BEVStereo: Enhancing Depth Estimation in Multi-View 3D Object Detection with Temporal Stereo

Yinhao Li[1, 3], **Han Bao**[2, 3], **Zheng Ge**[4], **Jinrong Yang**[5], **Jianjian Sun**[4], **Zeming Li**[4]
[1]Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, CAS
[2]State Key Lab of Processors, Institute of Computing Technology, CAS
[3]University of Chinese Academy of Sciences
[4]MEGVII Technology
[5]Huazhong University of Science and Technology
liyinhao20@mails.ucas.edu.cn, baohan20s@ict.ac.cn
{gezheng, yangjinrong, sunjianjian, lizeming}@megvii.com

Abstract

Restricted by the ability of depth perception, all Multi-view 3D object detection methods fall into the bottleneck of depth accuracy. By constructing temporal stereo, depth estimation is quite reliable in indoor scenarios. However, there are two difficulties in directly integrating temporal stereo into outdoor multi-view 3D object detectors: 1) The construction of temporal stereos for all views results in high computing costs. 2) Unable to adapt to challenging outdoor scenarios. In this study, we propose an effective method for creating temporal stereo by dynamically determining the center and range of the temporal stereo. The most confident center is found using the EM algorithm. Numerous experiments on nuScenes have shown the BEVStereo's ability to deal with complex outdoor scenarios that other stereo-based methods are unable to handle. For the first time, a stereo-based approach shows superiority in scenarios like a static ego vehicle and moving objects. BEVStereo achieves the new state-of-the-art in the camera-only track of nuScenes dataset while maintaining memory efficiency. Codes have been released[1].

Introduction

Due to its stability and inexpensive cost, multi-view 3D object detection has received a lot of interest lately. The field of 3D object detection has seen significant progress with numerous camera-based techniques (Wang et al. 2022b; Huang et al. 2021; Liu et al. 2022a; Li et al. 2022b; Huang and Huang 2022; Liu et al. 2022b; Li et al. 2022a). However, there is still a substantial gap between them and LiDAR-based approaches. When examining camera-based approaches, it is obvious to see that depth accuracy is still the main limitation of these methods.

Delving into depth-based multi-view object detection approaches, their depth module is a mono-depth estimation module due to the rarity of multi-view camera overlap. Dijk et.al (Dijk and Croon 2019) shows that monocular depth estimation is based on the object's connection with the ground and contextual information. These two cues let neural networks understand depth to some extent, but this degree of precision is hardly adequate for 3D object detection.

[1]https://github.com/Megvii-BaseDetection/BEVStereo

Given that monocular depth estimation has reached its limit and that time series input images are available in autonomous driving scenarios, it makes sense to use temporal stereo approaches to multi-view 3D object detection. However, if we incorporate the temporal stereo method into the multi-view 3D detector, there are two limitations:

1. **Large memory cost.** When we replace the depth module in BEVDepth with a basic temporal stereo method (Yao et al. 2018), the memory cost grows to 3.5 times that of BEVDepth despite bringing a 1.6 percent promotion on NDS, making it tremendous burden to apply it to a detection task;
2. **Unable to tackle complex outdoor scenarios.** As discussed in DFM (Wang, Pang, and Lin 2022), temporal stereo approaches are unable to handle situations like a static ego vehicle and moving objects. However, over 10% of the frames'ego vehicles are static in nuScenes, while approximately 25% of the objects are moving. These two circumstances are also crucial to the security of an autonomous system in real-world autonomous scenarios.

In this study, we present BEVStereo, a stereo-based multi-view 3D object detector that uses our dynamic temporal stereo method to improve memory efficiency while being able to adapt to challenging outdoor conditions. Reviewing stereo-based approaches (Wang, Pang, and Lin 2022; Wang et al. 2022a), it is clear that the majority of the computational memory cost is associated with constructing stereo. How to reduce the cost of building stereo is the key to saving memory. To achieve this, we apply a dynamic way to build temporal stereo. As opposed to employing all candidates along the depth axis, we construct it using the projected depth center (μ) and depth range (σ). This drastically reduces the number of candidates while eliminating the need for us to manually select sample density. Sharing the similar thoughts, MaGNet (Bae, Budvytis, and Cipolla 2022) achieves great success in indoor scenarios. However, its iteration mechanism is unable to deal with complex outdoor situations and introducing learnable parameters to update μ and σ also brings addition computational cost. Therefore, we apply the EM approach to update μ and σ rather than using addition network. Meanwhile, to avoid unnecessary situations brought

by circle NMS, we design a new NMS method that takes objects' size into account.

In conclusion, the following are our main contributions.

- We propose BEVStereo, a multi-view 3D object detector using temporal stereo to acquire a more reliable depth estimation. By applying the dynamic temporal stereo technique, BEVStereo saves a lot of memory compared to other stereo-based methods while adapting to complex outdoor scenarios that these methods can not handle.
- We design a new NMS method (namely size-aware circle NMS) which takes object's size into account when applying circle NMS.
- Under the same settings, BEVStereo improves mAP and NDS by 1.7% and 1.7%.

Related Work

Single-view 3D Object Detection

Many approaches have made their effort on predicting objects directly from single images. For the purpose of 3D object detection, Cai et al. (Cai et al. 2020) calculates the depth of the objects by integrating the height of the objects in the image with the height of the objects in the real world. Based on FCOS (Tian et al. 2019), FCOS3D (Wang et al. 2021b) extends it to 3D object detection by changing the classification branch and regression branch which predicts 2D and 3D attributes at the same time. M3D-RPN (Brazil and Liu 2019) treats mono-view 3D object detection task as a standalone 3D region proposal network, narrowing the gap between LiDAR-based approaches and camera-based methods. D^4LCN (Ding et al. 2020) replaces 2D depth map with pseudo LiDAR representation to better present 3D structure. DFM (Wang, Pang, and Lin 2022) integrates temporal stereo to mono-view 3D object recognition, improving the quality of depth estimation while minimizing the negative effects of difficult situations that temporal stereo is unable to handle.

Multi-view 3D Object Detection

Current multi-view 3D object detectors can be divided into two schemas: LSS-based (Philion and Fidler 2020) schema and transformer-based schema.

BEVDet (Huang et al. 2021) is the first study that combines LSS and LiDAR detection head which uses LSS to extract BEV feature and uses LiDAR detection head to propose 3D bounding boxes. By introducing previous frames, BEVDet4D (Huang and Huang 2022) acquires the ability of velocity prediction. To reduce memory usage, M^2BEV (Xie et al. 2022) decreases the learnable parameters and achieves high efficiency on both inference speed and memory usage. BEVDepth (Li et al. 2022a) uses LiDAR to generate depth GT for supervision and encodes camera intrinsic and extrinsic parameters to enhance the model's ability of depth perception.

DETR3D (Wang et al. 2022b) extends DETR (Carion et al. 2020) into 3D space, using transformer to generate 3D bounding boxes. Based on DETR, PETR (Liu et al. 2022a) and PETRV2 (Liu et al. 2022b) adds position embedding onto it. BEVFormer (Li et al. 2022b) uses deformable transformer to extract features from images and uses cross attention to link the feature between frames for velocity prediction.

Depth Estimation

Based on the number of images used for depth estimation, depth estimation methods can be divided into single-view depth estimation and multi-view depth estimation.

Although predicting depth from a single image is obviously ill-posed, it is still possible to estimate some of the depth of the objects by using the context as a signal. Therefore, many approaches (Bhat, Alhashim, and Wonka 2021; Eigen and Fergus 2015; Eigen, Puhrsch, and Fergus 2014a; Fu et al. 2018) use CNN method to predict depth.

For the task of multi-view depth estimation, Constructing cost volume is an effective way to predict depth. MVSNet (Yao et al. 2018) is the first research that uses cost volume for depth estimation. RMVSNet (Yao et al. 2019) reduces memory cost by introducing GRU module. MVSCRF (Xue et al. 2019) adds CRF module onto MVSNet. PointMVSNet (Chen et al. 2019) uses point algorithm to optimize the regression of depth estimation. Cascade MVSNet (Gu et al. 2020) uses cascade structure, making it able to use large depth range and a small amount of depth intervals. Fast-MVSNet (Yu and Gao 2020) uses sparse cost volume and Gauss-Newton layer to speed up MVSNet. Wang et al. (Wang et al. 2021a) use adaptive patchmatch and multi-scale fusion to achieve good performance while mataining high efficiency. Bae et al. (Bae, Budvytis, and Cipolla 2022) introduce MaGNet to better fuse single-view depth estimation and multi-view depth estimation.

Method

BEVStereo is a stereo-based multi-view 3D object detector. By applying our temporal stereo technique, it is able to handle complex outdoor scenarios while maintaining memory efficiency. We also propose a size-aware circle NMS approach to improve the proposal suppression process.

Preliminary Knowledge

Multi-view 3D object detection LSS-based (Philion and Fidler 2020) multi-view 3D object detectors currently include four components: an image encoder to extract the image features, a depth module to generate depth and context, then outer product them to get point features, a view transformer to convert the feature from camera view to the BEV view, and a 3D detection head to propose the final 3D bounding boxes.

Temporal stereo methods to predict depth MVS-based (Yao et al. 2018) methods predict depth by constructing cost volume. For every pixel on the reference feature, they initially put forth a number of candidates along the depth axis. They next convert these candidates from reference to source using a homography warping operation in order to retrieve the relevant source feature and create the

cost volume. After cost volume is constructed. For the purpose of predicting the confidence of each depth candidate, 3D convolution is performed to regularize the cost volume.

Dynamic Temporal Stereo

Based on BEVDepth (Li et al. 2022a), BEVStereo changes the way of generating depth prediction. Instead of predicting depth from a single image, BEVStereo predicts both depth from single feature (mono depth) and depth from temporal stereo (stereo depth). Additionally, Weight Net is used to create a weight map that will be applied on stereo depth. Mono depth and weighted stereo depth are combined to get the final depth. Our framework overview is illustrated in Fig. 1.

Depth Module Our Depth Module simultaneously predicts mono depth, μ, σ and context. After iterating μ and σ by our EM method, they are used to generate the stereo depth. The process of iterating μ and σ is illustrated in Fig. 2.

We choose to estimate μ and σ, which stand for the center and range of the sampling range to construct cost volume. When compared to the conventional method of splitting bins along the depth dimension, our method can dynamically change the search area while also lowering the number of candidates. After estimating μ and σ, we may obtain the depth of every candidate for each pixel. These candidates are used for homography warping operation to fetch the feature from source frame, as illustrated in Equ. 1, where P denotes the coordinate of the point, D denotes the depth of the candidate, src denotes source frame, ref denotes reference frame, $M_{ref2src}$ denotes the transformation matrix from the reference frame to source frame and K denotes the intrinsic matrix. The reference feature and the warpped source feature are used to construct cost volume. Similarity Net is followed to predict the confidence score of all candidates.

$$P_{src}[u\cdot z, v\cdot z, z] = K \times M_{ref2src} \times K^{-1} \times (D \cdot P_{ref}[u, v, 1]) \tag{1}$$

Inspired by the EM algorithm, We attempt to make the expectation of μ closer to the depth gt during the iteration process. Since we compute each point's confidence after sampling a number of points close to μ, it is only natural that we use this knowledge to further our objectives. As a result, we update μ using the weight sum method, which causes μ to become the expectation of the sample points for each iteration. The update rule is illustrated in Eq. 2, where D_i denotes the depth of the ith candidate and P_i denotes the probability of the ith candidate. When facing cases like static ego vehicle and moving objects, all candidates share the same low probability since it is hard to find the best match point on the source feature, μ is able to maintain its value by using the weight sum technique. For other scenarios, the value of μ will approach the true depth value in the process of iteration. Surprisingly, we discover that when μ and mono depth are trained together, the quality of initial μ is also enhanced under the direction of mono depth. Therefore, in all kinds of scenarios, our dynamic temporal stereo approach can improve depth prediction. As μ is being updated in the process of iteration, it is also critical to find the suitable σ to set the searching range. In accordance with existing information, the searching range should be reduced when the confidence of μ is high and expanded when it is low, we update σ following Equ. 3 where P_μ denotes the confidence of μ. Without introducing any learnable parameters both μ and σ will be adapted to the change of camera positions and the search range is optimized during iteration.

To prevent the scenario where the projected μ is far from the depth gt, making it difficult to optimize μ during iteration. we divide the depth into different ranges and use our iteration technique in each split range. After the iteration process is finished, the depth map is generated following Equ. 4 where P denotes the computed depth confidence and D denotes the depth of the split bins along the depth axis for each pixel.

$$\mu = \sum_{i=1}^{n} D_i \cdot P_i, \tag{2}$$

$$\sigma_{new} = \frac{\sigma_{old}}{2 \cdot P_\mu}, \tag{3}$$

$$P = exp(-\frac{1}{2} \cdot (\frac{D-\mu}{\sqrt{\sigma}})^2). \tag{4}$$

Weight Net Even while the temporal stereo is capable of accurately predicting depth, there are still some areas where it is unreliable because some reference feature points do not correlate to positions on source feature. Therefore, we introduce Weight Net to better combine mono depth and stereo depth. To do this, we apply the same homography warping operation to fetch the mono depth of the source frame, using μ as the depth. A similarity net is then applied to the warped mono depth from the source frame and the mono depth from the reference frame to construct the weight map.

Size-aware Circle NMS

The distance between the centers of two bounding boxes is used by circle NMS (Yin, Zhou, and Krahenbuhl 2021) function as a criterion for suppression. Circle NMS achieves excellent efficiency and good performance by bypassing the difficult process of computing rotated IoU of bouding boxes. However, ignoring the size of boxes will result in two drawbacks as illustrated in Fig. 3: 1) No matter how closely the boxes overlap, the NMS algorithm yields the same output as long as the box centers are fixed. 2) When boxes are placed differently, boxes with 0 IoU may be removed while boxes with high IoU are kept.

We propose size-aware circle NMS, which avoids computing rotated IoU while taking into consideration the size of the boxes. We separate the distance of two bounding boxes' centers into x axis and y axis. We use x_{thre} and y_{thre} as threholds of x axis and y axis, which are computed following Equ. 5 and Equ. 6, where θ denotes the orientation, w denotes the hyper parameter of scale factor, d_x denotes the length of the box and d_y denotes the width of the box. The box will be suppressed when the distance in x axis is smaller than x_{thre} and distance in y axis is smaller than y_{thre}. By applying size-aware circle NMS, the blue box with a lower score will be suppressed in scenarios like the left portion of

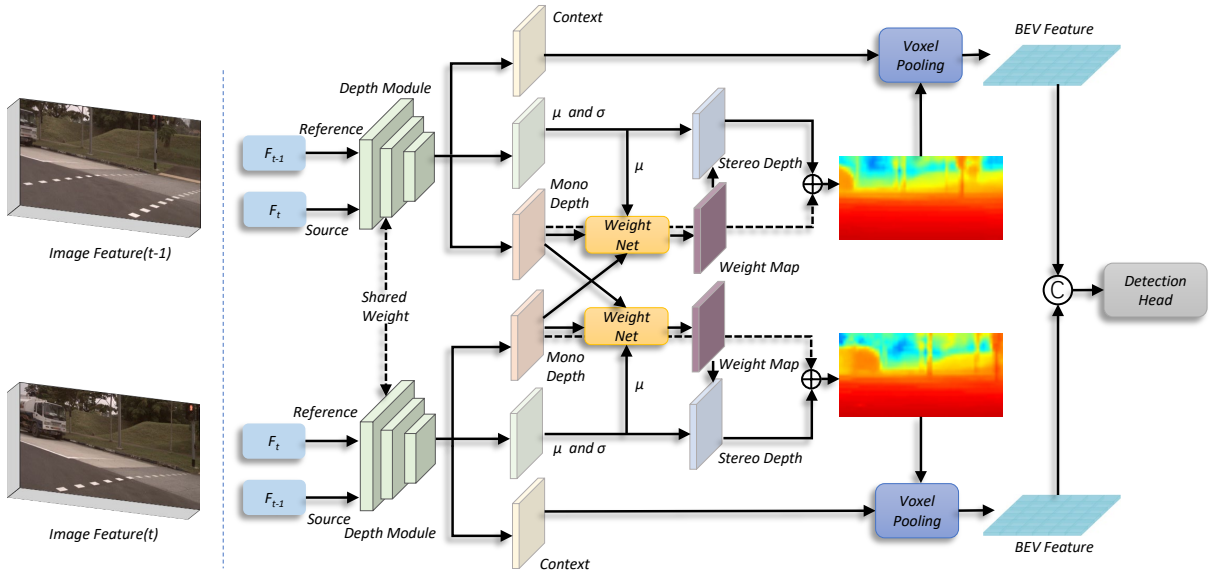


Figure 1: Framework of BEVStereo. The Depth Module uses the image feature of the reference frame and source frame as input to generate μ, σ, context, and mono depth. Stereo depth is produced using μ and σ. Weight Net uses μ and the mono depth of two frames to create a weight map that is applied to the stereo depth. Mono depth and weighted stereo depth are accumulated together to create the final depth. BEV Feature is produced when context is combined with it and is used by the detecting head.

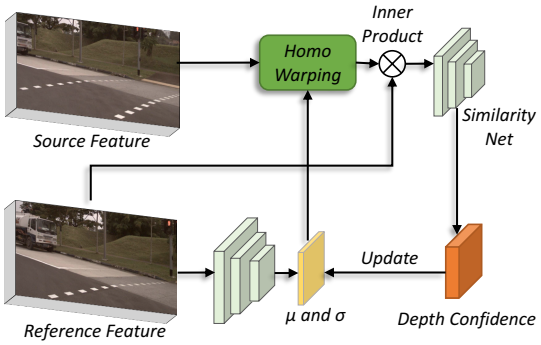


Figure 2: Iterative process of μ and σ. The initial μ and σ are generated using feature of the reference frame as input. For each round of iteration, μ and σ are used for homography warping to fetch the source feature. Similarity Net takes the inner product results of warpped source feature and reference feature as input to generate depth confidence which is used to update μ and σ.

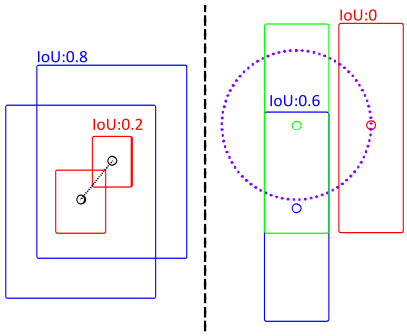


Figure 3: Drawbacks of circle NMS. In the left part of the figure, despite having distinct IoUs, the blue boxes and red boxes share the same center distance as long as their centers coincide. In the right part of the figure, when the green box has the highest score, the red box is more likely to be suppressed since its center is closer to the green box's center which goes against our common sense.

Fig 3 because it has a greater x_{thre} and y_{thre}. The blue box will be suppressed in scenarios like the right portion of Fig. 3 because the distances in the x and y axes are more likely to be smaller than x_thre and y_thre in the mean time.

$$x_{thre} = w \cdot (sin\theta_1 \cdot d_{x1} + cos\theta_1 \cdot d_{y1} + sin\theta_2 \cdot d_{x2} + cos\theta_2 \cdot d_{y2}). \quad (5)$$

$$y_{thre} = w \cdot (sin\theta_1 \times d_{y1} + cos\theta_1 \cdot d_{x1} + sin\theta_2 \cdot d_{y2} + cos\theta_2 \cdot d_{x2}). \quad (6)$$

Experiment

In this section, we first describe the experimental settings that we employ before going into the specifics of our implementation strategy. Experiments involving heavy ablation are carried out to confirm the efficacy and validity of BEVStereo.

Experimental Settings

Dataset and evaluation metrics We decide to run our experiments on the nuScenes (Caesar et al. 2020) dataset. In the case of image data, the key frame image and the fur-

Method	WN	mAP↑	mATE↓	mASE↓	mAVE↓	NDS↑
BEVDepth		32.7	70.1	27.7	55.8	43.3
BEVStereo		34.5	66.5	27.9	55.0	44.7
BEVStereo	✓	34.6	65.3	27.4	51.6	45.3

Table 1: Detection results on the nuScenes *val* set. WN denotes Weight Net.

Method	SILog↓	Abs Rel↓	Sq Rel↓	log10↓	RMSE↓
BEVDepth	21.74	0.155	1.223	0.060	5.269
BEVStereo	21.74	0.152	1.206	0.059	5.246

Table 2: Evaluation of depth prediction on the nuScenes *val* set.

thest sweep connected to it are used, whereas in the case of LiDAR data, only the key frame data is used. We assess the results of our method using detection and depth metrics. Memory usage is also used to assess the effectiveness of our method. We follow the established evaluation procedures for the depth estimation task (Eigen, Puhrsch, and Fergus 2014b), reporting scale invariant logarithmic error (SILog), mean absolute relative error (Abs Rel), mean squared relative error (Sq Rel), mean log10 error (log10), and root mean squared error (RMSE) to assess our approach.

Implementation details We implement BEVStereo based on BEVDepth (Li et al. 2022a). The feature map we employ for building the cost volume has a downsampling rate of 4 while the depth feature's final form remains unchanged. The MVS (Yao et al. 2018) approach is applied to replace the depth module in BEVDepth with the same input resolution and output resolution in order to fairly demonstrate the effectiveness of our method. The learning rate is set to 2e-4, the EMA technique is also used, and AdamW (Loshchilov and Hutter 2017) is used as the optimizer. During training, we use both image and BEV data augmentation.

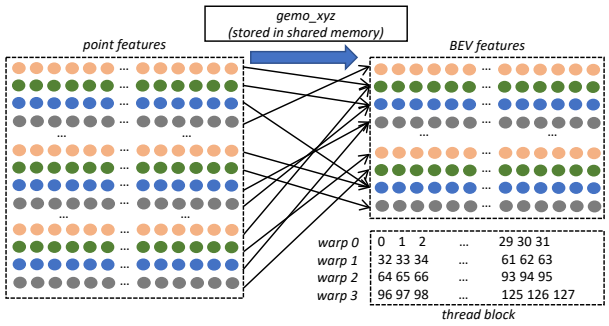


Figure 4: Thread mapping of point features to BEV features. Based on the point coordinates, the point features are atomically accumulated into the corresponding BEV features. Each thread block loads the point coordinates it is responsible for into the shared memory.

Method	Iter	TH=0.5	TH=1	TH=2	TH=4
BEVDepth		28.32	46.10	60.37	71.18
BEVDepth + MVS		27.67	46.40	59.99	71.26
BEVStereo		29.79	49.26	61.53	72.10
BEVStereo	✓	29.40	48.97	61.53	72.27

Table 3: Recall results on the nuScenes *val* set. Only boxes with velocity higher than 1m/s are maintained for analysis. BEVDepth + MVS denotes replacing depth module in BEVDepth with MVS approach. Different thresholds are utilized depending on the distance between boxes' center. Iter denotes whether to iterate μ during the inference stage.

Method	TH=0.5	TH=1	TH=2	TH=4
BEVDepth	32.80	53.58	70.00	80.89
BEVDepth + MVS	33.61	54.23	69.89	80.57
BEVStereo	33.90	54.79	70.51	81.01

Table 4: Recall results on the nuScenes *val* set. Only boxes with velocity lower than 1m/s are maintained for analysis.

Analysis

We perform numerous experiments to examine the mechanism of BEVStereo in order to better understand how it works. We choose BEVDepth as baseline, we also implement MVSNet on BEVDepth as a comparison to show the distinct benefit that BEVStereo provides, detection results and recall results are used for comparison.

Memory analysis We keep track of memory usage and detection results to demonstrate how effectively we use our memory. We also monitor the same metrics for the MVS-based approach for fair comparison.

As illustrated in Tab. 6, BEVStereo increases the metrics on mAP, mATE, and NDS considerably at the expense of adding little memory consumption. When compared to using MVS on BEVDepth, BEVStereo considerably reduces memory usage while boosting performance.

Performance analysis To begin with, we demonstrate the performance comparison under the nuScenes evaluation metrics. As shown in Tab. 1, Our BEVStereo outperforms BEVDepth on mAP, mATE and NDS. Tab. 2 shows that the accuracy of depth estimation is improved by introducing our design.

We assess the performance of BEVStereo under challenging conditions such as moving objects, and static ego vehi-

Method	Iter	mAP↑	mATE↓	NDS↑
BEVDepth		32.73	73.47	44.14
BEVDepth + MVS		31.55	78.06	43.21
BEVStereo		33.12	63.01	46.68
BEVStereo	✓	33.76	63.49	46.76

Table 5: Detection results on the nuScenes *val* set. Only frames with ego vehicles moving at speeds less than 1 m/s are employed for evaluation.

Method	Memory	mAP↑	mATE↓	NDS↑
BEVDepth	6.49GB	32.7	70.1	43.3
BEVDepth + MVS	24.04GB	34.7	67.1	44.9
BEVStereo	8.01GB	34.6	65.3	45.3

Table 6: Memory usage and detection results of BEVDepth, BEVDepth with MVS and BEVStereo.

num_iter	mAP↑	mATE↓	NDS↑
0	32.7	67.4	43.9
1	33.1	67.0	44.2
2	34.1	65.9	45.0
3	34.6	65.3	45.3

Table 7: Detection results on the nuScenes *val* set. num_iter denotes the number of iterations for μ.

Method	CA	mAP↑	mATE↓	NDS↑
circlenms		34.6	65.3	45.3
circle-nms	✓	24.9	80.6	38.0
size-aware-circlenms		35.1	64.7	45.6
size-aware-circlenms	✓	33.3	64.1	45.0

Table 8: Detection results on the nuScenes *val* set. CA denotes class-agnostic. All results are conducted under the best hyper parameters.

cles in order to show how well it adapts to complicated outdoor environments. Tab. 3 demonstrates that BEVStereo still has the ability to improve performance even while MVS approach fails when dealing with moving objects. The static objects, which make up the majority of MVS schema's contribution, are also used to evaluate our method. As shown in Tab. 4, BEVStereo's ability of perceiving static objects is even higher than BEVDepth with MVS. We choose frames whose ego vehicle has a low velocity for evaluation since MVS cannot handle situations when this occurs. As can be seen in Tab. 5, BEVStereo still improves performance even when MVS fails in these conditions. It is important to note that BEVStereo still produces the similar results when faced with circumstances like moving objects and static ego vehicles if μ is not updated during the inference step. This demonstrates that our schema is capable of guiding the Depth Module to produce better μ and maintaining the initial prediction of μ in the face of these eventualities. It is worth noting that we also conduct experiments in Tab. 3 and Tab. 4 without the Weight Net, and the results are comparable to those with the Weight Net, demonstrating that the Weight Net is not the primary reason that BEVStereo can handle moving objects and static ego vehicle scenarios.

Ablation Study

Iteration of μ and σ We conduct various experiments during the inference stage by modifying the number of iterations just to verify the function of iterating μ and σ. As illustrated in Tab. 7, the detection results improve as the number of iterations grows.

Method	Resolution	Modality	mAP↑	NDS↑
CenterPoint-Voxel	-	L	56.4	64.8
CenterPoint-Pillar	-	L	50.3	60.2
FCOS3D	900×1600	C	29.5	37.2
DETR3D	900×1600	C	30.3	37.4
BEVDet-R50	256×704	C	28.6	37.2
BEVDet-Base	512×1408	C	34.9	41.7
PETR-R50	384×1056	C	31.3	38.1
PETR-R101	512×1408	C	35.7	42.1
PETR-Tiny	512×1408	C	36.1	43.1
BEVDet4D-Tiny	256×704	C	32.3	45.3
BEVDet4D-Base	640×1600	C	39.6	51.5
BEVFormer-S	-	C	37.5	44.8
BEVFormer-R101-DCN	900×1600	C	41.6	51.7
BEVDepth-R50	256×704	C	35.9	48.0
BEVDepth-ConvNext	512×1408	C	46.2	55.8
BEVStereo-R50	256×704	C	37.6	49.7
BEVStereo-ConvNext	512×1408	C	47.8	57.5

Table 9: Comparison on the nuScenes *val* set. L denotes LiDAR and C denotes camera.

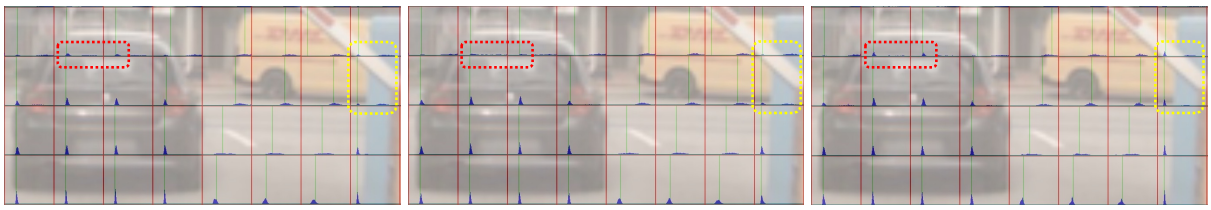
Weight Net We run the experiment under identical conditions without Weight Net to assess its validity. Weight Net promotes the detection results, as shown in Tab. 1.

Size-aware Circle NMS We compare BEVStereo with the size-aware circle NMS to BEVStereo with the conventional circle NMS as our baseline. They are subjected to class-aware and class-agnostic procedures in order to test the validity of size-aware circle NMS.

As shown in Tab. 8, our size-aware circle NMS improves on the matrices of mAP, mATE, and NDS when using class-aware NMS. The traditional distance-based circle NMS has completely lost its capacity to suppress under class-agnostic circumstance, while our size-aware circle NMS continues to function well.

Efficient Voxel Pooling v2 In the previous version of Efficient Voxel Pooling (Li et al. 2022a), threads within the same warp access memory discontinuously, leading to more memory transactions, which results in poor performance. We enhance Efficient Voxel Pooling by improving the way threads are mapped, as illustrated in Fig. 4. For each block, we employ 32 and 4 threads on the x and y axes. First, 128 point coordinates are loaded into shared memory by all the threads in one block. Then, one point feature at a time is processed by each warp. According to the point coordinates, the point feature is atomically accumulated to the matching BEV feature. The 128 point features are processed round robin by four warps in a block till they are finished. In this manner, performance-limiting memory transactions from the L2 cache and global memory are diminished.

We compare the latency of Efficient Voxel Pooling v1 and Efficient Voxel Pooling v2 using various resolutions. Efficient Voxel Pooling v2 is able to reduce the latency up to 40%.



(a) Baseline (b) Baseline + MVS (c) BEVStereo

Figure 5: Visualization of depth prediction. The blue area is the distribution of depth prediction, while the green line represents the depth GT produced by the point cloud. The red dotted boxes denotes the promotion of depth prediction on moving objects and the yellow dotted boxes denotes the the promotion of depth prediction on static objects.

Figure 6: Visualization of detection results. The blue dotted rectangle designates the object recognized by our approach is more accurate on localization, while the red dotted circle designates the object detected by BEVStereo but missed by the baseline.

Visualization

As illustrated in Fig. 5, we can find that BEVStereo has the ability to promote the accuracy of depth estimation on both moving and static objects. We also visualize the detection results, as shown in Fig. 6 which also demonstrates the performance promotion brought by BEVStereo.

Benchmark Result

We compare BEVStereo with other state-of-the-art methods (Yin, Zhou, and Krahenbuhl 2021; Wang et al. 2021b, 2022b; Huang et al. 2021; Liu et al. 2022a; Huang and Huang 2022; Li et al. 2022b,a). As shown in Tab. 9, BEVStereo achieves the highest score of camera-based methods on both mAP and NDS.

Conclusion

In this paper, a novel multi-view object detector is proposed, namely BEVStereo. BEVStereo improves performance without significantly increasing memory usage by applying dynamic temporal stereo technique to create temporal stereo. Some complex scenarios that other stereo-based approaches cannot handle can be resolved by our method. In addition, we propose size-aware circle NMS, which takes the size of boxes into account while avoiding the laborious computation of rotated IoU. Under both class-aware and class-agnostic circumstances, our size-aware circle NMS performs satisfactorily. Last but not least, we present Efficient Voxel Pooling v2, which speeds up voxel pooling by improving the efficiency of memory accesses.

References

Bae, G.; Budvytis, I.; and Cipolla, R. 2022. Multi-View Depth Estimation by Fusing Single-View Depth Probability with Multi-View Geometry. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2842–2851.

Bhat, S. F.; Alhashim, I.; and Wonka, P. 2021. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 4009–4018.

Brazil, G.; and Liu, X. 2019. M3d-rpn: Monocular 3d region proposal network for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9287–9296.

Caesar, H.; Bankiti, V.; Lang, A. H.; Vora, S.; Liong, V. E.; Xu, Q.; Krishnan, A.; Pan, Y.; Baldan, G.; and Beijbom, O. 2020. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11621–11631.

Cai, Y.; Li, B.; Jiao, Z.; Li, H.; Zeng, X.; and Wang, X. 2020. Monocular 3d object detection with decoupled structured polygon estimation and height-guided depth estimation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 10478–10485.

Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; and Zagoruyko, S. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, 213–229. Springer.

Chen, R.; Han, S.; Xu, J.; and Su, H. 2019. Point-based multi-view stereo network. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1538–1547.

Dijk, T. v.; and Croon, G. d. 2019. How do neural networks see depth in single images? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2183–2191.

Ding, M.; Huo, Y.; Yi, H.; Wang, Z.; Shi, J.; Lu, Z.; and Luo, P. 2020. Learning depth-guided convolutions for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 1000–1001.

Eigen, D.; and Fergus, R. 2015. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*, 2650–2658.

Eigen, D.; Puhrsch, C.; and Fergus, R. 2014a. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27.

Eigen, D.; Puhrsch, C.; and Fergus, R. 2014b. Depth map prediction from a single image using a multi-scale deep network. *Advances in neural information processing systems*, 27.

Fu, H.; Gong, M.; Wang, C.; Batmanghelich, K.; and Tao, D. 2018. Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2002–2011.

Gu, X.; Fan, Z.; Zhu, S.; Dai, Z.; Tan, F.; and Tan, P. 2020. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2495–2504.

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

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

Li, Y.; Ge, Z.; Yu, G.; Yang, J.; Wang, Z.; Shi, Y.; Sun, J.; and Li, Z. 2022a. BEVDepth: Acquisition of Reliable Depth for Multi-view 3D Object Detection. *arXiv preprint arXiv:2206.10092*.

Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Yu, Q.; and Dai, J. 2022b. BEVFormer: Learning Bird's-Eye-View Representation from Multi-Camera Images via Spatiotemporal Transformers. *arXiv preprint arXiv:2203.17270*.

Liu, Y.; Wang, T.; Zhang, X.; and Sun, J. 2022a. Petr: Position embedding transformation for multi-view 3d object detection. *arXiv preprint arXiv:2203.05625*.

Liu, Y.; Yan, J.; Jia, F.; Li, S.; Gao, Q.; Wang, T.; Zhang, X.; and Sun, J. 2022b. PETRv2: A Unified Framework for 3D Perception from Multi-Camera Images. *arXiv preprint arXiv:2206.01256*.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Philion, J.; and Fidler, S. 2020. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, 194–210. Springer.

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

Wang, F.; Galliani, S.; Vogel, C.; Speciale, P.; and Pollefeys, M. 2021a. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14194–14203.

Wang, T.; Lian, Q.; Zhu, C.; Zhu, X.; and Zhang, W. 2022a. MV-FCOS3D++: Multi-View Camera-Only 4D Object Detection with Pretrained Monocular Backbones. *arXiv preprint arXiv:2207.12716*.

Wang, T.; Pang, J.; and Lin, D. 2022. Monocular 3D Object Detection with Depth from Motion. *arXiv preprint arXiv:2207.12988*.

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

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

Xie, E.; Yu, Z.; Zhou, D.; Philion, J.; Anandkumar, A.; Fidler, S.; Luo, P.; and Alvarez, J. M. 2022. Mˆ 2BEV: Multi-Camera Joint 3D Detection and Segmentation with Unified Birds-Eye View Representation. *arXiv preprint arXiv:2204.05088*.

Xue, Y.; Chen, J.; Wan, W.; Huang, Y.; Yu, C.; Li, T.; and Bao, J. 2019. Mvscrf: Learning multi-view stereo with conditional random fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4312–4321.

Yao, Y.; Luo, Z.; Li, S.; Fang, T.; and Quan, L. 2018. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 767–783.

Yao, Y.; Luo, Z.; Li, S.; Shen, T.; Fang, T.; and Quan, L. 2019. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5525–5534.

Yin, T.; Zhou, X.; and Krahenbuhl, P. 2021. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11784–11793.

Yu, Z.; and Gao, S. 2020. Fast-mvsnet: Sparse-to-dense multi-view stereo with learned propagation and gauss-newton refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 1949–1958.