Single Neural Network Sensor Fusion for Automotive Application

Marcelo Eduardo Pederiva , José Mario De Martino and Alessandro Zimmer
{m122580@dac.unicamp.br, martino@unicamp.br, Alessandro.Zimmer@thi.de}

Departamento de Engenharia de Computação e Automação (DCA)
Faculdade de Engenharia Elétrica e de Computação (FEEC)
Universidade Estadual de Campinas (Unicamp)
Campinas, SP, Brasil

Abstract – Autonomous vehicles are becoming a reality in ground transportation. Computational advancement has enabled powerful methods to sense, map, locate, and process large amounts of data required to drive on urban streets safely. The fusion of multiple sensors allows for building accurate world models to improve autonomous vehicles' navigation and behavior. Among the current techniques, the fusion of LIDAR, RADAR, and Camera data have shown significant improvement in perception tasks. Current methods use parallel networks to explore each sensor separately. Despite its significant accuracy, its response time and high demand for computational resources are still limitations for a real-world self-driving application. Fusing these sensors using a single network is still an open question and a promising candidate to avoid these problems. The paper presents a Ph.D. project under development. It presents a preliminary approach for early sensor fusion and its performance with one sensor to detect 3D objects.

Keywords – Object Detection, Sensor Fusion, Machine Learning, Multi-task Learning, Autonomous Vehicles

1. Introduction

Autonomous vehicles have been the target of great interest in universities, research centers, and industry. With the advance of computer technology and computational techniques, autonomous cars' implementation became increasingly viable. However, implementing autonomous vehicles on urban streets requires a thorough perception of the environment, including detecting objects and their movements. These tasks require exteroceptive sensors to measure the car's surroundings. Sensors in this category include Cameras, Radio Detection and Ranging sensors (RADAR), and Light Detection and Ranging sensors (LIDAR).

Currently, LIDARs are widely used to detect objects around the vehicle. LIDAR data can be used to precisely estimate an object's geometry [11]. However, this type of sensor presents significant limitations in estimating motion. In contrast, RADARs allow robust motion estimation, providing accurate object velocity and direction measurements. Additionally, RADARs support reliable detection despite adverse weather conditions [12]. LIDARs and RADARs have significant limitations concerning recognizing objects despite their reliable performance in the situations mentioned. Object recognition using Cameras is an improving research area [1]. The state-of-art object recognition approaches using cameras deliver accurate real-time recognition (under 0.04 seconds) of several objects simultaneously observed in an image [2].

In general, the environment perception has been performed using a combination of two sensors: LIDAR-Camera, LIDAR-RADAR, or RADAR-Camera. Methods based on LIDAR-Camera have shown convincing results concerning visual detection, distance, and geometry estimation of objects [3]. However, the methods are not adequate to estimate objects' velocities [13]. Velocity estimation is better tackled by LIDAR-RADAR fusion methods [9]. However, the absence of cameras in LIDAR-RADAR approaches precludes objects' visual identification, impacting autonomous decision-making. Finally, the RADAR-Camera detection shows gains in performance for detecting objects in low light and rainy/cloudy weather [4]. Although RADARs are reliable all-weather sensors, they can not provide a dense environment sampling as LIDARs. Consequently, the combination RADAR-Camera does not support a high-quality geometry estimation of the detected objects.

At the end of 2020, the first LIDAR-RADAR-Camera fusion Deep Learning-based for 3D object detection in a real-world scene was proposed. Based on Frustum PointNet (F-PointNet) [8], the method uses the Fast R-CNN object detector to estimate a Region of Interest (RoI) of the camera view. This output is combined with the LIDAR measurements to estimate and classify the 3D object model. Simultaneously, the RADAR sensor provides the detections' velocity estimation with a different neural network. As a result, the model achieved high accuracy and small velocity error compared to the current methods that used only a two sensors fusion [10].

The mentioned approaches and main 3D object detection models use a Late Fusion of the sensors. Based on parallel networks, each sensor passes through a different neural network, and in the end or in the middle, the results are combined. This method provides a precise result. However, it requires a high demand for computing resources, including footprint and energy consumption. On the other hand, the Early Fusion fuses the sensors' information before the network, implementing a single architecture for the prediction. This process has low computation requirements and a low memory budget. Nevertheless, the learning process of mixed features of multiple different sensors increases the prediction challenge.

Although Early Fusion approaches for 3D object detection are still challenging, their low computational demand is of great relevance for application in an autonomous vehicle with limited resources. Therefore, our project aims to reduce computational demand with a new competitive fusion approach based on a single Neural Network. This network will combine and analyze the information acquired by LIDAR, RADAR, and Camera sensors to detect vehicles on the street.

The remaining paper is organized as follows. We first review the related work in section 2. Then in section 3, we present some experiments based on a known dataset. Next, we show some preliminary results and analysis in section 4. Finally, in section 5, we conclude the paper and present the remaining approaches in the project.

2. Related work

The main challenge for implementing an Early Fusion is effectively merging data from different types of sensors. For this, it is necessary to represent all the data in a single reference.

Image-based Object Detection research has been fast developed in recent years. Furthermore, current methods present a high accuracy and fast response in detecting multiple objects presented in the same image. In this way, this project aims to use the Camera sensor as a reference for other sensors and build our model based on the state-of-art 2D Object Detection models.

In the 2D Object Detection field, among the state-of-art models, the "You Only Look Once", namely YOLO, has been standing out in 2D object detection models in the last years. Its ability to recognize objects' bounding boxes and classify them quickly makes it an excellent candidate for many tasks. Consequently, variations of YOLO have surged in different fields, such as Object Detection, Object Tracking, Image Segmentation, and Landmark Detection. Due to this versatility and its remarkable performance in different fields, the YOLO approach is a promising initial candidate to use as inspiration to develop our 3D detection model.

Our approach converts the labels of the cars' position into a three-dimensional grid, a spatial representation of the environment. The grid is divided into a Sx×Sy×Sz grid, where Sx, Sy, and Sz represents the division in each X, Y, and Z axis, respectively. As a result, our model uses $Sx = 13$, $Sy = 5$, and $Sz = 13$. The Figure 1 shows the grid representation, where each grid cell stores an 9 length array prediction: $class, P, x, y, z, w, h, l, \theta$. The first value represents the object class, followed by a confidence of the grid cell estimation, then the six values representing the 3D bounding box (center of the object and its dimensions (width, height, length)), and the last, the rotation angle in Y axis of the object.

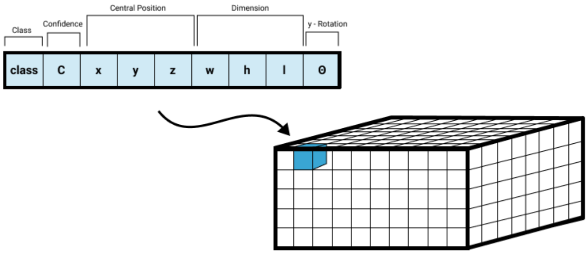

Figure 1. Three-dimensional grid of features.

Based on Supervised Learning, the network was trained to return a similar three-dimensional grid as provided by the label. In other words, the network output a tensor of predictions accommodated in a $13 \times 5 \times 13 \times 9$ tensor.

For the input of our network, the point cloud sensors (LIDAR and RADAR) are converted to the camera reference and are concatenated into an input tensor $width \times height \times 8$, where 8 represents the channels of each sensor: R, G, B, X, Y, Z, Vx, Vz. Furthermore, each channel is normalized individually.

2.1. Network

The proposed network is based on a Multi-Task Learning approach with hard parameter sharing. In other words, the architecture presents a sequence of convolutional layers and in the end, the last layer is branched into five output tasks.

The network combines two types of convolutional layers, 2D and 3D. First, it is used a known backbone as a 2D feature extractor. We choose the ResNet50, aiming for a combination of good performance in a fast response. It receives the input tensor (W×H×channels) and outputs a 15×15×2048 shape. This output pass trough one convolutional layer with 625 filters $kernel\_size =$

1, $strides = 1$, and leakyReLU activation function. Next, to introduce one dimension to the architecture, it is passed by a reshape layer (25×17×25×17). This new shape passes by a sequence of 3D convolutional and residual layers, representing a ResNet approach with 3D convolutions. Finally, the output is branched by five 3D convolutional layers. Each branch will predict each characteristic (class, confidence, position, dimension, and rotation angle) separately and be concatenated into a Sx×Sy×Sz×9 tensor of predictions.

2.2. Loss

$$loss = loss_{conf} + loss_{class} + loss_{box} \quad (1)$$

To calculate the error, we consider the L2 loss of all prediction features. The loss equation (Equation 1) is divided by 3 losses: Confidence loss ($loss_{conf}$), Class loss ($loss_{class}$), Box loss ($loss_{box}$).

The Confidence loss is represented by the reliance on each model prediction. It is represented by correct cell prediction ($loss_{obj}$) and incorrect cell prediction ($loss_{noobj}$). The Classification loss is represented by the error in the prediction of the object's classification ($loss_{class}$). Finally, the Box loss is defined by the errors of the Intersection over Union score ($loss_{IoU}$) and the y-axis rotational estimation ($loss_{\theta}$). Furthermore, each error has a weight parameter (λ) to balance the relevance of each feature in the total loss value. Currently, the weight values is defined as follows: $\lambda_{obj} = 20$; $\lambda_{noobj} = 1$; $\lambda_{class} = 1$; $\lambda_{IoU} = 10$; $\lambda_{\theta} = 10$.

3. Experiments

Our project starts exploring its performance using only the Camera as input. With a good candidate for 3D object detections, the model will be tested with LIDAR data. Finally, the RADAR sensor will be linked to the tensor to predict objects' velocity at a later phase.

Our model performance has been tested in the KITTI dataset. However, as the KITTI test dataset is closed and has a limited number of submissions, for experimental studies, we divided the open content (7581 images) into train and test sets for our model. The training step was done with 80% to train and 10% for validation. The last 10% was used for the testing step. The model was trained in a GPU: RTX2080ti and CPU: i7 9700KF with 200 epochs and used the Adam optimizer with a learning rate = 0.003.

4. Preliminary Results

In the Object Detection field, the Intersection over Union (IoU) score is a particular evaluation metric used to represent the matching percent between the predicted bounding box and the ground truth. Then, to evaluate the mean Average Precision (mAP), we considered a correct detection if the prediction presents an $IoU > 0.7$ and an object recognized if it presents an $IoU > 0.1$.

Table 1. Model's Performance.

	mAP	Mean IoU	Max IoU	Average Recognition
3D	25.34 %	0.3636	0.9119	79.81 %
2D	25.38 %	0.3667	0.9537	79.02 %

As a result, our proposed model using only the camera sensor achieve great precision and recognized most of the vehicles presented in the scene. In Table 1, the results considering the 3D and 2D (Bird Eye View) IoU scores are shown. The *mAP*, *mean IoU*, and *Max IoU* represent the mean Average Precision, the mean Intersection over Union, and the maximum Intersection over Union, respectively.

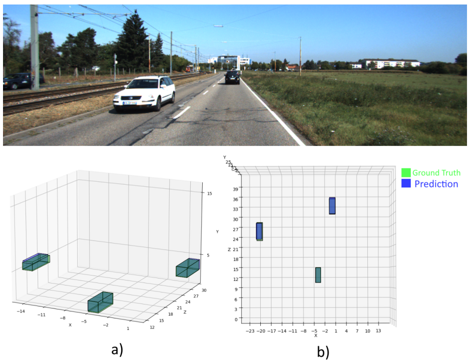

Figure 2. Example of model detection. The top image is the camera input. On a), we have an overview and on b), a bird-eye-view perspective.

The Figure 2 shows the model performance detecting 3 vehicles correctly. There are two vehicles fully visible and one partially occluded in this view. The ground truth is represented by the green box and the prediction with a blue box.

Table 2. Monocular Detection on KITTI dataset. Methods with * trained/tested with a small dataset.

Methods	mAP$_{3D}$	mAP$_{2D}$
Ground-Aware [6]	14.94%	20.29%
MonoEF [15]	15.62%	22.00%
MonoFlex [14]	15.30%	21.62%
GUPNet [7]	16.80%	23.23%
MonoCon [5]	17.64%	24.07%
Ours*	**25.34%**	**25.38%**

The state-of-art models presented in Table 2 were trained using all training set (open content) and were tested in the closed test set. The KITTI test set presents similar images, in the same environments, from the open content. Although our model was trained and tested in a small dataset, its results showed a good candidate for 3D object detection.

5. Conclusions and Remaining Work

This study presents a beginning approach to fuse sensors' data. It shows the use of a single network to predict the 3D bounding box of objects for autonomous vehicles. The initial results, using a single sensor, showed itself a good candidate for 3D object detection. Using a small dataset to train and test, the model showed a competitive result against state-of-art models.

For the remaining work, LIDAR data will be implemented in the input of the network aiming to enhance the prediction precision. Next, the RADAR data will be implemented to estimate the velocity of the objects.

The detection model proposed by this research has applications that are not limited to autonomous cars. With technical improvement, the model can be used in different areas that need an autonomous perception of the environment.

Acknowledgment

The authors acknowledge the funding received from The National Council for Scientific and Technological Development (CNPq 141061/2021-9).

References

[1] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. YOLOv4: Optimal Speed and Accuracy of Object Detection. http://arxiv.org/abs/2004.10934, 2020.

[2] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection, 2020.

[3] Luca Caltagirone, Mauro Bellone, Lennart Svensson, and Mattias Wahde. LIDAR–camera fusion for road detection using fully convolutional neural networks. *Robotics and Autonomous Systems*, 111:125–131, 2019.

[4] Kamil Kowol, Matthias Rottmann, Stefan Bracke, and Hanno Gottschalk. YOdar: Uncertainty-based Sensor Fusion for Vehicle Detection with Camera and Radar Sensors. http://arxiv.org/abs/2010.03320, 2020.

[5] Xianpeng Liu, Nan Xue, and Tianfu Wu. Learning auxiliary monocular contexts helps monocular 3d object detection, 2021.

[6] Yuxuan Liu, Yuan Yixuan, and Ming Liu. Ground-aware monocular 3d object detection for autonomous driving. *IEEE Robotics and Automation Letters*, 6(2):919–926, 2021.

[7] Yan Lu, Xinzhu Ma, Lei Yang, Tianzhu Zhang, Yating Liu, Qi Chu, Junjie Yan, and Wanli Ouyang. Geometry uncertainty projection network for monocular 3d object detection, 2021.

[8] Charles R. Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas. Frustum PointNets for 3D Object Detection from RGB-D Data. https://arxiv.org/abs/1711.08488, 2018.

[9] Meet Shah, Zhiling Huang, Ankit Laddha, Matthew Langford, Blake Barber, Sidney Zhang, Carlos Vallespi-Gonzalez, and Raquel Urtasun. LiRaNet: End-to-End Trajectory Prediction using Spatio-Temporal Radar Fusion. https://arxiv.org/abs/2010.00731, 2020.

[10] L. Wang, T. Chen, C. Anklam, and B. Goldluecke. High dimensional frustum pointnet for 3d object detection from camera, lidar, and radar. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, pages 1621–1628, 2020.

[11] Ruisheng Wang. 3d building modeling using images and lidar: a review. *International Journal of Image and Data Fusion*, 4(4):273–292, 2013.

[12] Bin Yang, Runsheng Guo, Ming Liang, Sergio Casas, and Raquel Urtasun. RadarNet: Exploiting Radar for Robust Perception of Dynamic Objects. https://arxiv.org/abs/2007.14366, 2020.

[13] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Center-based 3d object detection and tracking. https://arxiv.org/abs/2006.11275, 2021.

[14] Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3289–3298, June 2021.

[15] Yunsong Zhou, Yuan He, Hongzi Zhu, Cheng Wang, Hongyang Li, and Qinhong Jiang. Monocular 3d object detection: An extrinsic parameter free approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7556–7566, June 2021.