-RGB*), the four-channel feature extraction for depth integration (*VED-RGBD* and *PON-RGBD*), and the *VED-Fusion*, and under the three proposed data augmentation techniques. Tests were performed 30 times, and their mIoU metrics recorded. In Table 3, the best mIoU scores are shown. The 30 experimental runs were utilized to perform a statistical analysis (see Section 3.3.1).

From the results shown in Table 3, it may be observed that models trained using augmented methods tend to perform better. Specifically, the VED model performed slightly better, achieving the highest scores in the three augmentation methods (*Non-augmented*, *Classic data augmentation*, *layer-based data augmentation*). The implemented LbDA method performed better than classic augmentation in all five trained models, demonstrating that for this particular depth integration approaches, the layer-based data augmentation could help to further improve perspective transformation performance.

Moreover, the highest scores were also achieved by the models that were trained with depth information (between *RGBD* and *VED-Fusion*), suggesting that including depth information substantially improves the perspective transformation and semantic segmentation performance.

Table 4. Mean and standard deviation of all models in the generalization test. Best performing models are bolded (non_aug: Non-augmented dataset, aug_cl: Classic data augmentation, LbDA: Layer-based data augmentation)

	Generalization test statistics		
Model	non_aug	aug_cl	LbDA
VED-RGB	0.2069 (±0.004)	0.2485 (±0.004)	0.2705 (±0.003)
PON-RGB	0.2455 (±0.004)	0.2541 (±0.004)	0.2726 (±0.005)
VED-RGBD	**0.2733 (±0.005)**	**0.2762 (±0.004)**	0.2796 (±0.005)
PON-RGBD	0.2572 (±0.003)	0.2443 (±0.004)	0.2641 (±0.005)
VED-Fusion	0.2623 (±0.004)	0.2308 (±0.004)	**0.3071 (±0.006)**

Furthermore, the proposed fusion module outperformed the four-channel (RGB-Depth) feature extraction models, suggesting as well that encoding color and depth features and fusing them after is a better approach than including depth information as an additional image channel.

3.3.1 Statistical Analysis

To support the discussion of the results, a statistical analysis was conducted. Table 4 presents the mean and standard deviation of the experimental results. These statistical measures provide a clear summary of the data, where the mean represents the average performance across trials, and the standard deviation indicates the variability or consistency of the results.

The statistics obtained from the 30 experimental runs for each model show that the variability (standard deviation) on the mIoU metric has a maximum value of ±0.006 among all models, which supports that there is statistically significant difference between the models, as well as in the augmentation approaches.

4 Conclusion and Future Work

This study overviews a representation learning technique used in autonomous vehicles to convert front-view perspectives into bird's-eye view representations. We implemented and compared two state of the art deep learning architectures, the variational encoder-decoder (VED) and the pyramid occupancy network (PON) to explore the effects of data augmentation and sensor fusion (RGB and depth images) on perspective transformation performance. To test the perspective transformation we have extended our synthetic dataset by adding more maps, scenes, and depth information. We demonstrate that our proposed layer-based data augmentation method improves the perspective transformation performance on the implemented architectures.

Furthermore, we propose a sensor fusion block to enhance the VED architecture. The conducted experiments demonstrated that the proposed VED-Fusion architecture the most robust generalization capability among all the implemented and tested model, concluding that both utilizing our proposed layer-based data augmentation method and fusing front RGB and depth features improve the perspective transformation performance.

In future work, we plan to incorporate LiDAR's point cloud features and further improve our data augmentation technique, additionally we propose to explore estimating the distance between the ego vehicle and the objects of interest, also we plan to test our methodology on real-world data to further evaluate its efficiency.

Acknowledgment

Daniel A. Martinez Barba acknowledges the Consejo Nacional de Humanidades, Ciencia y Tecnología (CONAHCyT), México, for the support provided through the Postgraduate Scholarship.

References

1. **Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V. (2017).** Carla: An open urban driving simulator. Conference on robot learning, PMLR, pp. 1–16.
2. **Florea, H., Petrovai, A., Giosan, I., Oniga, F., Varga, R., Nedevschi, S. (2022).** Enhanced perception for autonomous driving using semantic and geometric data fusion. Sensors, Vol. 22, No. 13, pp. 5061.
3. **Garnett, N., Cohen, R., Pe'er, T., Lahav, R., Levi, D. (2019).** 3d-lanenet: end-to-end 3d multiple lane detection. Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2921–2930.
4. **Gunn, J., Lenyk, Z., Sharma, A., Donati, A., Buburuzan, A., Redford, J., Mueller, R. (2024).** Lift-attend-splat: Bird's-eye-view camera-lidar fusion using transformers. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops, pp. 4526–4536.
5. **Justs, D. J., Novickis, R., Ozols, K., Greitans, M. (2020).** Bird's-eye view image acquisition from simulated scenes using geometric inverse perspective mapping. 17th Biennial Baltic Electronics Conference (BEC), IEEE, pp. 1–6. DOI: 10.1109/BEC49624.2020.9277042.
6. **Kingma, D., Ba, J. (2015).** Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR), San Diega, CA, USA, pp. 1–26.
7. **Kocić, J., Jovičić, N., Drndarević, V. (2018).** Sensors and sensor fusion in autonomous vehicles. 2018 26th Telecommunications Forum (TELFOR), pp. 420–425. DOI: 10.1109/TELFOR.2018.8612054.
8. **Li, H., Sima, C., Dai, J., Wang, W., Lu, L., Wang, H., Zeng, J., Li, Z., Yang, J., Deng, H., et al. (2023).** Delving into the devils of bird's-eye-view perception: A review, evaluation and recipe. IEEE Transactions on Pattern Analysis and Machine Intelligence.

9. **Li, Z., Wang, W., Li, H., Xie, E., Sima, C., Lu, T., Qiao, Y., Dai, J. (2022).** Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. European conference on computer vision, Springer, pp. 1–18.

10. **Liu, M., Yurtsever, E., Zhou, X., Fossaert, J., Cui, Y., Zagar, B. L., Knoll, A. C. (2024).** A survey on autonomous driving datasets: Data statistic, annotation, and outlook. arXiv:2401.01454.

11. **Liu, Z., Tang, H., Amini, A., Yang, X., Mao, H., Rus, D. L., Han, S. (2023).** Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 2774–2781. DOI: 10.1109/ICRA48891.2023.10160968.

12. **Lu, C., van de Molengraft, M. J. G., Dubbelman, G. (2019).** Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks. IEEE Robotics and Automation Letters, Vol. 4, No. 2, pp. 445–452.

13. **Martinez-Barba, D. A., Valentin-Coronado, L. M., Becerra, I., Salzar-Colores, S., Paredes-Orta, C. (2024).** Front-to-bird's-eye-view transformation for autonomous vehicles: A class imbalance-based approach. **Mezura-Montes, E., Acosta-Mesa, H. G., Carrasco-Ochoa, J. A., Martínez-Trinidad, J. F., Olvera-López, J. A.**, editors, Pattern Recognition, Springer Nature Switzerland, Cham, pp. 166–176.

14. **Pan, C., He, Y., Peng, J., Zhang, Q., Sui, W., Zhang, Z. (2023).** Baeformer: Bi-directional and early interaction transformers for bird's eye view semantic segmentation. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9590–9599.

15. **Ranga, A., Giruzzi, F., Bhanushali, J., Wirbel, E., Pérez, P., Vu, T.-H., Perotton, X. (2020).** Vrunet: Multi-task learning model for intent prediction of vulnerable road users. Electronic Imaging, Vol. 32, No. 16, pp. 109–1–109–1. DOI: 10.2352/ISSN.2470-1173.2020.16.AVM-109.

16. **Roddick, T., Cipolla, R. (2020).** Predicting semantic map representations from images using pyramid occupancy networks. in 2020 ieee. CVF Conference on Computer Vision and Pattern Recognition, CVPR, pp. 13–19.

17. **Saval-Calvo, M., Medina-Valdés, L., Castillo-Secilla, J. M., Cuenca-Asensi, S., Martínez-Álvarez, A., Villagrá, J. (2017).** A review of the bayesian occupancy filter. Sensors, Vol. 17, No. 2, pp. 344.

18. **Yang, Y., Deng, Y., Zhang, J., Nie, J., Zha, Z.-J. (2024).** Bevtrack: A simple and strong baseline for 3d single object tracking in bird's-eye view.

19. **Zhou, T., He, D., Lee, C.-H. (2020).** Pixel-level bird view image generation from front view by using a generative adversarial network. 2020 6th International Conference on Control, Automation and Robotics (ICCAR), IEEE, pp. 683–689. DOI: 10.1109/ICCAR49639.2020.9107991.

20. **Zou, J., Zhu, Z., Huang, J., Yang, T., Huang, G., Wang, X. (2023).** Hft: Lifting perspective representations via hybrid feature transformation for bev perception. 2023 IEEE International Conference on Robotics and Automation (ICRA), pp. 7046–7053. DOI: 10.1109/ICRA48891.2023.10161214.

Article received on 15/06/2024; accepted on 20/10/2024.
Corresponding author is Luis M. Valentín-Coronado.