Deep Sensor Fusion for 3D Bounding Box Estimation and Recognition of Objects

Ayush Gupta
Department of Electrical Engineering
Stanford University
ayushg@stanford.edu

Malavika Bindhi
Department of Electrical Engineering
Stanford University
mbindhi@stanford.edu

Abstract

The objective of this work is to use deep learning for an early fusion of camera and lidar information to perform 3D bounding box estimation and object identification through a PointFusion architecture. Using a PointNet model to directly consume the point clouds and a standard CNN to process the corresponding camera image, the obtained point and image features are learnt to be combined optimally by a fusion network in order to perform 3D object detection with accuracy. The model is evaluated on KITTI 3D Object Detection dataset that features real traffic driving scenes captured through a camera-lidar setup. Based on the results obtained, our model achieves a state-of-the-art accuracy with a bounding box average IOU of 0.71 *and a classification accuracy of* 95.62%.

1. Introduction

The world is witnessing a cusp in technical revolution, where autonomous vehicles (AVs) are much closer to becoming a reality than ever before. However, one major concern before the autonomous vehicle technology can be put to use is its reliability and safety, a task that highly depends on the detection of the vehicle's surrounding. While sensing of the environment is predominantly performed by imaging sensors, LIDAR sensors are mostly entrusted with perception and localization [SH16]. However, both types of sensors carry certain inadequacies which can be taken care of by fusing their individual capabilities into a robust sensing and perception mechanism.

In this context, we aspire to integrate the important information carried by an image from a camera and a 3D point cloud from a LIDAR. Ideally, an AV should be able to identify and locate various signs and obstacles with precision to avoid collisions and damages. However, the main challenge here lies in 3D object detection that crucially impacts the AV's performance. While Convolutional Neural Networks have significantly advanced the state of 2D detection, 3D object detection problem still remains an open challenge. Thus, in this project, we aim to implement a deep learning approach that can combine image and depth information to give a 6 DoF pose and the 3D bounding box dimensions, along with identification of all objects of interest in the scene.

The report is organized as follows. After conducting a brief review of the recent works associated with object pose estimation in Section 2, we discuss the PointFusion model in Section 3. Further, the specifications about the dataset with the implementation and experimentation details are discussed in Section 4. The obtained performance results are also discussed. Finally, conclusions are drawn in Section 5 and target for future work is explained in Section 6.

2. Related Work

Seif et al. looks into the details of some of the necessary technologies required for autonomous driving [SH16]. Informing on all aspects, the need for a robust and precise environment detection system is predominantly emphasized. Specific to this problem, some works have used geometry-based approaches, relying either on strong category shape priors or ground-truth object CAD models [AME$^+$14] [FTVG06] [ZDY$^+$14]. This approach makes it difficult for them to scale to larger datasets. Another work leverages on depth information per frame and multi-view scene information to obtain accurate 3D object proposals [PLJ$^+$17].

On the other hand, some new studies have proposed to directly tackle the 3D object detection problem in discretized 3D spaces. However, these methods are often prohibitively expensive because of the discretized volumetric representation, which makes the computation time and space expensive [SX14] [SX16].

Insightful of the potential of deep learning approaches, some researchers have used deep learning approaches to perform 3d box regression from images or depth data. In one such work [XCLS17], Xiang et al. jointly learn a viewpoint dependent detector and a pose estimator by clustering 3D voxel patterns learned from object models. Another example of such approach that uses preprocessing of the 3D

input before using a deep learning model is [Li17], where Li et al. focus on detecting vehicles and process the voxelized input with a 3D fully convolutional network.

One of the important works that explore fusing image and lidar data is a multi-view 3D detection model, MV3D [CMW+17], that generates object detection proposals in the top-down lidar view. It then projects the proposals to the front-lidar and image views, fusing all the corresponding features to do oriented box regression. However, this model assumes that all objects lie on the same spatial plane and can be localized solely from a top-down view of the point cloud. Another approach by Asvadi et al. performs multimodal vehicle detection using LiDAR and camera data [AGP+17]. The point cloud from LiDAR is transformed to form a depth dense map and reflection dense map. Then 3 separate models are trained and vehicles are detected by using the combined predictions. However, such a method highly relies on simplifying assumptions and takes decisions independent of the information from other sensors.

Overcoming such limitations, the PointFusion [XAJ17] network has no scene or object-specific limitations, as well as no limitation on any kind and number of depth sensors. Using a PointNet [QSMG17] architecture to directly process the point clouds prevents any lossy or simplifying preprocessing of data. As a result, our generic method estimates both the 6-DoF pose and spatial dimensions of an object without object category knowledge or CAD models.

3. PointFusion

This section explains the basic architecture of PointFusion which performs 3D bounding box regression by retrieving information from a corresponding camera image generated by the camera sensor and a raw point cloud prepared by the LiDAR sensor. This model has three main components: 1) a PointNet [QSMG17] network's variant for processing raw point cloud through deep learning while well respecting the permutation invariance of points in the input; 2) a CNN that extracts image appearance and classification features; 3) a fusion network that combines both features to perform the 3D bounding box estimation and recognition on the input. Figure 1 gives an overview of this model.

As can be understood from Figure 1, the PointNet variant is responsible for processing the raw point cloud data, and the ResNet-50 model extracts visual features from an input image. The obtained point and image features are then fed in to a fusion network that processes its input using several layers and outputs a 3D bounding box prediction and object classification. The following subsections offer more insight into these comprising parts of our PointFusion model.

3.1. PointNet

PointNet is a unified architecture that directly takes point clouds as input and effectively learns a set of optimization functions/criteria that select interesting or informative points of the point cloud and encode the reason for their selection. The basic architecture of PointNet processes each point identically and independently. Key to the approach is the use of a single symmetric function, max pooling, which respects the permutation invariance of the points. Also, since the input is just a point, it is easy to apply rigid or affine transformations as each point transforms independently, making it easy to add a data-dependent spatial transformer. Figure 2 presents a symbolic representation of the architecture of PointNet which gives us the required point features.

3.2. ResNet

ResNet-50 [HZRS15] pretrained on ImageNet is available as part of the Keras Applications and can be directly imported. Using a transfer learning approach, we employed this trained model to give useful image features for the images in consideration. Following a general pre-processing of the dimension space to match the requirements of the ResNet-50 model, the output from the final average pooling layer gives us our image features.

3.3. Global Fusion

As shown in Figure 1, the last comprising layer in the PointFusion architecture is the fusion layer which processes the image and point cloud features and directly regresses the 3D locations of the eight corners of the target bounding box along with the learned classification result. Consisting of 3 hidden layers having 512, 128, and 128 units, respectively, the fusion layer gives the box-corner locations (in the point cloud coordinates) and the classification prediction as two outputs.

4. Experiments and Results

Selection of various choices for training the layer, as well as for the pre-processing and processing is an intuitive task supported by empirical observations. This section details the experimentation specifications along with the obtained results.

4.1. Data

The presented model is trained on the KITTI 3D Object Detection dataset, which contains annotated real-world traffic situations ranging from freeways to inner-city scenes captured from a VW station wagon.The dataset in total derives from 6 hours recording of traffic scenarios at 10100 Hz using a variety of sensor modalities such as high-resolution color and grayscale stereo cameras, a Velodyne 3D laser

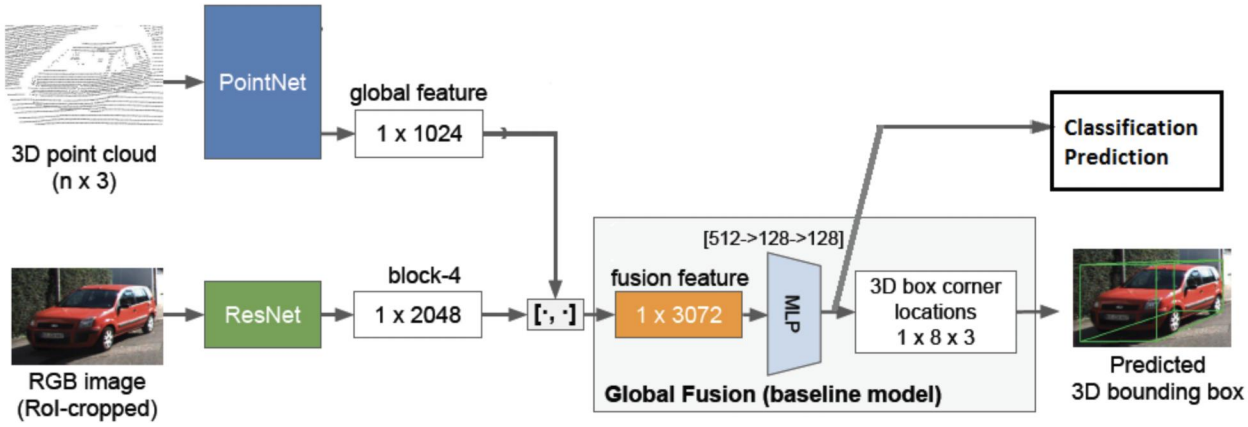

Figure 1. An overview of PointFusion architecture [XAJ17]

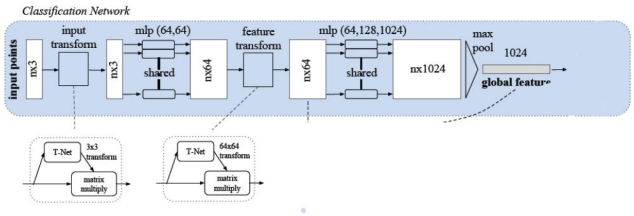

Figure 2. An overview of the PointNet layer

scanner and a high-precision GPS/IMU inertial navigation system.

For our purpose, we use the point clouds generated by the Velodyne 3D laser scanner and the corresponding left color image captured by a Sony ICX267 CCD. In all, we have 7481 examples that are split for training, development, and test, in a 90-5-5 percentage ratio. The split is random and the number of precise examples in each set are given in Table 1.

	Train Data	Dev. Data	Test Data
No. of examples	6,750	365	366

Table 1. Train-Dev-Test Split

4.2. Pre-Processing

The velodyne setup on the station wagon that produces the point clouds is a rotating 3D laser scanner that generates data points at a rate of 10 Hz, 64 beams, with 0.09 degree angular resolution, 2 cm distance accuracy, collecting 1.3 million points/second, with a horizontal and vertical field of view of 360° and 26.8°, respectively. These are a lot of points and we need to trim down the input size for correspondence, feasibility and relevance. For our use, we filter the point clouds falling in the camera view angle and randomly sample 2048 points from it. These are then fed through a Spatial Transformation Network in order to canonicalize the input space. Further, the ground truth labels are transformed to the velodyne coordinates for tractability in prediction.

4.3. Network Specifications

This subsection lists the network specifications arrived at after extensive experimentation. The used model has 1,808,027 trainable parameters, most of them belonging to the PointNet architecture. The original PointNet model [QSMG17] uses batch normalization following all fully connected layers to reduce the covariance shift in the input features. However, we found that batch normalization hampers the 3D bounding box estimation performance. our understanding on this phenomenon is that batch normalization aims to eliminate the scale and bias in its input data, which is detrimental to the task of 3D regression, where absolute numerical values of the point locations are helpful. Therefore, our PointNet variant has all batch normalization layers removed.

Additionally, we are using Adam Optimization and a decaying learning rate for training.

4.4. Loss Function

Using an optimum loss function is an important selection of training an effective model. Since our model has two outputs, the loss function used to train the model for the classification is mean-squared error (gave better results than categorical cross entropy loss) and the box output is smoothL1 given by:

$$L = \sum_i smoothL1(x^{i*}_{offset}, x^i_{offset})$$

Here, x^{i*}_{offset} is the offset between the ground truth box corner locations and the i-th input point.

SmoothL1 loss was found to perform better than a mean squared error loss, as the loss value is not compromised for a false result with very distinct values than a ground truth. Figure 3 shows the accuracy curve from 100 epochs for the bounding box output using mean squared error, which when compared to the main model's loss curves in Figure 4 validates the reason for choice. The box accuracy for smoothL1 is more than the box accuracy for mean squared error.

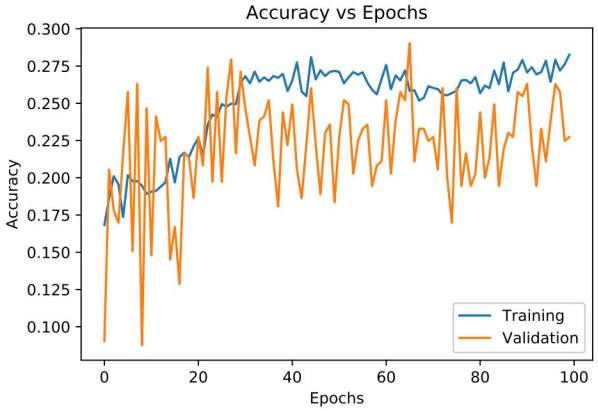

Figure 3. Box output accuracy vs Epochs (100) for training with mean squared error

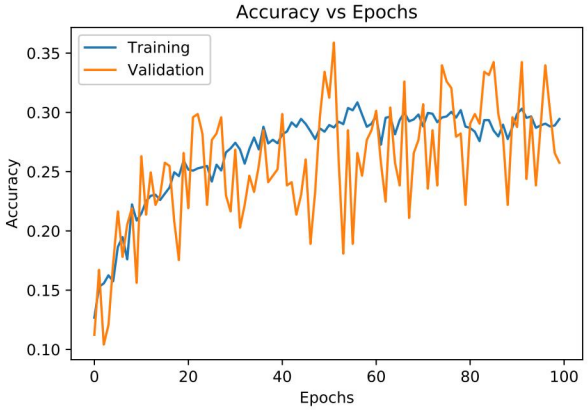

Figure 4. Box output accuracy vs Epochs (100) for training with smoothL1

4.5. Results

Thus, with these important training choices, the model was trained to give the loss curves, presented in Figure 5 and Figure 6. It may be noted here that the model has some avoidable variance in absence of batch normalization layers. However, a better box corners prediction accuracy is the reward.

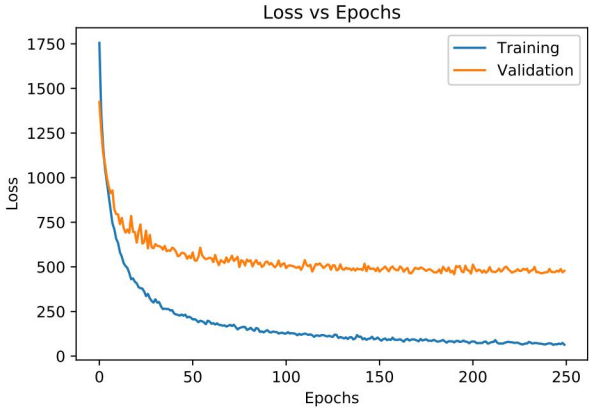

Figure 5. Box output loss vs Epochs (250)

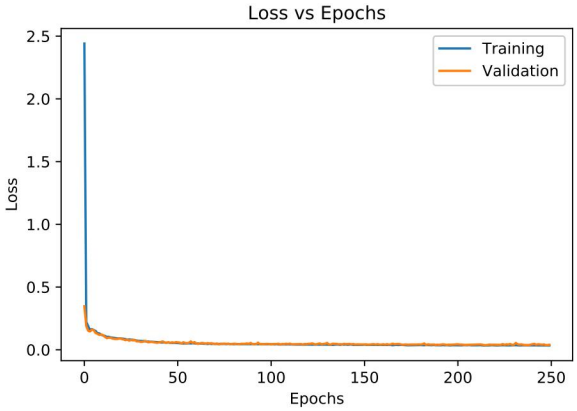

Figure 6. Classification ouptut loss vs Epochs (250)

Table 2 offers more information about the model's performance on the training, development, and test set. The bounding box accuracy of the model is evaluated by using average IOU. Just for reference of performance scale, [PLJ+17] gets a best case IOU of 0.55 on UW-RGBD dataset.

	Train Data	Dev. Data	Test Data
Class. Accuracy	96.27%	96.16%	95.62%
Box Average IOU	0.73	0.73	0.71

Table 2. Model Performance

Further, Figures 7, 8 and 9 demonstrate some correct result predictions by the developed PointFusion model for the considered categories of image examples (Red: Ground truth; Ink Blue: Car; Cyan Blue: Pedestrian; Green: Van). It may be noted here that the reason these classification classes are chosen is because they form the dominant category of annotations in the KITTI dataset. An interesting observation to note here is that the model is able to perform well even for challenging examples like the highly occluded

car in Figure 7, giving an IOU as high as 0.84.

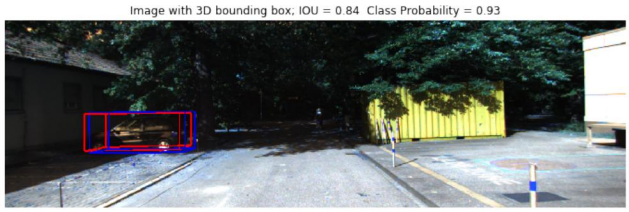

Figure 7. Correct Prediction: Car

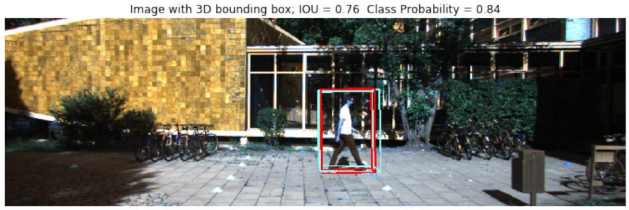

Figure 8. Correct Prediction: Pedestrian

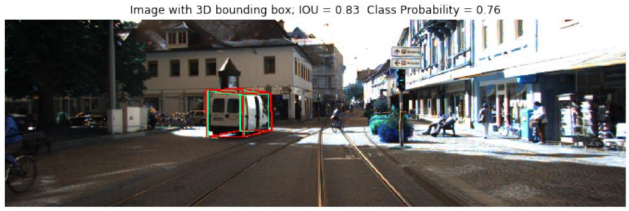

Figure 9. Correct Prediction: Van

Also, Figures 10, 11, and 12 show some incorrect model prediction cases with an objective to derive a holistic understanding about the model's performance. The model's prediction with Figure 10 is almost correct in terms of localization and volume estimate. However, it fails to pass the threshold set for an IOU to be classified as correct. Sometimes, liberal and non-tight ground truth annotations can also be a reason for such types of errors.

Figure 11 shows a worse prediction, which can be considered almost correct in terms of localization but not with respect to volume. Further, Figure 12 shows an incorrect prediction with respect to localization and volume estimate of the bounding box. Perhaps, this example is extremely difficult as the car is highly occluded.

5. Conclusion

This work performs deep fusion of camera and lidar information using a PointFusion model and is able to achieve credible 3D bounding box estimation and recognition results. The strength of the model comes from the fact that

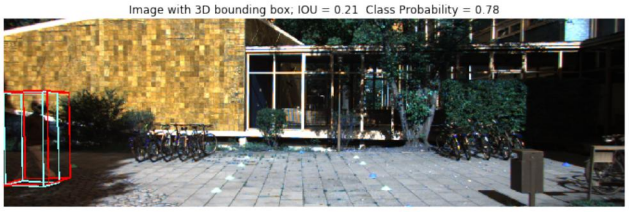

Figure 10. Wrong Prediction: IOU Calculation

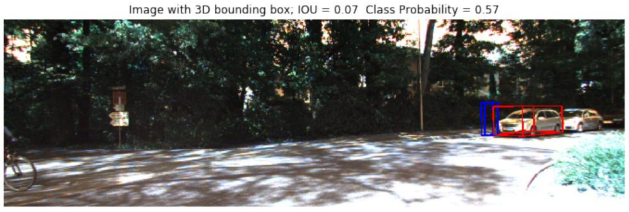

Figure 11. Wrong Prediction: Volume Estimate

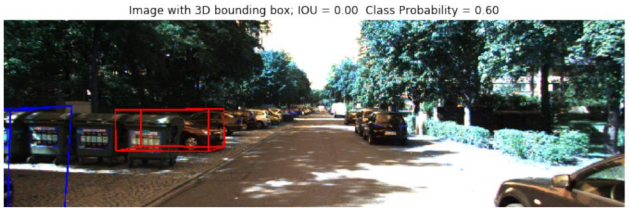

Figure 12. Wrong Prediction: Volume & Position Estimate

inputs are processed using a heterogeneous network architecture and lossy input preprocessing is duly avoided. Also, unlike other methods that use a multi-stage pipeline and perform late fusion by employing techniques like Kalman Filters, the used model performs an early deep fusion. In our hope, such a technique can prove useful with industries beyond autonomous vehicle technology, industrial robotic arms being one such case.

6. Future Work

In the presented work, the variance of the regression target is directly dependent on the particular scenario. Generating box proposals by sliding windows instead of directly regressing can prove useful. Also, instead of randomnly sampling a set of points from the camera view angle, biasing the sampling for points in regions of interest can improve results. Also, a single end-to-end 3D detector can be a promising research possibility.

7. Contributions

Ayush Gupta: Lidar Chain
Malavika Bindhi: Camera Chain

8. Code

https://github.com/malavikabindhi/CS230-PointFusion

References

[AGP+17] Alireza Asvadi, Luis Garrote, Cristiano Premebida, Paulo Peixoto, and Urbano J. Nunes. Multimodal vehicle detection: fusing 3d-lidar and color camera data. In *Pattern Recognition Letters*, 2017.

[AME+14] Mathieu Aubry, Daniel Maturana, Alexei A Efros, Bryan C Russell, and Josef Sivic. Seeing 3d chairs: exemplar part-based 2d-3d alignment using a large dataset of cad models. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3762–3769, 2014.

[CMW+17] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *IEEE CVPR*, volume 1, page 3, 2017.

[FTVG06] Vittorio Ferrari, Tinne Tuytelaars, and Luc Van Gool. Simultaneous object recognition and segmentation from single or multiple model views. *International Journal of Computer Vision*, 67(2):159–188, 2006.

[HZRS15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[Li17] Bo Li. 3d fully convolutional network for vehicle detection in point cloud. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 1513–1518. IEEE, 2017.

[PLJ+17] Ramanpreet Singh Pahwa, Jiangbo Lu, Nianjuan Jiang, Tian Tsong Ng, and Minh N Do. Locating 3d object proposals: A depth-based online approach. *arXiv preprint arXiv:1709.02653*, 2017.

[QSMG17] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 1(2):4, 2017.

[SH16] Heiko G. Seif and Xiaolong Hu. Autonomous driving in the icity hd maps as a key challenge of the automotive industry. In *Engineering Volume 2, Issue 2, Pages 159-162*, 2016.

[SX14] Shuran Song and Jianxiong Xiao. Sliding shapes for 3d object detection in depth images. In *European conference on computer vision*, pages 634–651. Springer, 2014.

[SX16] Shuran Song and Jianxiong Xiao. Deep sliding shapes for amodal 3d object detection in rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 808–816, 2016.

[XAJ17] Danfei Xu, Dragomir Anguelov, and Ashesh Jain. Pointfusion: Deep sensor fusion for 3d bounding box estimation. 11 2017.

[XCLS17] Yu Xiang, Wongun Choi, Yuanqing Lin, and Silvio Savarese. Subcategory-aware convolutional neural networks for object proposals and detection. In *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*, pages 924–933. IEEE, 2017.

[ZDY+14] Menglong Zhu, Konstantinos G Derpanis, Yinfei Yang, Samarth Brahmbhatt, Mabel Zhang, Cody Phillips, Matthieu Lecce, and Kostas Daniilidis. Single image 3d object detection and pose estimation for grasping. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 3936–3943. IEEE, 2014.