International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. , No. (Mon-20..)

http://dx.doi.org/10.12785/ijcds/XXXXXX

Multi-Sensor Data Fusion using DarkNET - CNN

Vinodh S[1] and Ramakanth P[2]

[1,2]*Department of Computer Science, R V College of Engineering, Bengaluru, India*

Received Mon. 20, Revised Mon. 20, Accepted Mon. 20, Published Mon. 20

Abstract: Multi-sensor data fusion is ubiquitous; therefore, the associated research is significant. There are several instances in the day-to-day activities where data fusion can be observed. The present generation autonomous driving system requires a thorough understanding followed by a voluminous dataset for training the model. The experimental data of imagery and proximity sensors are significant for the model's performance. The projection of the camera to LiDAR proves ineffective as the semantic density of the camera is suppressed in the process. The present work attempts to enhance the conventional point-level fusion techniques by allocating prime importance to semantic density. This is facilitated by performance optimization by identifying the hindrances and enhancing the transformation of the view by the Bird's-eye-View pooling. The object tracking is facilitated through the Extended Kalman Filter(EKF) by fusing the LiDAR data with the camera detections. The detection precision is found to be 0.9546, and the detection recall is 0.9344, while the mAP is evaluated to be 71.2%

Keywords: sensor fusion, LiDAR, multi-sensor data, DarkNET, Convolutional Neural Network(CNN)

1. Introduction

Autonomous driving has observed increased complexity recently due to several sensors involved in the act. The inevitable usage of camera and LiDAR-based sensors mounted on the vehicle demonstrates the significance of the Multi-Sensor Data Fusion(MSDF). The LiDAR sensor provides data in three-dimensional space, while the camera generates two-dimensional data. For a reliable and precise perception of the data, the semantic information from the camera and spatial information of the LiDAR should be mapped. Hence, MSDF forms the crucial aspect of autonomous driving.

There have been several efforts to develop reliable three-dimensional object detection systems for autonomous driving cars. Laser-based sensors perform exceptionally well in relation to the depth of information, while cameras provide semantic information to a greater depth. Therefore, a fusion of camera and LiDAR-based sensors complement each other, permitting the development of a formidable three-dimensional detection system for a safe and exceptional autonomous driving experience.

However, there are associated challenges due to the difference in modalities generated by the data of each sensor. In order to achieve a multi-modal and multi-task fusion, there is a need for a unified representation of the data from different sensors. In the earlier studies, the perception in two-dimensional space has been a great success, which is extended through the projection of spatial LiDAR data onto the semantic camera data. The distortion of the geometry observed when the LiDAR data is projected onto the camera(Fig.1a) rendered the process less effective in terms of object detection in three-dimensional space[1].

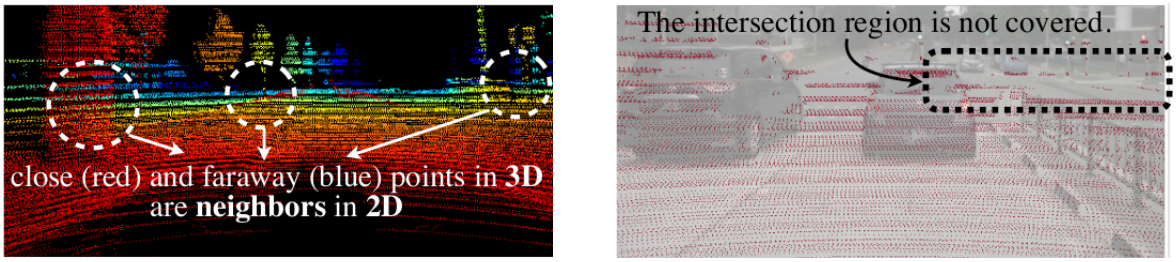
Some of the recent efforts in sensor fusion aim at enhancing the LiDAR point cloud data with CNN features[2], semantic labels[3], [4] and two-dimensional image-based virtual points[5]. Though there is a commendable detection performance on large-scale benchmarks, the point-level-based fusion is less impressive on tasks of semantic nature such as BEV-Segmentation[6], [7], [8], [9], which can be attributed to the semantically-lossy behavior of the projection of camera to LiDAR(Fig.1b). Further, the differences in density are more pronounced for sparser LiDAR data.

2. Literature Survey

Over a decade, immense efforts have been put forth to develop a reliable and robust method for the fusion of sensors with different modalities. However, there is a great scope for developing more sophisticated and accurate models, as the existing models are identified with few challenges in overcoming the projection accuracy through reduction in geometric and semantic losses.

The earlier works to achieve three-dimensional perception based on LiDAR-only data include the single-stage 3D Object detectors[10], [8], [11], [12], [13], which provided the platform for the evolution of many robust and sophisticated models. The model is enhanced by using PointNets[14] and SparseConvNet[15] for extracting the flattened point-cloud features. Nevertheless, the restriction offered through the bounding box in the earlier models is overcome by introducing the anchorless

E-mail address: vinodh23021988@gmail.com



(a) *Geometric-lossy* (b) *Semantic-lossy*

Figure 1. Projection losses[1]

models[16], [17], [18], [19]. Further investigations have led to the development of two-stage models through the amalgamation of the Region-based Convolutional Neural Network(R-CNN) architecture with the existing one-stage-based object detection model[20], [21], [22], [23], [24], [25]. The most crucial task for the offline construction of HD maps is the three-dimensional segmentation of the semantic features. The models[15], [26], [27], [28], [29] developed to address the seminal task, analogous to U-Net, are note-worthy.

The LiDAR sensors are expensive, paving the way for exploring cheaper means of 3D object detection. Commendable efforts are made to achieve three-dimensional perception based only on Camera data. The FCOS3D[30] model utilizes three-dimensional regression branches suitably coupled with the image detectors[31], which is later enhanced to achieve greater depth in detection[32], [33]. Irrespective of perspective view-based object detection, models that learn from the object queries in the three-dimensional space coupled with the Deformable Transformer(DETR)[34] detection model, *viz.* DETR3D[35], PETR[36] and Graph-DETR3D[37], are also developed. The view transformer-based camera-only three-dimensional perception models explicitly transform camera data to perspective bird's eye view[6], [38], [39], [7]. The state-of-art models such as BEVDet[40] and M^{2}BEV[41], utilize Lift-Splat- Shoot(LSS)[7] and Orthographic Feature Transform(OFT)[39] for three-dimensional object detection. Also, the three-dimensional object detection models through time-dependent cues using multiple cameras *viz.* BEVDet4D[42], BEVFormer[43] and PETRv2[36] are some salient developments in single-frame methods. However, the models such as BEVFormer[43], CVT[9], and EGO3RT[44] also perform exceptionally well through multi-head attention for view transformation.

Lastly, there are efforts put forth to study the models for multi-task learning. Simultaneous detection of objects and instant segmentation form the key aspects of multi-task learning[45], [46]. Further, the simultaneous detection and segmentation is extended to human-object interaction[42], [47], [48], [49]. The models that perform detection of object and instance segmentation simultaneously are M^2BEV[41], BEVFormer[43] and BEVerse[50]. However, there are a couple of challenges associated with the models specified. Firstly, the models have not considered multi-sensor data fusion, and secondly, the computational time and hardware requirement to carry out the activities simultaneously significantly add to the computational cost. The MMF model[51] though performs detection and segmentation simultaneously, it is object-centric, which cannot be extended to BEV Segmentation. The most recent attempts aim to significantly improve the detection performance by fusing sensors of different modalities. The methods can be categorized into *proposal-level* and *point-level*. The proposal-level methods are *object-centric* and therefore do not support map segmentation effectively, whereas point-level techniques are both *object-centric* and *geometric-centric*. Some of the exceptional contributions towards proposal-level techniques include MV3D[52], F-PointNet[53], F-ConvNet[48], CenterFusion[54]. FUTR3D[55] and TransFusion[56], while point-level techniques include PointPainting[2], PointAugmenting[3], MVP[5], FusionPainting[57], AutoAlign[58], DeepContinuousFusion[51], Deep Fusion[4], and FocalSparseCNN[59]. Not all techniques can be incorporated to process the camera and LiDAR data. LiDAR data processing can be carried out very effectively through input-level decoration models *viz.* PointPainting[2], PointAugmenting[3], MVP[5], FusionPainting[57], AutoAlign[58], and FocalSparseCNN[59], while camera images require feature-level decoration *viz.* DeepContinuousFusion[51], Deep Fusion[4]

The present study attempts to develop a model for three-dimensional object detection through data from multiple sensors(three cameras and three LiDAR). Further, the geometric and semantic information from the sensors are granted equal weightage.

3. Methodology

The methodology adopted for the present study is discussed in this section.

A. Dataset

The dataset comprises three Camera RGB Images and three LiDAR data. The camera images are well-nourished with semantic information, while the LiDAR data precisely provide the spatial information. The samples three monocular images from the camera, covering 180° field of view, and three 32-beam LiDAR data scans.

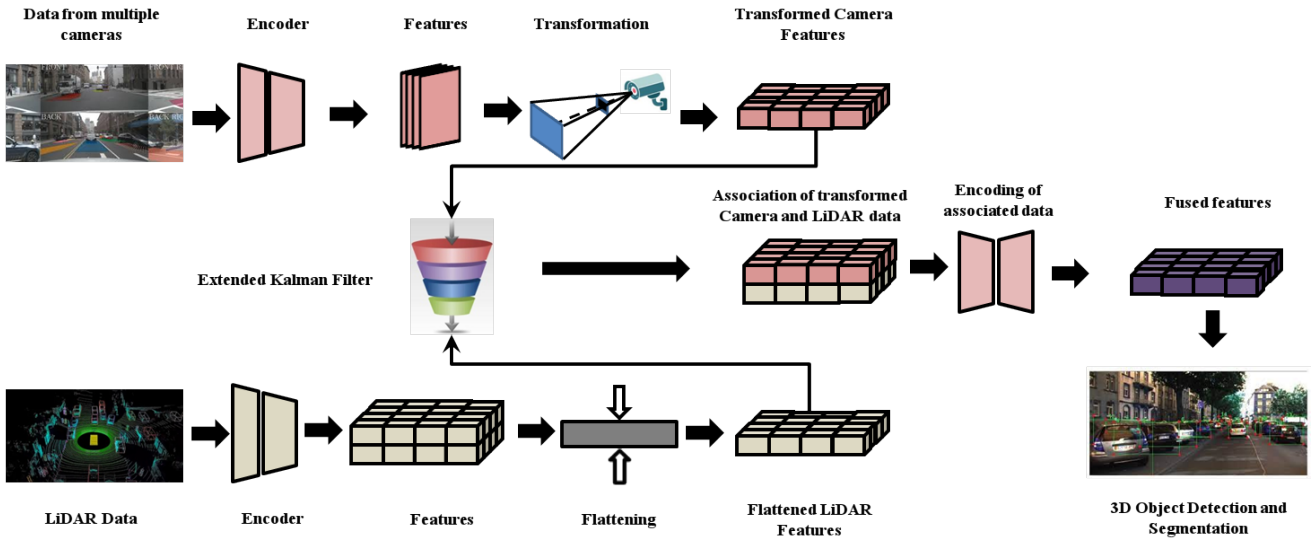


Figure 2. Framework

B. Framework

The framework followed for the present study is depicted in Fig.2, where three cameras and LiDAR sensor data are input to the respective encoders. The encoders are convolutional and follow the DarkNET-CNN architecture depicted in Fig.3. The features of the camera images are extracted and transformed, which forms the key aspect to achieve higher accuracy. LSS[7] and BEVDet[60] models are followed to achieve the transformation of camera images. The transformed images are filtered through an Extended Kalman Filter(EKF) before the data is fused. EKF processes the transformed, inherently non-linear LiDAR and Camera data. Therefore, EKF generates a system matrix and evaluates noise-covariance by compensating for the quadratic effects of the data.

The filtered data is mapped by estimating the error through the update-and-predict of the EKF input state. The root mean squared error estimated during the present study for a single target is around 0.32. EKF predicts the model's state, while Mahalanobis distance(MD) matches the states of multiple sensors[61] and the EKF updates the state based on the error generated by the MD calculation. Eq.1 is used to evaluate the MD, where x is the observations to be made, X_i is the calibration data set for the corresponding i^{th} sensor, while $\overline{X_i}$ is the mean and M_i is the root-mean-squared-error of the i^{th} sensor calibration data.

$$D^2(x) = (x - \overline{X_i}) \times M_i(x - \overline{X_i}) \quad (1)$$

C. Model

DarkNet-CNN generates the feature map by fusing the multi-scale camera images, which are down-sampled to 256×704. The LiDAR data is handled using VoxelNET[10] model, with the data down-sampled to 0.075 and 0.1 for detection and segmentation, respectively. The three crucial activities that have direct implications on the accuracy are listed in the section 3-C1, section 3-C2 and section 3-C3.

1) Unified Representation

Distinct qualities may be present in various viewpoints. LiDAR and radar features, for example, are usually in the three-dimensional bird's-eye view, whereas camera features are in the perspective view. Every camera function, such as front, back, left, and right, has a unique viewing angle. Due to this perspective mismatch, feature fusion becomes challenging because the same element may correspond to entirely different spatial locations in distinct feature tensors (naïve element-wise feature fusion will not operate in this scenario). Thus, it is imperative to identify a shared representation that is easily convertible to it without sacrificing information and appropriate for various purposes[1].

2) To Camera

One option is to project the LiDAR point cloud onto the camera plane and display the $2.5D$ sparse depth driven by RGB-D data. This conversion is geometrically lossy. In the 3D space, two neighbors on the depth map may be very far apart. For activities like 3D object detection that rely on the geometry of the item or scene, this reduces the effectiveness of the camera view.

3) To LiDAR

The majority of cutting-edge sensor fusion techniques [2], [5], [4] embellish LiDAR points with the matching camera features (e.g., virtual points, CNN features, or semantic labels). But this projection from the camera to LiDAR is semantically lossy. Because of the stark differences in densities between LiDAR and camera features (for a 32-channel LiDAR scanner) < 5% of camera features match a LiDAR point. On semantic-oriented tasks (such as BEV map segmentation), the model's performance is significantly

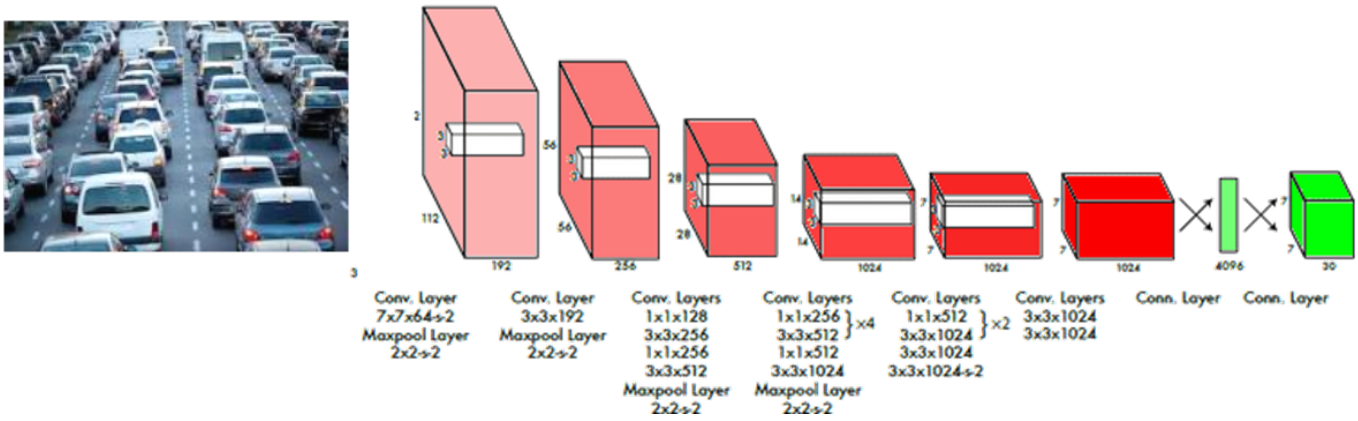


Figure 3. DarkNET-CNN Architecture

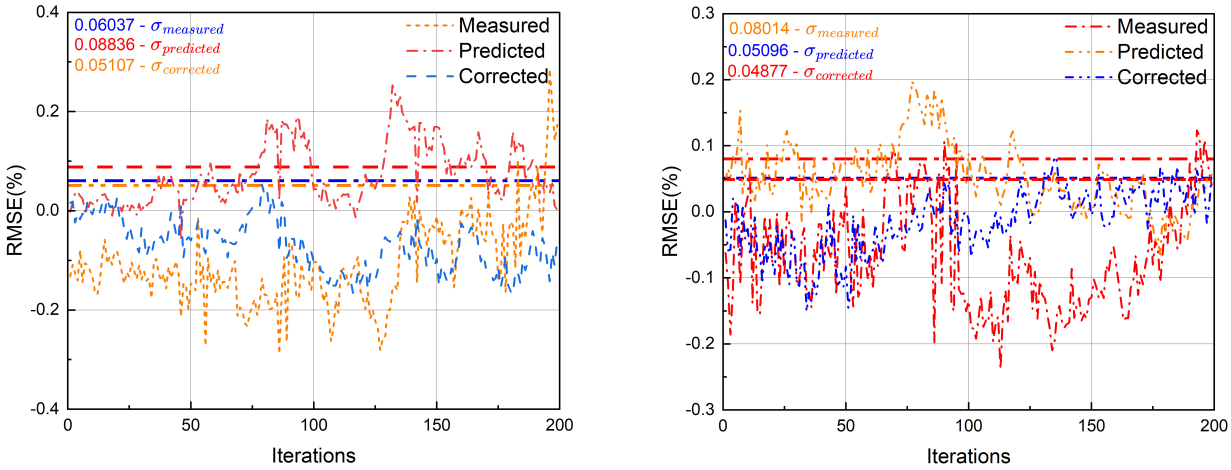
(a) IoU = 0.954(Excellent) (b) IoU = 0.739(Good) (c) IoU = 0.453(Poor)

Figure 4. Intersection over Union

affected by giving up the semantic density of camera features. More modern fusion techniques in the latent space, including object query, have comparable demerits[56], [18].

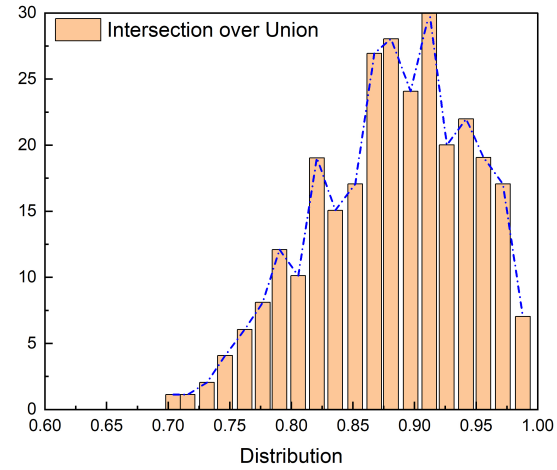
4) To BEV

The lossy identified and explained through Fig.1a and Fig.1b are considered during the transformation. The projection of LiDAR data to BEV evens out the sparse features in the height dimension, thereby eliminating the aspect

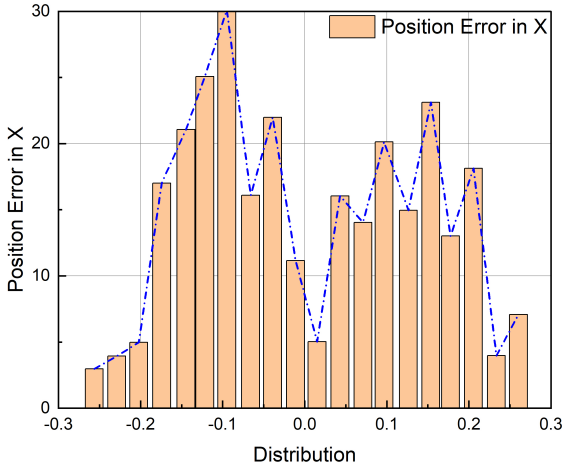


(a) Detected Signal (b) Tracking Signal

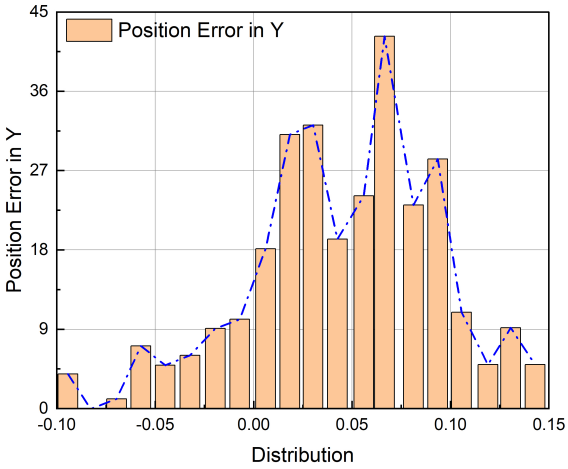
Figure 5. Treatment of detection and tracking signals



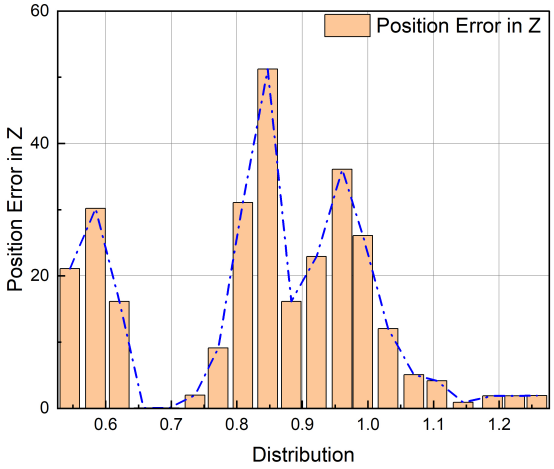
(a) Intersection Over Union



(b) X-Error in Signal



(c) Y-Error in Signal



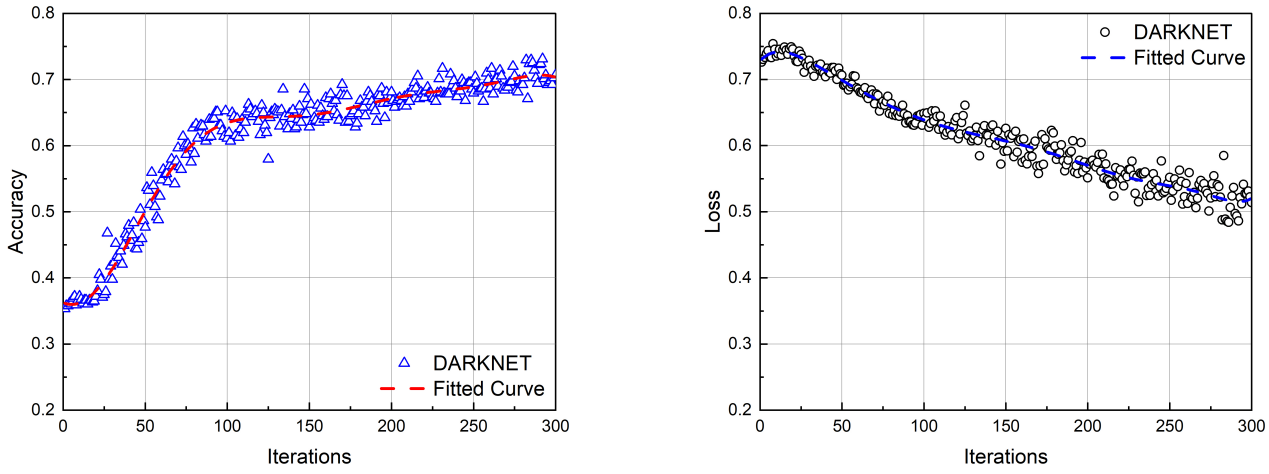
(d) Z-Error in Signal

Figure 6. Evaluation of Object Detection Performance

of geometric lossy. On the contrary, the transformation of camera images to BEV is non-trivial due to the inherent depth associated. The depth distribution of the pixels of the camera images is predicted using LSS[7] and BEVDet[40], [60]. The features are re-scaled upon scattering of the pixels of each feature to $\mathcal{D}$ discrete points along the ray of the camera. A cloud of the feature points is generated with a size of $\mathcal{NHWD}$, where $\mathcal{N}$ is the number of the cameras(in the present study, it is 3 numbers), while, $\mathcal{H}$ and $\mathcal{W}$ are the height and the width of the image, respectively. The grid size considered in the cartesian coordinate system is $0.35m \times 0.35m$, which is evened out in the z-direction.
The transformation of camera-to-BEV consumed a computational time of $\approx 465ms$ with a Quadro P6000 Graphics processing. This can be attributed to the large number of grid points generated per frame of the camera feature. The LiDAR features are, therefore, less dense and computationally inexpensive. Nevertheless, curtailing the computational time for the camera features demands a pre-computation and reduction in the interval considered earlier.
The *Pre-computation* involves the association of the camera features to BEV grid points. From the calibration of the camera, the intrinsic and extrinsic stay the same, permitting to locate coordinates of the feature cloud of the camera. The task facilitates the pre-computing of the indices of BEV grid points, thereby reducing the grid-association latency by $\approx 65\%$.
The *Interval Reduction* entails aggregation of the grid-points generated during the precomputation through symmetric functions *viz. mean, maximum, and summation*, within the BEV grid.

D. Multi-tasking

Practically, the majority of the 3D perception activity is carried out under detection and segmentation. The object center is evaluated based on the size, velocity, and rotation, which is based on the earlier 3D detection articles[56], [16], [5]. On the other hand, the segmentation is carried out by



(a) Accuracy (b) Loss

Figure 7. Performance of DarkNET-CNN based MSDF model

treating the features classwise and associating the binary segments to each of them. The training of the segmentation head is carried out through CVT[9], with the focal loss being treated using Lin *et.al* model[62].

E. Training

The training of the model is carried out end-to-end to avoid camera-encoder freezing, as observed in earlier models[2], [3], [56]. The weight decay is ≈ 0.001, and optimization is achieved through AdamW[63] model.

F. Metrics

The evaluation of the model is made based on the following parameters discussed in section 3-F1 and section 3-F2.

1) Intersection over Union(IoU)

The accuracy with which the data is predicted can be obtained through Intersection over Union*(IoU)*, which is defined as the percentage of overlap between the actual value(ground-truth) and the predicted value(Fig.4), mathematically represented by Eq.2.

$$IoU = \frac{|A \cap B|}{|A| \cup |B|} \tag{2}$$

where A is the ground truth and B is the predicted value. IoU can be given by Eq.3 for binary data classification.

$$IoU = \frac{TP}{TP + FP + FN} \tag{3}$$

where TP is the True Positive, FP is the False Positive and FN is the False Negative

2) Mean Average Precision(mAP)

The mAP is calculated based on the Average Precision(AP) obtained from the area under the precision-recall curve. The AP is averaged for N samples as indicated by Eq.4 to obtain mAP.

$$mAP = \frac{1}{N} \sum_{i=1}^{N} AP_i \tag{4}$$

4. Results and Discussion

Based on the methodology defined in the earlier section, LiDAR signals are processed for detection and tracking. The predicted signal generated through the EKF is compared with the measured signal, which is corrected based on the root-mean-squared error(RMSE) represented in percentage. The standard deviation between measured and predicted data for the detected and tracking signals is ≈ 9%, which is corrected to achieve a standard deviation between measured and corrected signal as ≈ 5%(Fig.5a). Also, for tracking signal, the RMSE for predicted values ≈ 5%, which is corrected to achieve an error of ≈ 4.8% Fig.5b).

Fig.6a represents the IoU for the model, which demonstrates a good performance with a minimum score of 70%, while most of the distribution is within the range of 85−98%. The error plots Fig.6b, and Fig.6c show symmetricity about zero with the maximum distribution close to zero, whereas in the case of error plot in the Z-direction, the range is between 0.5 to 1, with maximum peaks between to 0.8−1. The mean position errors in X, Y, and Z directions are 0.0049, 0.0453, and 0.8247, respectively.

The detection performance can be evaluated through accuracy and loss data plots as depicted in Fig.7a and Fig.7b, respectively. The model's accuracy is ≈ 72% while the loss is calculated to be ≈ 51%. The detection precision and recall are ≈ 0.9546 and ≈ 0.9344, respectively, and mAP is 71.2. The results are compared with the existing models, as demonstrated in Table.I. It can be observed that the model's performance is marginally better than the BEVFusion, which is ≈ 1.4%. However, for the present study, three

Models	Modality	mAP
BEVDet[40]	C	42.2
M^{2}BEV[41]	C	42.9
BEVFormer[43]	C	44.5
BEVDet4D[60]	C	45.1
PointPillars[8]	L	-
SECOND[64]	L	52.8
CenterPoint[16]	L	60.3
PointPainting[2]	C+L	–
PointAugmenting[3]	C+L	66.8
MVP[5]	C+L	66.4
FusionPainting[57]	C+L	68.1
AutoAlign[58]	C+L	–
FUTR3D[55]	C+L	–
TransFusion[56]	C+L	68.9
BEVFusion[1]	C+L	70.2
DarkNET-CNN model(present work)	C+L	**71.2**

TABLE I. Comparison with the existing models

cameras and three LiDAR data are used, unlike 6 Cameras and 1 LiDAR data in the case of BEVFusion model[1].

5. Conclusion

It is evident from the earlier discussion that many MSDF models have demonstrated greater accuracy in the recent past. However, challenges persist that can be attributed to the environmental or operating conditions that induce errors in the data as discussed in section 1. A formidable correction has to be incorporated, which otherwise can affect the accuracy of the model. The model presented in this paper attempts to fuse the multi-modal data from different sensors in order to enhance object detection for future autonomous driving purposes.

The model demonstrates performance that is fairly well placed against the existing models, particularly BEVFusion. The BEVFusion model was developed by considering six cameras and one LiDAR data, while the present model considers three cameras and three LiDAR data for fusion. Hence, there are differences in the modalities handled in the course of development of the model. However, the present model is observed to have an accuracy of 72% with detection precision and recall of 0.9546 and 0.9344, respectively. The mean Average Precision is 71.2%, which is marginally better than BEVFusion by $\approx 1.3\%$.

There is still great scope for developing multi-modal 3D object detection models, with inherent challenges associated with accurate depth estimation. The model can be improved by utilizing ground-truth to supervise the view-transformer[65], [66] that can be considered for future developments.

References

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

[2] S. Vora, A. H. Lang, B. Helou, and O. Beijbom, "Pointpainting: Sequential fusion for 3d object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 4604–4612.

[3] C. Wang, C. Ma, M. Zhu, and X. Yang, "Pointaugmenting: Cross-modal augmentation for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 11 794–11 803.

[4] Y. Li, A. W. Yu, T. Meng, B. Caine, J. Ngiam, D. Peng, J. Shen, Y. Lu, D. Zhou, Q. V. Le *et al.*, "Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17 182–17 191.

[5] T. Yin, X. Zhou, and P. Krähenbühl, "Multimodal virtual point 3d detection," *Advances in Neural Information Processing Systems*, vol. 34, pp. 16 494–16 507, 2021.

[6] B. Pan, J. Sun, H. Y. T. Leung, A. Andonian, and B. Zhou, "Cross-view semantic segmentation for sensing surroundings," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4867–4873, 2020.

[7] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer, 2020, pp. 194–210.

[8] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 12 697–12 705.

[9] B. Zhou and P. Krähenbühl, "Cross-view transformers for real-time map-view semantic segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 13 760–13 769.

[10] Y. Zhou and O. Tuzel, "Voxelnet: End-to-end learning for point cloud based 3d object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4490–4499.

[11] Z. Yang, Y. Sun, S. Liu, and J. Jia, "3dssd: Point-based 3d single stage object detector," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 11 040–11 048.

[12] B. Zhu, Z. Jiang, X. Zhou, Z. Li, and G. Yu, "Class-balanced grouping and sampling for point cloud 3d object detection," *arXiv preprint arXiv:1908.09492*, 2019.

[13] Y. Zhou, P. Sun, Y. Zhang, D. Anguelov, J. Gao, T. Ouyang, J. Guo, J. Ngiam, and V. Vasudevan, "End-to-end multi-view fusion for 3d object detection in lidar point clouds," in *Conference on Robot Learning*. PMLR, 2020, pp. 923–932.

[14] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "Pointnet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in neural information processing systems*, vol. 30, 2017.

[15] B. Graham, M. Engelcke, and L. Van Der Maaten, "3d semantic segmentation with submanifold sparse convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 9224–9232.

[16] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3d object detection and tracking," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 11 784–11 793.

[17] R. Ge, Z. Ding, Y. Hu, W. Shao, L. Huang, K. Li, and Q. Liu, "1st place solutions to the real-time 3d detection and the most efficient model of the waymo open dataset challenge 2021," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, vol. 1, 2021.

[18] Q. Chen, L. Sun, Z. Wang, K. Jia, and A. Yuille, "Object as hotspots: An anchor-free 3d object detection approach via firing of hotspots," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*. Springer, 2020, pp. 68–84.

[19] C. R. Qi, Y. Zhou, M. Najibi, P. Sun, K. Vo, B. Deng, and D. Anguelov, "Offboard 3d object detection from point cloud sequences," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6134–6144.

[20] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 770–779.

[21] C. Yilun, L. Shu, S. Xiaoyong, and J. Jiaya, "Fast point r-cnn," in *ICCV*, 2019.

[22] S. Shi, Z. Wang, J. Shi, X. Wang, and H. Li, "From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 8, pp. 2647–2664, 2020.

[23] S. Shi, C. Guo, L. Jiang, Z. Wang, J. Shi, X. Wang, and H. Li, "Pv-rcnn: Point-voxel feature set abstraction for 3d object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 529–10 538.

[24] S. Shi, L. Jiang, J. Deng, Z. Wang, C. Guo, J. Shi, X. Wang, and H. Li, "Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. arxiv 2021," *arXiv preprint arXiv:2102.00463*.

[25] Z. Li, F. Wang, and N. Wang, "Lidar r-cnn: An efficient and universal 3d object detector," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7546–7555.

[26] C. Choy, J. Gwak, and S. Savarese, "4d spatio-temporal convnets: Minkowski convolutional neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3075–3084.

[27] H. Tang, Z. Liu, S. Zhao, Y. Lin, J. Lin, H. Wang, and S. Han, "Searching efficient 3d architectures with sparse point-voxel convolution," in *European conference on computer vision*. Springer, 2020, pp. 685–702.

[28] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10 012–10 022.

[29] X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin, "Cylindrical and asymmetrical 3d convolution networks for lidar segmentation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 9939–9948.

[30] T. Wang, X. Zhu, J. Pang, and D. Lin, "Fcos3d: Fully convolutional one-stage monocular 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 913–922.

[31] Z. Tian, C. Shen, H. Chen, and T. He, "Fcos: Fully convolutional one-stage object detection," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9627–9636.

[32] T. Wang, Z. Xinge, J. Pang, and D. Lin, "Probabilistic and geometric depth: Detecting objects in perspective," in *Conference on Robot Learning*. PMLR, 2022, pp. 1475–1485.

[33] H. Chen, P. Wang, F. Wang, W. Tian, L. Xiong, and H. Li, "Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2781–2790.

[34] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," *arXiv preprint arXiv:2010.04159*, 2020.

[35] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *Conference on Robot Learning*. PMLR, 2022, pp. 180–191.

[36] Y. Liu, T. Wang, X. Zhang, and J. Sun, "Petr: Position embedding transformation for multi-view 3d object detection," in *European Conference on Computer Vision*. Springer, 2022, pp. 531–548.

[37] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, and F. Zhao, "Graph-detr3d: rethinking overlapping regions for multi-view 3d object detection," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 5999–6008.

[38] T. Roddick and R. Cipolla, "Predicting semantic map representations from images using pyramid occupancy networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 11 138–11 147.

[39] T. Roddick, A. Kendall, and R. Cipolla, "Orthographic feature transform for monocular 3d object detection," *arXiv preprint arXiv:1811.08188*, 2018.

[40] J. Huang, G. Huang, Z. Zhu, Y. Ye, and D. Du, "Bevdet: High-performance multi-camera 3d object detection in bird-eye-view," *arXiv preprint arXiv:2112.11790*, 2021.

[41] E. Xie, Z. Yu, D. Zhou, J. Philion, A. Anandkumar, S. Fidler, P. Luo, and J. M. Alvarez, "M^2 bev: Multi-camera joint 3d detection and segmentation with unified birds-eye view representation," *arXiv preprint arXiv:2204.05088*, 2022.

[42] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

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

[44] J. Lu, Z. Zhou, X. Zhu, H. Xu, and L. Zhang, "Learning ego 3d representation as ray tracing," in *European Conference on Computer Vision*. Springer, 2022, pp. 129–144.

[45] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[46] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6154–6162.

[47] K. Sun, B. Xiao, D. Liu, and J. Wang, "Deep high-resolution representation learning for human pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 5693–5703.

[48] Z. Wang and K. Jia, "Frustum convnet: Sliding frustums to aggregate local point-wise features for amodal 3d object detection," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 1742–1749.

[49] G. Gkioxari, R. Girshick, P. Dollár, and K. He, "Detecting and recognizing human-object interactions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8359–8367.

[50] Y. Zhang, Z. Zhu, W. Zheng, J. Huang, G. Huang, J. Zhou, and J. Lu, "Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving," *arXiv preprint arXiv:2205.09743*, 2022.

[51] M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, "Multi-task multi-sensor fusion for 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 7345–7353.

[52] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, "Multi-view 3d object detection network for autonomous driving," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 1907–1915.

[53] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum pointnets for 3d object detection from rgb-d data," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 918–927.

[54] R. Nabati and H. Qi, "Centerfusion: Center-based radar and camera fusion for 3d object detection," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pp. 1527–1536.

[55] X. Chen, T. Zhang, Y. Wang, Y. Wang, and H. Zhao, "Futr3d: A unified sensor fusion framework for 3d detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 172–181.

[56] X. Bai, Z. Hu, X. Zhu, Q. Huang, Y. Chen, H. Fu, and C.-L. Tai, "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 1090–1099.

[57] S. Xu, D. Zhou, J. Fang, J. Yin, Z. Bin, and L. Zhang, "Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 3047–3054.

[58] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, F. Zhao, B. Zhou, and H. Zhao, "Autoalign: pixel-instance feature aggregation for multi-modal 3d object detection," *arXiv preprint arXiv:2201.06493*, 2022.

[59] Q. Chen, S. Vora, and O. Beijbom, "Polarstream: Streaming object detection and segmentation with polar pillars," *Advances in Neural Information Processing Systems*, vol. 34, pp. 26 871–26 883, 2021.

[60] J. Huang and G. Huang, "Bevdet4d: Exploit temporal cues in multi-camera 3d object detection," *arXiv preprint arXiv:2203.17054*, 2022.

[61] J. L. Crowley and Y. Demazeau, "Principles and techniques for sensor data fusion," *Signal processing*, vol. 32, no. 1-2, pp. 5–27, 1993.

[62] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[63] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[64] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolutional detection," *Sensors*, vol. 18, no. 10, p. 3337, 2018.

[65] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander, "Categorical depth distribution network for monocular 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8555–8564.

[66] D. Park, R. Ambrus, V. Guizilini, J. Li, and A. Gaidon, "Is pseudo-lidar needed for monocular 3d object detection?" in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 3142–3152.

Vinodh S is a doctoral student under the supervision of Dr. Ramakanth P.

Dr. Ramakanth P is a Professor at the Department of Computer Science and Engineering, R V College of Engineering, Bengaluru, Karnataka, INDIA. He has 28 years of teaching and 14 years of Research experience. He has supervised and co-supervised more than 13 Doctoral students. He has authored or co-authored more than 63 publications, which include 67 publications in international journals and 18 international conference proceedings. He has published books on Advanced Data Structures Using C++, Object Oriented Programming with C++, and Combined Lab Primer. His research interests include Multi-sensor data fusion, Digital Image Processing, Pattern Recognition and Natural Language processing. He may be contacted at email: ramakanthkp@rvce.edu.in.