Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 183 (2021) 579–588

Procedia Computer Science

www.elsevier.com/locate/procedia

10th International Conference of Information and Communication Technology (ICICT-2020)

A survey of LiDAR and camera fusion enhancement

Huazan Zhong[a], Hao Wang[b], Zhengrong Wu[a], Chen Zhang[c], Yongwei Zheng[c], Tao Tang[d,*]

[a]China Southern Power Grid Company Limited, Guangzhou 510700, China
[b]China southern Power Grid Digital Grid Research Institute co., Ltd, Guangzhou 510663, China
[c]The State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University Wuhan 430072, China
[d]Wuhan Dynspai Technology Company Limited, Wuhan 430079, China

Abstract

Recently, two types of common sensors, LiDAR and Camera, show significant performance on all tasks in 3D vision. LiDAR provides accurate 3D geometry structure, while camera captures more scene context and semantic information. The fusion of two different sensor becomes a fundamental and common idea to achieve better performance. To give a thorough cognition of the complementary and boosting about two kind of sensors. This paper briefly reviews the fusion and enhancement systems between both two sensors in the field of depth completion, 3D object detection, 2D\3D semantic segmentation and 3D object tracking. Meanwhile, the state of art fusion algorithms is quantitatively demonstrated, in this paper, based on the in KITTI widely-used public dataset. Furthermore, the technical challenge and the future potential of LiDAR and camera fusion are also discussed.

Keywords: Fusion enhancement; Survey; KITTI; LiDAR; camera

1. Fusion of LiDAR and camera

As far as the perception in 3D vision is concerned, the perception system based on monocular vision shows satisfactory performance at a low cost, but fails to provide reliable 3D geometric information. Binocular cameras provide 3D geometric information, but the computational cost is high, and cannot work reliably in the environment with high occlusion and no texture. The visual perception system has low robustness under complex illumination

* Corresponding author. Tel.: +86-159-1433-3294
*E-mail address:*dowtowne@dynspai.com

10.1016/j.procs.2021.02.100

conditions, which greatly limits its all-weather sensing capability. However, the LiDAR system is not affected by light conditions and can quickly and efficiently provide high-resolution 3D geometric information of the environment [1]. Nevertheless, due to low resolution, low refresh rate and high cost of high-resolution sensors, LiDAR sensors are also greatly affected by speed shift of targets, different size of targets, local occlusion, real-time requirements and other problems.

For enhancing the performance and reliability of the sensing system, improving the sensing ability of the surrounding environment through the complementary advantages of multiple sensors has become a trend. The fusion technology based on camera and LiDAR has gradually become a research hotspot [2]. At present, several major difficulties and challenges hamper the development of this technology: The camera records information by projecting the real world onto the camera, while the point cloud directly stores the spatial geometric information of the targets. Multi-model fusion is the first problem to be solved. Second is the difference in data expression. The point is irregularly, disordered and continuously stored, while the image is regular, orderly and discrete. A series of differences and problems have led to huge differences in point cloud and image processing algorithms such as whether calibration is required between the two types of systems. How to overcome the differences between the two is of great importance for the study of laser and visual information fusion perception systems. Research on laser radar and camera fusion.

2. Research on LiDAR and camera fusion

2.1. Depth Completion

The disorder and sparseness of the laser point cloud greatly limits and perplexes the 3D perception algorithms. Depth completion aims to deal with this problem by upsampling sparse and irregular depth data into dense, regular data. The method based on LiDAR-camera fusion usually uses high-resolution images to guide depth up-sampling and adopts an encoder-decoder architecture that can produce high-resolution depth images. Most current studies use monocular images to guide depth completion. These methods believe that the color and gray value of the image contain three-dimensional geometric information. Therefore, it can be used as a reference for depth up-sampling.

There are three data fusion levels in monocular images. The first is signal-level fusion. Ma et al. [2] proposed an auto-encoder network based on ResNet, which uses images connecting with sparse depth maps to generate dense depth maps. This method utilizes signal-level fusion which makes its performance better than others. But the pixel-level depth ground truth(GT) is required, which is difficult to acquire. In order to settle this problem, Ma et al. [3] proposed a model-based self-supervised framework. Under the assumption that the target is stationary, only a limited size of images and sparse depth maps are needed for training. In addition, sparse depth constraints, luminosity loss and smoothness loss are utilized to perform self-supervised training. However, the depth outputs it produces is fuzzy, and the input depth may not be preserved. In order to generate clear and dense depth maps real-timely, Cheng et al. [4] input RGB-D images into a new convolutional space propagation network (CSPN). The goal of CSPN is to directly extract image-related affinity matrices from the data, leading to dramatically better results with shorter run time. Cheng et al. [5] further proposed the CSPN++ network architecture, which reduces the amount of calculation by modifying the convolution kernel size and the number of iterations.

The second fusion level is feature-level data fusion. Jaritz et al. [6] proposed an automatic encoding network, which achieves depth completion and semantic segmentation from sparse depth and RGB images in the absence of validity masks. The image and sparse depth map are fed to a parallel encoders based on NASNet and then are merged into the shared decoder. So it can also achieve quite good performance even though the input data is very sparse. Wang et al. [7] designed a PnP module that uses sparse depth maps to improve the performance of existing image-based depth prediction networks. Valada et al. [8] extended the single feature-level fusion to multiple stages of deep network changes. Similarly, GuideNet [9] fuses image features to different stages of the encoder with sparse depth features to guide sparse depth up-sampling. GuideNet is currently the model with best performance in the KITTI.

The last type of fusion is multi-level data fusion. For example, Gansbeke et al. [10] further combine signal-level fusion with feature-level fusion scheme in the image-guided depth completion network. The network consists a global branch and a local branch. Before fusing RGB-D data and depth data, parallel processing ensures the real-time performance of the algorithm. The method ranked the first on the KITTI when the paper is published.

Compared with monocular images, the geometric information calculated by the disparity of the stereo camera is richer and more accurate. In depth completion tasks, stereo camera and LiDAR are more complementary in theory, resulting denser and more accurate depth information. However, in practical applications, the effective distance range of the stereo camera is limited. It is unreliable in the environment with high occlusion and sufficient texture, such as some urban roads.

The stereo image is mainly fused at the feature level. One of the ground-breaking work comes from Park et al. [11]. First, a high-resolution dense disparity map is generated from dense stereo disparity and point clouds based on a two-stage convolutional neural network. The first stage uses LiDAR and stereo disparity to generate fusion disparity. In the second stage, the fused disparity and the RGB image are integrated in the feature space to predict the final high-resolution disparity. Finally, this high-resolution disparity is used to reconstruct a 3D scene. The limitation of this method is the requirement of the large-scale labelled stereo LiDAR datasets, while the later are currently not publicly available. LidarStereoNet [12] avoids this difficulty through unsupervised learning. The scheme does not require prior label information. This unsupervised method conducts end-to-end training utilizing image distortion, with the combination loss function of luminosity term, sparse depth term, smoothness term and plane fitting term. In addition, the feedback supervision signal makes this architecture more robust to noise point clouds and sensor misalignment.

2.2. 3D Object Detection

3D object detection task aims to position, classify and estimate the object bounding box with direction in 3D space. Currently two main kinds of object detection method are proposed: multi-stage and single-stage. The multi-stage based model is generally composed of a candidate box generation module and box regression module. In the stage of candidate box generation, all regions that may contain objects of interest are detected and proposed. In the object bounding box regression stage, the region is further screened according to the characteristics of the candidate region. But the performance of the final model is limited by each stage. A single-stage model contains only one stage, which usually simultaneously handle 2D and 3D information in a parallel manner.

2.2.1 Multi-Stage Model

(1) Multi-stage model based on 2D candidate area

This part of the model first generates 2D candidate regions based on image, which makes it possible to use existing image processing models. This method uses image object detector to generate 2D candidate regions and project them back into the 3D point cloud space, and further complete the regression detection of 3D bounding boxes in these 3D search spaces. Basically, projecting 2D candidate regions into 3D point cloud space could be classified two-folds. The first one is directly projecting the 2D bounding box to the 3D point cloud, thereby forming a cone-shaped 3D search space. The second method projects the point clouds to the image, linking the point cloud with the corresponding 2D semantic information point by point. In the point cloud, distant or occluded objects usually consist of only a small number of sparse points, which increases the difficulty of 3D bbox regression in the second stage.

1) Result-level fusion

One of the early work is F-PointNets[13], in which a 2D bounding box is generated from image, and then projected into 3D space. Du et al. [14] extended the 2D proposal generation stage into higher dimension and added a candidate box optimization stage. In the refinement stage, the background points in the seed region are filtered out by means of a model fitting based method. Finally, the filtered points are fed into the bounding box regression network, which further reduces dummy calculations on the background points. RoarNet[15] followed a similar idea, but refining by a neural network in the candidate box refinement stage. First, it uses geometric consistency search [16] to generate multiple 3D cylinder candidates based on each 2D bounding box. The method yields better performance on the KITTI test set than the existing technology, including the non-synchronized sensors case. However, these methods assume that each area contains only one object of interest, but they are not applicable to small objects such as crowded scenes and pedestrians.

One possible way to solve the above problem is replacing the 2D object detector with 2D semantic segmentation, as well as replacing the area direction candidate with per-point direction. Yang et al.[17] proposed a point-based dense object detector. A 2D semantic segmentation module is tailored to filter out background points by projecting points onto the image and associated points with 2D semantic labels. The obtained former scenic spot cloud retains contextual information and fine-grained location information. In the next stage of point-by-point candidate box generation and bounding box regression, a two-point network-based network is used for candidate box feature extraction and bounding box prediction. In order to shorten the training and reasoning time, they proposed a new judgment criterion, named PointsIoU.

2) Multi-level fusion

The neural network at present not only fuses the high dimension feature in the result level but also in the feature space, which is marked as PointFusion. PointFusion firstly explores 2D object detector to generate a 2D bounding box, which gives an initial prior clue to select the corresponding points by projecting the points onto the image. A network based on ResNet backbone and PointNet back bone combines image and point features to estimate 3D object. This method realizes the final detection of 3D targets by fusing image features and point cloud features, which facilitates the regression of 3D bounding boxes. In order to accurately locate the object in three-dimensional space, the current research work usually adopts the point fusion method. However, 2D semantics are only attached to the point clouds as an additional channel. This makes it easy to input point clouds and additional information into deep object detection architectures, such as PointRCNN [18], voxel network[19], and PointPillars [20].

(2) Multi-stage model based on 3D candidate area

Models based on 3D candidate regions directly generate 3D candidate regions from 2D or 3D data. By eliminating the conversion from 2D to 3D, the 3D search space is greatly reduced. Common methods for generating 3D candidate regions include multi-view methods and voxel-based methods. The multi-view-based method uses a bird's-eye view (BEV) of the point cloud to generate a 3D proposal. The bird's-eye view avoids perspective occlusion and retains the original information of the target's direction and x and y coordinates. These directions and x, y coordinate information are essential for 3D object detection, and the coordinate conversion between bird's-eye view and other perspectives is relatively straightforward. The model based on voxel rearrange continuous irregular data into discrete regular. It is possible to use standard convolution in 3D space. However, if the point cloud is voxelized and then transferred to the neural network, it will lose part of the spatial fine-grained 3D structure information and introduce boundary distortion. The final application result is affected by the initial voxel resolution.

MV3D is a pioneering work to generate 3D candidate box from BEV images [21]. MV3D generates 3D proposal on the LiDAR feature map in a top-down manner pixel by pixel, and then projects these 3D candidate targets to the LiDAR front view and image to extract and fuse regional features for bounding box regression. Although MV3D shows significant advantages on the latest models, there are still some shortcomings. First, when generating a 3D candidate frame on the BEV, all objects of interest are captured without interference from viewpoint and sensor. This is not applicable to small object, which can be completely obscured by other large targets. Secondly, the spatial information of small target is lost between the convolution operation. Third, the target-centered fusion combines the feature mapping of the image and the point cloud through the ROI pool, destroying the fine-grained geometric information in the fusion process. It is worth noting that in the bounding box regression stage, redundant candidate boxes will lead to repeated calculations. In order to alleviate these challenges, many methods have been proposed to improve MV3D.

In order to generalized to the small targets, AVOD (Aggregate View Object Detection Network, AVOD) first improved the proposal generation module in MV3D, using BEV point clouds and image. By adopting the auto-encoder architecture, the final feature is mapped to its original size, which alleviates the problem that small targets may be downsampled to a pixel through continuous convolution operations. The proposed feature fusion region candidate frame generation network first extracts isometric feature vectors from multiple patterns (BEV point clouds and images) through cropping and resizing operations. Then, the 1x1 convolution operation of feature space dimensionality reduction is performed, which reduces the amount of calculation and improves the operation speed. Lu et al.[22] also used an encoder-decoder-based candidate box generation network, which includes a spatial channel attention (SCA) module and an extended spatial up-sampling (ESU) module. SCA can capture multi-scale context information, while the ESU module can restore spatial information.

The ContFuse[23] network architecture proposed by Liang et al. processes the lost information through point-by-point fusion. They achieve point-by-point fusion through a continuous convolutional fusion layer, which bridges images and point cloud features of different scales in multiple stages of the network. By first extracting K nearest neighbor points for each pixel in the BEV representation of the point cloud. These points are then projected onto the image to retrieve relevant image features. Finally, according to the geometric offset between the fusion feature vector and the target pixel, the fusion feature vector is weighted, and then it is input into the neural network. However, when the LiDAR points are sparse, point fusion may not make full use of high-resolution images.

Liang et al.[24] combined multiple fusion methods such as signal-level fusion (RGB-D), feature-level fusion, multi-view and deep fusion, and further extended point-based fusion. In particular, depth completion uses image information to upsample the sparse depth map to generate dense pseudo point clouds. The upsampling process alleviates the fusion problem of the sparse point direction, which is beneficial to the learning of cross-modal representation. They believe that the integration of multiple tasks, such as ground estimation, depth completion, and 2D/3D object detection, can help the network achieve better overall performance. However, when a point in the point cloud is associated with multiple pixels in the image or another point is associated with data fusion, the problem of feature blur will occur, which is also one of the key technologies that need to be overcome.

2.2.2 One-Stage Model

Single-stage models merge the candidate region generation and bounding box regression together as one stage. This kind of models are usually efficient on computation so are more suitable for real-time applications, especially the mobile computing platforms.

Point fusion and voxel fusion was proposed by MVX Net[25]. The 2D CNN can conduct image feature extraction, and a voxel-based network can estimate the target from the fusion point cloud. The point fusion method projects point into image and extracts features before voxelization, then uses the voxel-based network for processing. The voxel fusion methods firstly group the point clouds into voxels, and then projects the non-empty voxels into the image for extracting features. These voxel features are only attached to the corresponding voxels at the later stage of the voxel network. The MVX network performs the most advanced results and outperforms other methods basing on LiDAR on KITTI. Meyer et al. extended LaserNet[26] to a multi-task and multi-model network for 3D object detection and 3D semantic segmentation relying on fusion images and LiDAR. The depth image and the front view image are processed by two CNNs in a parallel manner, and points are merged by projecting the points on the image. LaserNet predicts the point-by-point distribution of the bounding box and combines them to obtain the ultimate 3D candidate box. This algorithm guarantees better real-time performance while ensuring accuracy.

2.3. 2D/3D Semantic Segmentation

Feature-level fusion of 2D semantic segmentation network: Jaritz et al.[6] proposed an automatic encoding network based on NASNet, which can use images and sparse depth for two-dimensional semantic segmentation or depth completion. Two parallel encoders process the image and the corresponding sparse depth map respectively and then a shared decoder works on the merged data. Valada et al.[8] designed a multi-level feature-level fusion model of different depths to promote semantic segmentation. Caltagirone et al.[27] used high-sampling depth images and images for two-dimensional semantic segmentation. The dense depth image is upsampled from the sparse depth map and image generated from point clouds. The author also discussed three different fusion methods: early fusion, late fusion and cross fusion relatively. The best cross-fusion method is to process dense depth images and image data in two parallel CNN branches, and fuse these two feature maps in the final convolutional layer.

Feature-level fusion of 3D semantic segmentation network: Dai et al.[28] proposed a 3D semantic segmentation multi-view network 3DMV, which combines image semantics and point features in voxel. Image features are extracted from multiple aligned images by 2D CNNs and then back-projected into 3D space. These multi-view image features are the largest set of voxels integrated with geometry information, and then passed into a 3D CNNs network for prediction semantic for each voxel.

3) Feature-level fusion:

Early attempts at multi-mode fusion were carried out in pixels, in which 3D geometry was converted into an image or attached as an additional channel of the image. The intuition is to project the 3D geometry onto the image and transfer 2D methods to extract information. But the generated output is on the image, so it can't be regarded ideal for locating the target in 3D space Gupta et al. proposed DepthRCNN[29] which is a 2D object detection, instance and semantic segmentation architecture basing on RCNN[30]. It encodes the 3D geometry from the Microsoft Kinect camera in the RGB channels of the image. These channels are horizontal disparity, ground clearance, and gravity angle. Gupta et al. [31] extended the depth of RCNN for 3D object detection by calibrating the 3D CAD model, which significantly improved the performance. Gupta et al. also developed a new supervised knowledge transfer technique for transferring from image to other kind of modalities[32]. Schlosser et al.[33] further developed a learning representation on a 2D CNN for pedestrian detection. However, HHA data is generated by the LiDAR depth map rather than the depth camera. The author also noticed that if the fusion of RGB and HHA occurs in the deep layers of the network, better results can be obtained.

In order to mitigate the voxelization's adverse effects, Chiang et al.[34] proposed a point-based semantic segmentation framework called Unified Point-Based Framework, which can effectively represent the features of image, geometric structure and global context. The semantic segmentation network is able to extract the features from the multi-view image and project them into the 3D space for feature fusion. The fused point cloud is processed by an encoder. Local and global features are extracted and then processed by a decoder for point-wise semantic label prediction.

3D-SIS[35] is a two-stage 3D CNN that conducts voxel-level 3D instance segmentation on multi-view images and RGB-D scan data. In the 3D detection stage, the method utilizes ENet-based network to extract and downsample the features of multi-viewpoint images. Downsampling solves the feature mismatch between high-resolution image and low-resolution voxel. Then the downsampled feature is map projected back to the voxel space, and attached to the corresponding 3D geometric features. The fused feature is fed into the 3D CNN to predict the category and its bounding box pose.

2.4. 3D Object Tracking

There are currently two research directions for 3D object tracking. The tracking based on the detection framework is divided into two stages. In the first stage, the objects of interest are detected. The second stage correlates detected objects over time and generates their trajectories, which are expressed as linear programs. Frossard et al.[36] proposed an end-to-end framework, which consists of multiple independent networks that use images and point clouds at the same time. The framework implements functions such as object detection, candidate box matching and scoring, and linear optimization. In order to realize end-to-end learning, detection and matching are achieved through a deep structure model. Zhang et al.[37] proposed a sensor-agnostic framework that utilizes a loss coupling scheme for image and point cloud fusion, consisting of three stages: object detection, neighbor estimation and linear optimization. Complexer YOLO[38] is a real-time detection framework for 3D object detection and tracking that separates image and point cloud. In the 3D object detection stage, 2D semantics are embedded and merged into each point cloud. The point clouds are voxelized and fed into the 3D complex YOLO for object detection.

3. Experiments and evaluation

This section briefly introduces the performance of various LiDAR-camera fusion enhancement methods in different research directions on the KITTI dataset, and then intuitively summarizes the development along with its trend of this emerging technology. The KITTI benchmark is widely-used in autonomous driving scenarios, providing a platform to evaluate the performance of all SOTA technologies, which involves visual depth estimation, 3D object detection, 3D tracking and other tasks. KITTI contains real image data collected from scenes such as urban areas, rural areas, and highways. The entire data set consists of 389 pairs of stereo images and optical flow map, 39.2 km visual ranging sequence and more than 200k 3D-annotated images of objects.

3.1. Evaluation on depth completion method

Various methods' performance is accurately evaluated and tested through a variety of metrics including iRMSE, iMAE, RMSE, MAE. More details in Table 1. Among them, iRMSE represents the root mean square error corresponding to the inverted depth; iMAE represents the average absolute error corresponding to the inverted depth; RMSE represents the root mean square error; MAE represents the average absolute error. These four metrics all reflect the gap between the overall depth inverted by the algorithms and the true value, which could be quantitatively demonstrate the performance of the algorithm.

Table 1. Comparison results of KITTI depth completion benchmarks[39]

Method	Fusion level	Way of learning	Model	Run-time	iRMSE	iMAE	RMSE	MAE
Mono-lidar fusion	Signal-level	supervised	Sparse2Dense	0.08s	4.07	1.57	1299.85	350.32
		self-supervised	Sparse2Dense++	0.08s	2.80	1.21	814.73	249.95
	Feature-level	supervised	CSPN	1s	2.93	1.15	1019.64	279.46
		supervised	CSPN++	0.2s	2.07	0.90	743.69	209.28
		supervised	Spade-RGBsD	0.07s	2.17	0.95	917.64	234.81
		supervised	NConv-CNN	0.02s	2.60	1.03	829.98	233.26
		supervised	GuideNet	0.14s	2.25	0.99	736.24	218.83
	Multi-level	unsupervised	RGB_uidecertainty	0.02s	2.19	0.93	772.87	215.02
Stereo-lidar fusion	Feature-level	supervisd	HDE-Net	0.05s	-	-	-	-
		unsupervisedself-	LidarStereoNet	-	-	-	-	-
		supervised	LiStereo	-	2.19	1.10	832.16	283.91

3.2. Evaluation on 3D object detection methods

The distribution and types of features in the environment are often uneven, and large differences exist, which can decrease the performance of the perception system, and there are also large differences in performance evaluation. Simply considering accuracy as the standard often cannot accurately represent the capability and performance of the model. Therefore, it is a more objective method to measure the detection accuracy and overall performance of the model for different features by using Intersection-over-Union (IOU).

3.3. Evaluation on 3D object tracking methods

This section reviews the object tracking methods based on LiDAR-camera fusion and compares their performances on the KITTI. Among them, MOTA stands for multi-object tracking accuracy, which is reflected in the accuracy of determining the number of targets and related attributes of the targets. MOPT stands for multi-object tracking accuracy, which is reflected in the accuracy of determining the target position. MT stands for most tracking, that is, the proportion of tracks that meet Ground Truth's matching success only in less than 20% of the time, in all tracking targets. ML represents most of the loss, that is: the proportion of tracks that meet Ground Truth at least 80% of the time in all tracking objects. IDS represents the number of times the ID assigned by Ground Truth has changed. FRAG represents the total number of trajectory fragmentation.

Table 2. Comparison results of KITTI 3D object detection benchmarks (medium difficulty)[39]

Method		Fusion level	Fusion method	Point cloud representation	Model	Runtime	Veh	Ped	Cyclist
Multi-stage	2D candidate region	Result	N/A	Point cloud	F-PointNet	0.17s	69.79	42.15	56.12
				Point cloud	F-ConvNet	0.47s	76.39	43.38	65.07
				Voxel	PC-CNN	0.5s	73.79	-	-
				Point cloud	RoarNet	0.1s	73.04	-	53.46
				Point cloud	IPOD	0.2s	72.57	44.68	-
		Feature	Point-wise	Multi way	PointPainting	0.4s	71.70	40.97	63.78
		Multi-level	ROI-wise	Point	PointFusion	1.3s	63.00	28.04	29.42
			Point-wise	Point	SIFRNet	-	72.05	60.85	60.34
	3D candidate region	Feature	ROI-wise	2D	MV3D	0.36s	63.63	-	-
			Pixel-wise/ROI-wise	2D	AVOD-FPN	0.08s	71.76	42.27	50.55
			Pixel-wise/ROI-wise	2D	SCANet	0.17s	68.12	37.93	53.38
			Pixel-wise/ROI-wise	2D	ContFuse	0.06s	68.78	-	-
			Point-wise	2D	BEVF	-	-	-	45
			Pixel-wise						
		Multi-level	Point-wise	2D	MMF	0.08s	77.43	-	-
One-stage		Feature	Point-wise	Voxel	MVX-Net	0.15s	75.86	43.73	61.03
			Pixel-wise	2D	LaserNet++	0.04s	-		

Table.3 Comparison results of KITTI multi-object tracking benchmark (CAR)[39]

Method	Data association mode	Model	Runtime	MOTA (%)	MOTP (%)	MT (%)	ML (%)	IDS	FRAG
Detection-based	min-cost flow	DSM	0.1s	76.15	83.42	60.00	8.31	296	868
	min-cost flow	mmMOT	0.02s	84.77	85.21	73.23	2.77	284	753
	Hungarian algorithm	MOTS-fusion	0.44s	84.83	85.21	73.08	2.77	275	759
Non-detection-based	Finite random set	Complexer-YOLO	0.01s	75.70	78.46	58.00	5.08	1186	2092

4. Discussions

The future trends in the development of LiDAR-camera fusion enhancement technology in recent years can be summarized in the following aspects: 1) With the rapidly growth of feature extraction module directly based on the 3D space, the missions, derived from the traditional 2D space, will be explored in 3D space obtaining more attention. 2) Some technical trials combining multiple complementary subtasks together to achieve better overall performance have provoking many thoughts about the convenience of handling all tasks in one pipeline. 3) Signal-level to multi-level fusion. Early researchers often used signal-level fusion to convert 3D geometry to images to facility rely on the existing image processing models, while recent models are exploring deep fusion between point clouds and multi-level images in time sequences.

At the same time, how to improve the performance and efficiency of the LiDAR-camera fusion, and how to achieve intelligent perception of the surrounding environment more efficiently and stably is also a research direction that scholars are focusing on. Many scholars have also put forward innovative insights and opinions: 1) Optimizing the feature representation of the fusion data. The feature representation of fusion data is the basis for designing fusion algorithm. The current feature representation includes: additional depth information channels on images. Since this method can be processed by existed image processing models, early signal-level fusion often uses this

form of expression. However, the resolution mismatch between the high-resolution image and the low-resolution point clouds will affect the efficiency of this method. Convert image and point cloud features/signals into other data representations. Future research can explore other novel intermediate data structures instead of point clouds, voxel and lattice, such as graphs, trees, etc., to improve network performance. 2) Adding geometric constraints. Comparing with other 3D data sources, such as RGB-D data, LiDAR has a longer detection range and higher accuracy. Projecting the point clouds onto the image seems to be the most natural solution, but the sparsity of the point clouds will create holes. In addition, the use of monocular images to predict depth information and the introduction of self-supervised learning between consecutive frames are also expected to alleviate this problem. However, how to add this geometric information into the fusion process is still a problem needs to be digged.

5. Conclusion

In the field of computer vision, intelligent perception of the environment is an extremely important part of the content, and the environment perception system based on multi-source data fusion is the main development trend in the future. In this article, the latest research progress of the LiDAR-camera fusion on enhanced perception system is mainly reviewed. It covers the application and development of this technology in related fields such as depth completion, 3D object detection, 2D/3D semantic segmentation and so on. At the same time, this article shows the performance of the popular and excellent fusion algorithm models in the public dataset KITTI, and quantitatively shows the development status of LiDAR-camera fusion enhancement technology. Finally, the article summarized and discussed the general trend of the LiDAR-camera fusion enhancement technology. The current major challenges and possible future breakthroughs have also given deep consideration to some of the problems exist in this technology.

Acknowledge

The research work in this article was funded by the China Southern Power Grid Corporation's science and technology project, the super-large power grid 3D visualization management and spatial-temporal analysis technology research project (ZBKJXM20170229).

References

1. Y. Cui, R. Chen, W. Chu, L. Chen, D. Tian, and D. Cao, "Deep Learning for Image and Point Cloud Fusion in Autonomous Driving: A Review," 2020.
2. F. Mal, and S. Karaman, “Sparse-to-Dense: Depth Prediction from Sparse Depth Samples and a Single Image,” in International Conference on Robotics and Automation, 2018, pp. 4796-4803.
3. F. Ma, G. V. Cavalheiro, and S. Karaman, “Self-Supervised Sparse-to-Dense: Self-Supervised Depth Completion from LiDAR and Monocular Camera,” in International Conference on Robotics and Automation, 2019, pp. 3288-3295.
4. X. Cheng, P. Wang, and R. Yang, “Depth Estimation via Affinity Learned with Convolutional Spatial Propagation Network,” in European Conference on Computer Vision, 2018, pp. 108-125.
5. X. Cheng, P. Wang, C. Guan, and R. Yang, “CSPN++: Learning Context and Resource Aware Convolutional Spatial Propagation Networks for Depth Completion,” in National Conference on Artificial Intelligence, 2020, pp. 10615-10622.
6. M. Jaritz, R. D. Charette, E. Wirbel, X. Perrotton, and F. Nashashibi, “Sparse and Dense Data with CNNs: Depth Completion and Semantic Segmentation,” in International Conference on 3D Vision, 2018, pp. 52-60.
7. T.-H. Wang, F.-E. Wang, J.-T. Lin, Y.-H. Tsai, W.-C. Chiu, and M. Sun, “Plug-and-Play: Improve Depth Prediction via Sparse Data Propagation,” in International Conference on Robotics and Automation, 2019, pp. 5880-5886.
8. A. Valada, R. Mohan, and W. Burgard, “Self-Supervised Model Adaptation for Multimodal Semantic Segmentation,” international journal of computer vision, vol. 128, no. 5, pp. 1239-1285, 5/1/2020, 2020.
9. J. Tang, F.-P. Tian, W. Feng, J. Li, and P. Tan, "Learning Guided Convolutional Network for Depth Completion," 2019.
10. W. V. Gansbeke, D. Neven, B. D. Brabandere, and L. V. Gool, “Sparse and Noisy LiDAR Completion with RGB Guidance and Uncertainty,” in International Conference on Machine Vision, 2019, pp. 1-6.
11. K. Park, S. Kim, and K. Sohn, “High-Precision Depth Estimation Using Uncalibrated LiDAR and Stereo Fusion,” ieee transactions on intelligent transportation systems, vol. 21, no. 1, pp. 321-335, 1/1/2020, 2020.
12. X. Cheng, Y. Zhong, Y. Dai, P. Ji, and H. Li, “Noise-Aware Unsupervised Deep Lidar-Stereo Fusion,” in Computer Vision and Pattern Recognition, 2019, pp. 6339-6348.

13. C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, “Frustum PointNets for 3D Object Detection from RGB-D Data,” in Computer Vision and Pattern Recognition, 2018, pp. 918-927.
14. X. Du, M. H. Ang, S. Karaman, and D. Rus, “A General Pipeline for 3D Detection of Vehicles,” in International Conference on Robotics and Automation, 2018, pp. 3194-3200.
15. K. Shin, Y. P. Kwon, and M. Tomizuka, “RoarNet: A Robust 3D Object Detection based on RegiOn Approximation Refinement,” in IEEE Intelligent Vehicles Symposium, 2019, pp. 2510-2515.
16. A. Mousavian, D. Anguelov, J. Flynn, and J. Kosecka, “3D Bounding Box Estimation Using Deep Learning and Geometry,” in Computer Vision and Pattern Recognition, 2017, pp. 5632-5640.
17. Z. Yang, Y. Sun, S. Liu, X. Shen, and J. Jia, "IPOD: Intensive Point-based Object Detector for Point Cloud," 2018.
18. S. Shi, X. Wang, and H. Li, “PointRCNN: 3D Object Proposal Generation and Detection From Point Cloud,” in Computer Vision and Pattern Recognition, 2019, pp. 770-779.
19. Y. Zhou, and O. Tuzel, “VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection,” in Computer Vision and Pattern Recognition, 2018, pp. 4490-4499.
20. A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, “PointPillars: Fast Encoders for Object Detection From Point Clouds,” in Computer Vision and Pattern Recognition, 2019, pp. 12697-12705.
21. X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, “Multi-view 3D Object Detection Network for Autonomous Driving,” in Computer Vision and Pattern Recognition, 2017, pp. 6526-6534.
22. H. Lu, X. Chen, G. Zhang, Q. Zhou, Y. Ma, and Y. Zhao, “Scanet: Spatial-channel Attention Network for 3D Object Detection,” in International Conference on Acoustics, Speech, and Signal Processing, 2019, pp. 1992-1996.
23. M. Liang, B. Yang, S. Wang, and R. Urtasun, “Deep Continuous Fusion for Multi-Sensor 3D Object Detection,” in European Conference on Computer Vision, 2018, pp. 663-678.
24. M. Liang, B. Yang, Y. Chen, R. Hu, and R. Urtasun, “Multi-Task Multi-Sensor Fusion for 3D Object Detection,” in Computer Vision and Pattern Recognition, 2019, pp. 7345-7353.
25. V. A. Sindagi, Y. Zhou, and O. Tuzel, “MVX-Net: Multimodal VoxelNet for 3D Object Detection,” in International Conference on Robotics and Automation, 2019, pp. 7276-7282.
26. G. P. Meyer, A. Laddha, E. Kee, C. Vallespi-Gonzalez, and C. K. Wellington, “LaserNet: An Efficient Probabilistic 3D Object Detector for Autonomous Driving,” in Computer Vision and Pattern Recognition, 2019, pp. 12677-12686.
27. L. Caltagirone, M. Bellone, L. Svensson, and M. Wahde, “LIDAR–camera fusion for road detection using fully convolutional neural networks,” robotics and autonomous systems, vol. 111, pp. 125-131, 1/1/2019, 2019.
28. A. Dai, and M. Nießner, “3DMV: Joint 3D-Multi-View Prediction for 3D Semantic Scene Segmentation,” in European Conference on Computer Vision, 2018, pp. 458-474.
29. S. Gupta, R. B. Girshick, P. A. Arbeláez, and J. Malik, “Learning Rich Features from RGB-D Images for Object Detection and Segmentation,” in European Conference on Computer Vision, 2014, pp. 345-360.
30. R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation,” in Computer Vision and Pattern Recognition, 2014, pp. 580-587.
31. S. Gupta, P. Arbelaez, R. Girshick, and J. Malik, “Aligning 3D models to RGB-D images of cluttered scenes,” in Computer Vision and Pattern Recognition, 2015, pp. 4731-4740.
32. S. Gupta, J. Hoffman, and J. Malik, “Cross Modal Distillation for Supervision Transfer,” in Computer Vision and Pattern Recognition, 2016, pp. 2827-2836.
33. J. Schlosser, C. K. Chow, and Z. Kira, “Fusing LIDAR and images for pedestrian detection using convolutional neural networks,” in International Conference on Robotics and Automation, 2016, pp. 2198-2205.
34. H.-Y. Chiang, Y.-L. Lin, Y.-C. Liu, and W. H. Hsu, “A Unified Point-Based Framework for 3D Segmentation,” in International Conference on 3D Vision, 2019, pp. 155-163.
35. J. Hou, A. Dai, and M. NieBner, “3D-SIS: 3D Semantic Instance Segmentation of RGB-D Scans,” in Computer Vision and Pattern Recognition, 2019, pp. 4421-4430.
36. D. Frossard, and R. Urtasun, “End-to-end Learning of Multi-sensor 3D Tracking by Detection,” in International Conference on Robotics and Automation, 2018, pp. 635-642.
37. W. Zhang, H. Zhou, S. Sun, Z. Wang, J. Shi, and C. C. Loy, “Robust Multi-Modality Multi-Object Tracking,” in International Conference on Computer Vision, 2019, pp. 2365-2374.
38. M. Simon, K. Amende, A. Kraus, J. Honer, T. Samann, H. Kaulbersch, S. Milz, and H. M. Gross, “Complexer-YOLO: Real-Time 3D Object Detection and Tracking on Semantic Point Clouds,” in Computer Vision and Pattern Recognition, 2019, pp. 1-10.
39. A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? The KITTI vision benchmark suite,” in Computer Vision and Pattern Recognition, 2012, pp. 3354-3361.