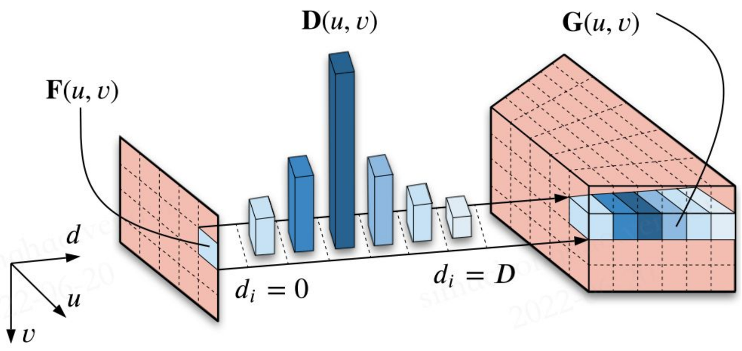


Academic Solution

- General BEV Perception / A summary of the evolution
- 3D-2D v.s. 2D-3D / Comparison of two methods

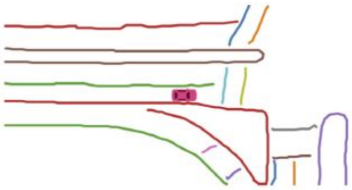
Trending in Academia: BEV Perception

2021.7

- HDMapNet

- Given HD map in BEV coordinates
- Propose the aggregation of feature extracted from both Camera and LiDAR

- Output BEV Map



Vectorized HD map

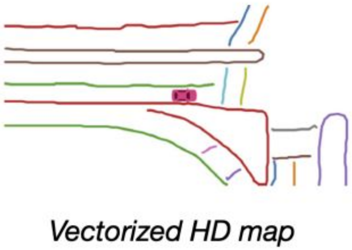
Trending in Academia: BEV Perception

2021.7

- HDMapNet

- Given HD map in BEV coordinates
- Propose the aggregation of feature extracted from both Camera and LiDAR

- Output BEV Map

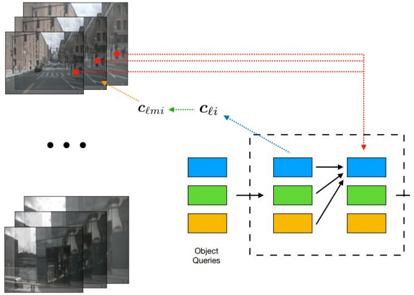


2021.10-12

- DETR3D
- BEVDet

- Fused Object Detection using omnidirectional cameras in BEV

- Implicitly processing BEV features

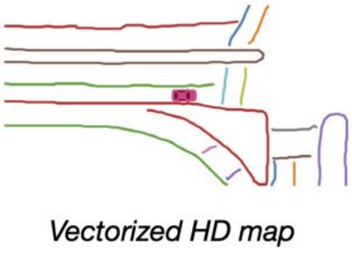


Trending in Academia: BEV Perception

2021.7
- HDMapNet

- Given HD map in BEV coordinates
- Propose the aggregation of feature extracted from both Camera and LiDAR

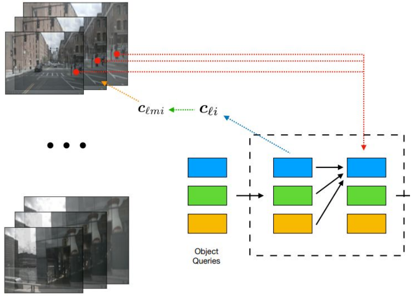
- Output BEV Map



2021.10-12
- DETR3D
- BEVDet

- Fused Object Detection using omnidirectional cameras in BEV

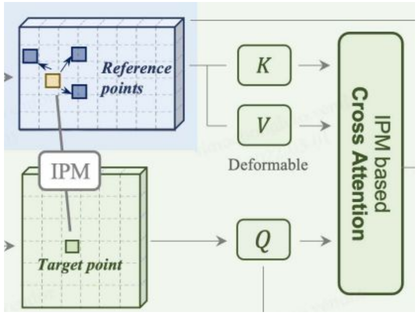
- Implicitly processing BEV features



2022.3
- **BEVFormer**
- **PersFormer**

- **Explicitly Construct BEV representation via camera parameters**

- Explicitly processing BEV feature



Trending in Academia: BEV Perception

2021.7

- HDMapNet

- Given HD map in BEV coordinates
- Propose the aggregation of feature extracted from both Camera and LiDAR

2021.10-12

- DETR3D
- BEVDet

- Fused Object Detection using omnidirectional cameras in BEV

2022.3

- **BEVFormer**
- **PersFormer**

- **Explicitly Construct BEV representation via camera parameters**

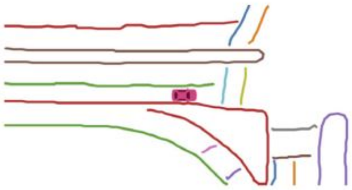
2022.5

- BEVFusion(Damo Academy)
- BEVFusion(MIT)
- FUTR3D

- **Multimodal feature fusion in BEV**

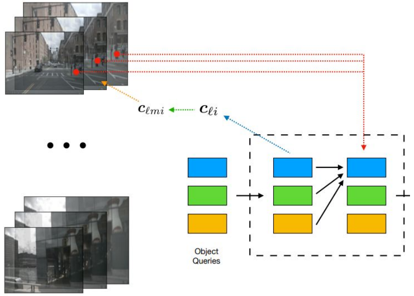
Core Question: How to model the View Transformation from front view to BEV to obtain more effective features?

- Output BEV Map

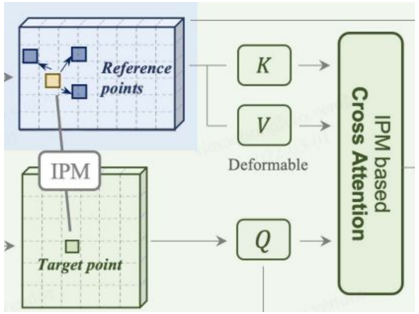


Vectorized HD map

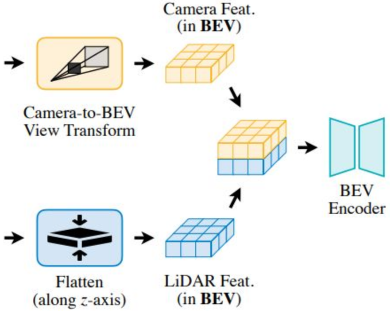
- Implicitly processing BEV features100



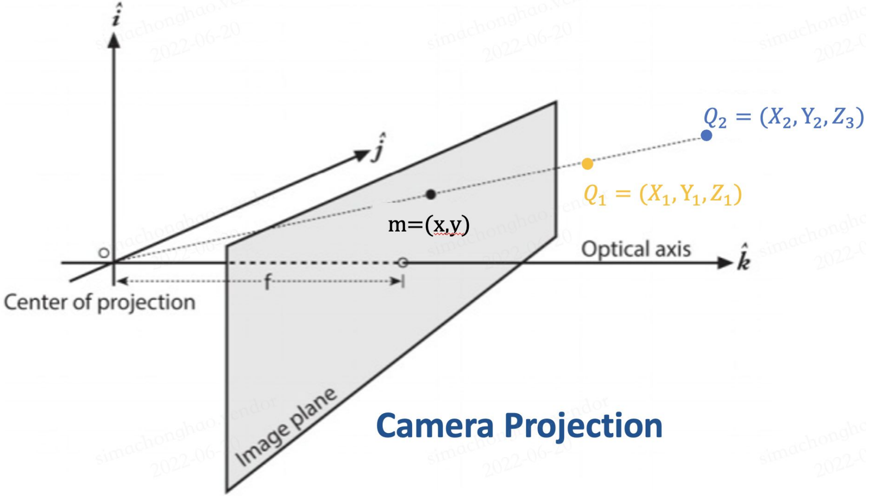
- Explicitly processing BEV features



- Fusion on the dimension of BEV-feature



View Transformation

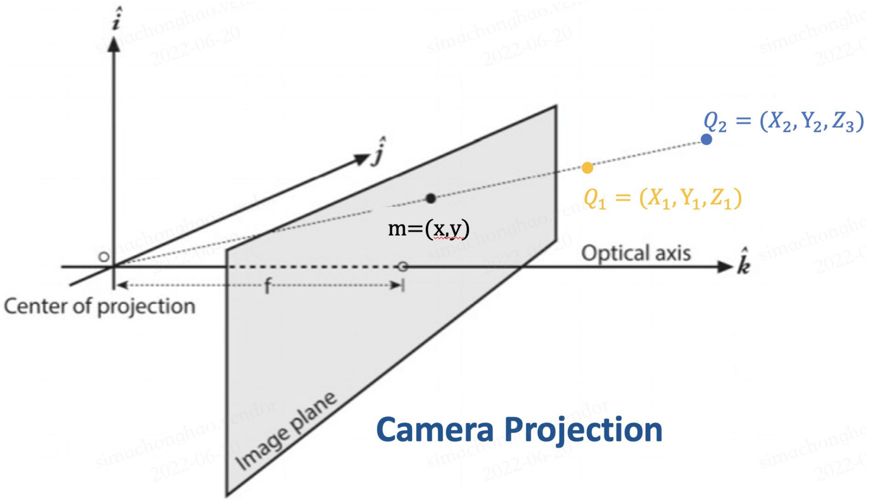


Camera Projection

Issues:

- From 3D to 2D:

View Transformation

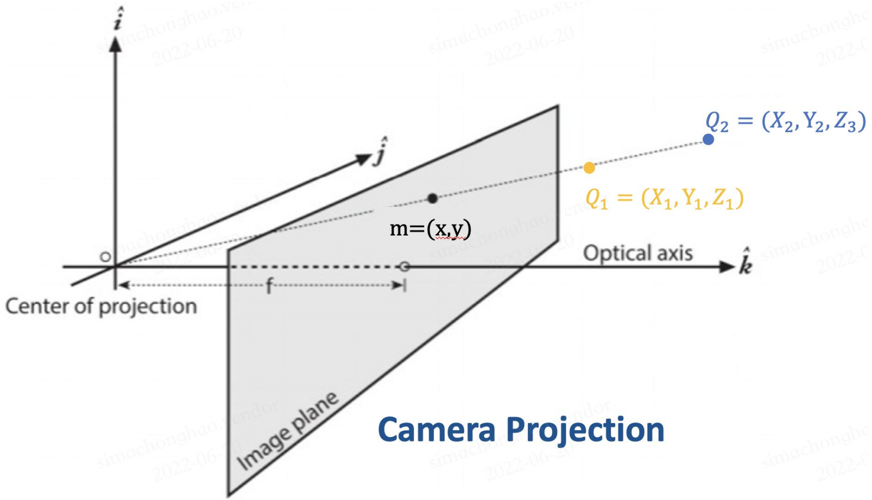


Camera Projection

Issues:

- From 3D to 2D:
  - Multiple 3D points will hit the ***same 2D pixel***

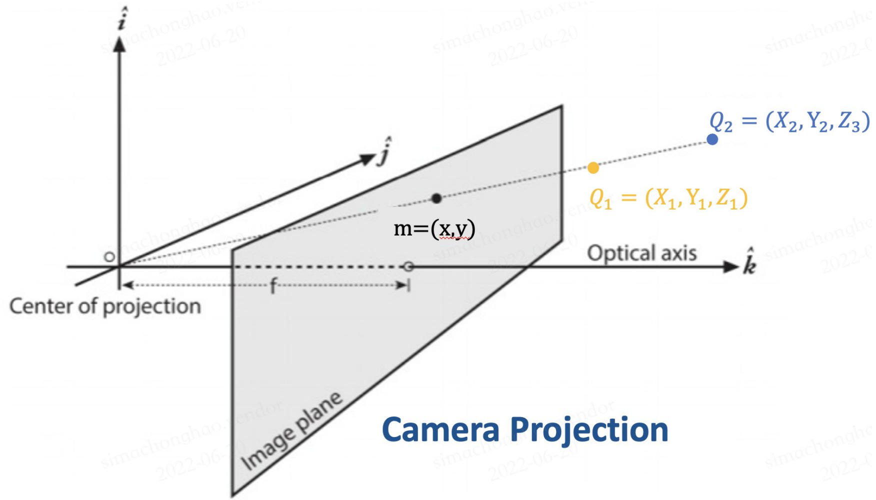
View Transformation



Issues:

- From 3D to 2D:
  - Multiple 3D points will hit the ***same 2D pixel***
- From 2D to 3D:

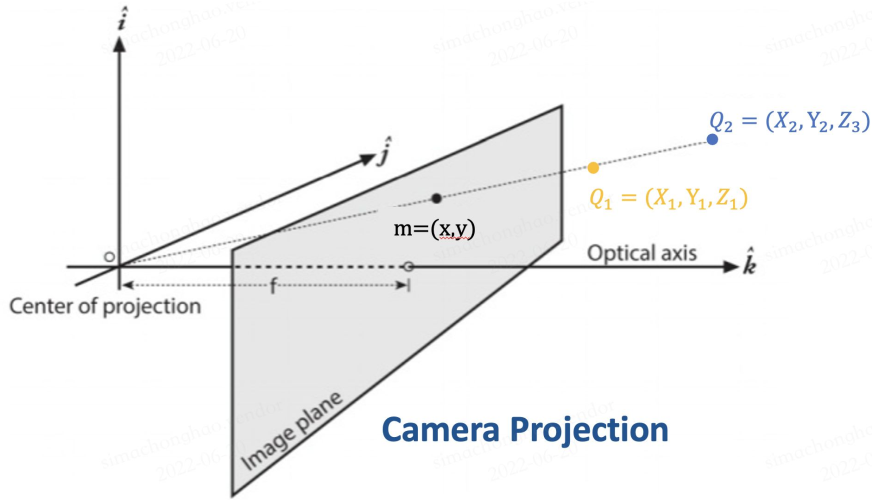
View Transformation



Issues:

- From 3D to 2D:
  - Multiple 3D points will hit the ***same 2D pixel***
- From 2D to 3D:
  - ***Depth*** is unknown

View Transformation



Issues:

- From 3D to 2D:
  - Multiple 3D points will hit the ***same 2D pixel***
- From 2D to 3D:
  - ***Depth*** is unknown

No matter what, the transformation is ill-posed

Two Ways to Address View Transformation

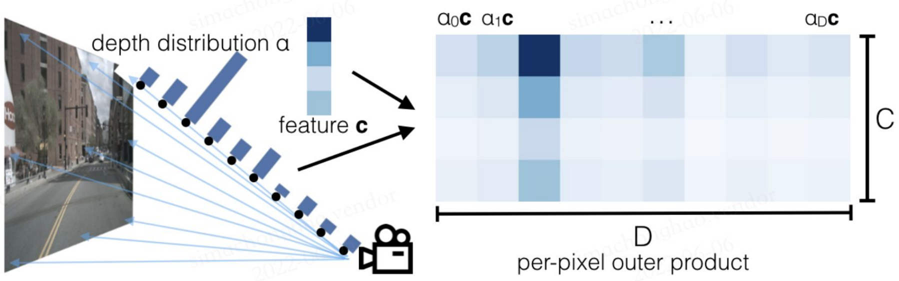
Way 1: From-2D-to-3D prior

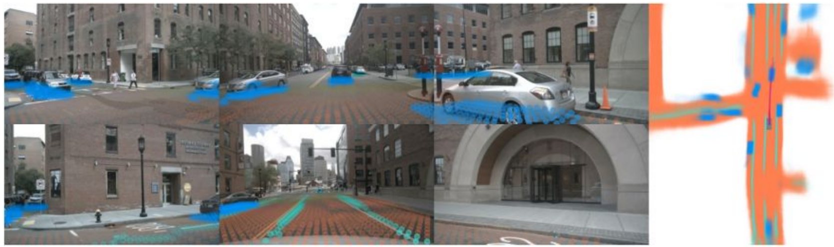
- Now that depth is unknown, we predict depth
  1. Lift, Splat, Shoot and its derivant
  2. Pseudo-LiDAR Family

Way 2: From-3D-to-2D prior

- Index local features according to the projection from 3D to 2D
  1. DETR3D and its derivant
  2. Explicit BEV feature
- Implicit 3D Positional Embedding

2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant

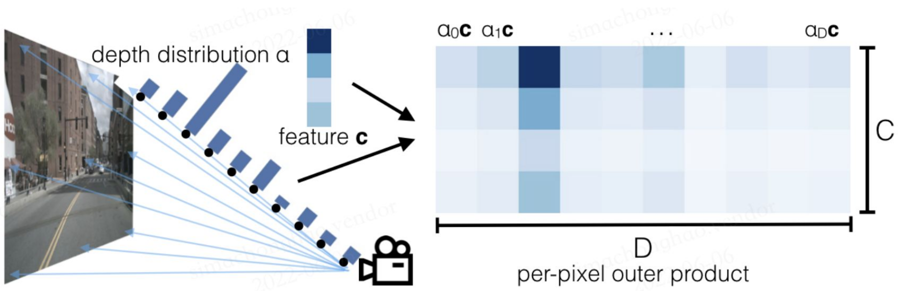


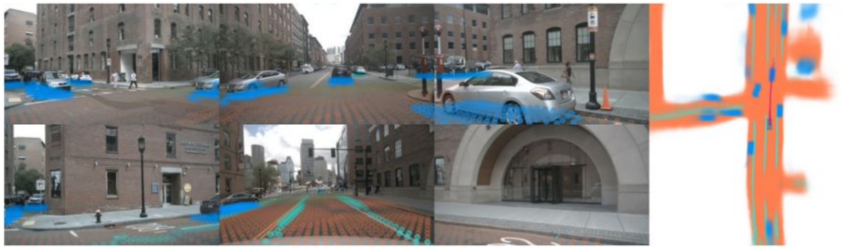


- Lift-Splat-Shoot(LSS) [1] : Using binned depth distribution instead of continuous depth estimation

[1] Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D, *ECCV 2020.*

2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant





- Lift-Splat-Shoot(LSS) [1] : Using binned depth distribution instead of continuous depth estimation
- **Pros** :
    - Depth distribution is easier to generate
- **Cons** :
    - Generated distribution is discrete and sparse, strongly different from real scenes.
    - Object boundaries are difficult to process
- **Following works** :
    - CaDDN [2]
    - FIERY [3]
    - **BEVDet / BEVFusion / BEVDepth**

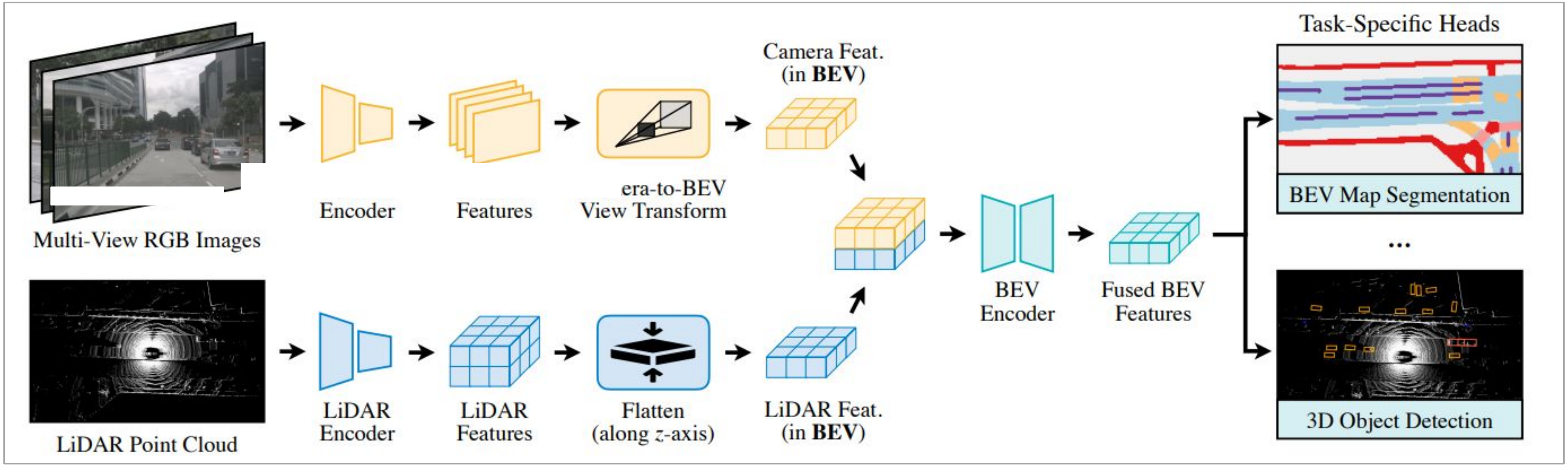
[1] Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D, *ECCV 2020.*

[2] Categorical Depth Distribution Network for Monocular 3D Object Detection, *CVPR 2021.*

[3] FIERY: Future Instance Prediction in Bird's-Eye View from Surround Monocular Cameras, *ICCV 2021 (Oral).*

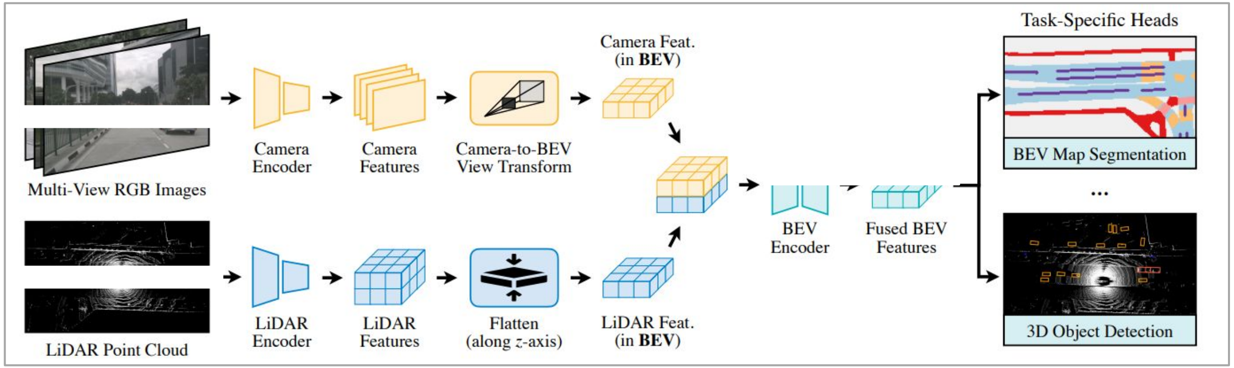
2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant

- BEVFusion [1] : LSS + VoxelNet



[1] BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation, *arxiv:2205.13542.*

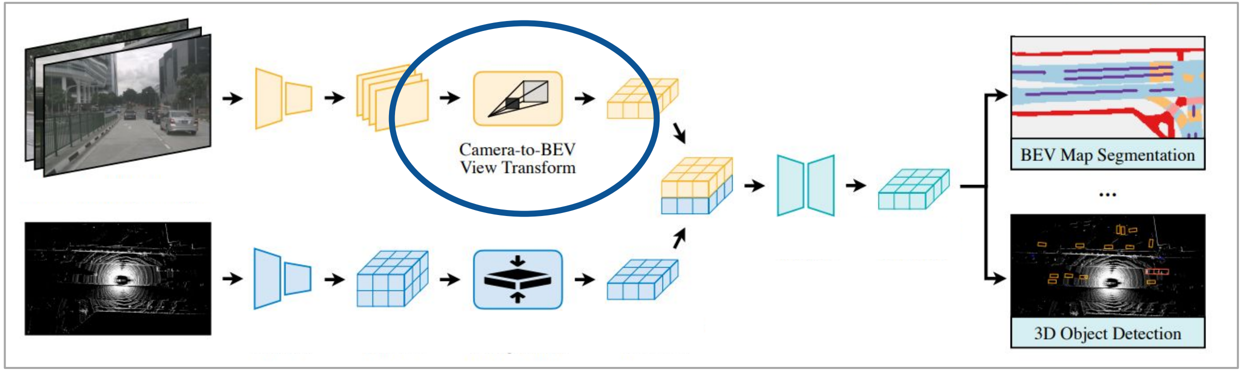
2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant

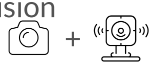


- BEVFusion [1]: LSS + VoxelNet
- **Camera-to-BEV View Transform** accelerates BEV pooling based on LSS
- **Fuse** Camera and LiDAR feature in BEV based on TransFusion

[1] BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation, *arxiv:2205.13542.*

2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant

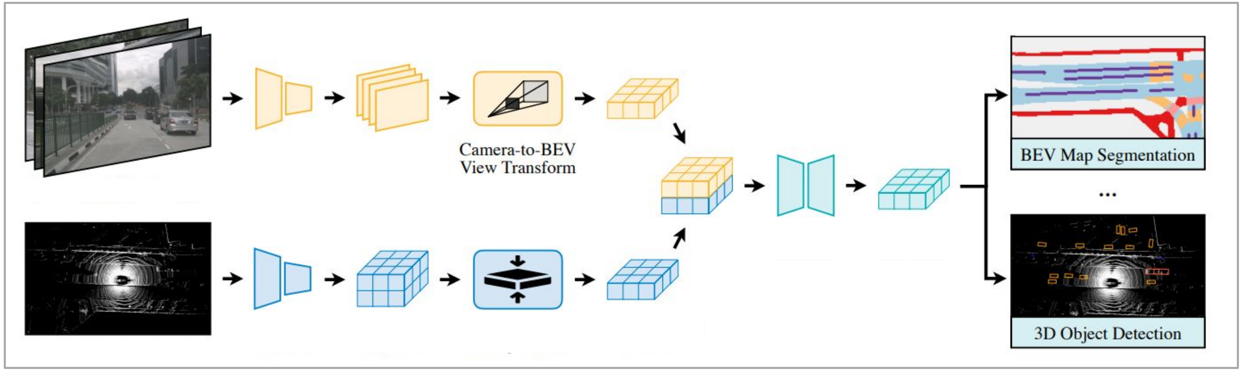


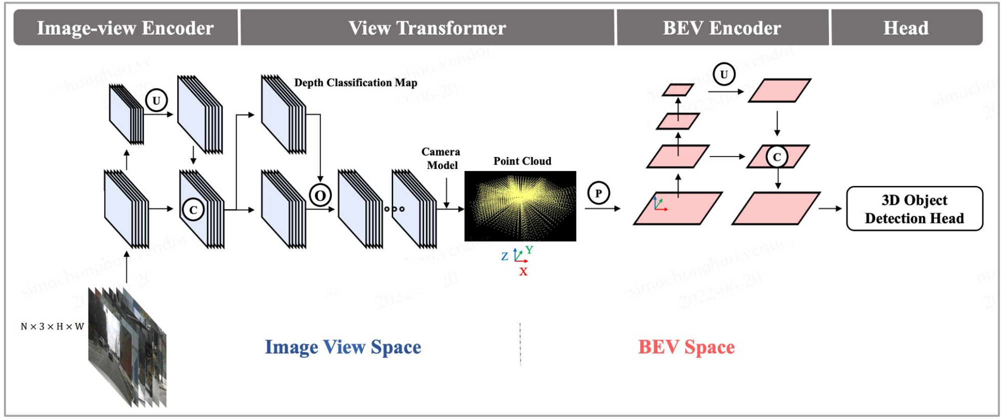
- BEVFusion [1]: LSS + VoxelNet
- **Camera-to-BEV View Transform** accelerates BEV pooling based on LSS
- Fuse Camera and LiDAR feature in BEV based on TransFusion
- 
- nuScenes NDS: 0.761

Current SOTA
Any Modality

[1] BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation, *arxiv:2205.13542*.

2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant





- BEVFusion [1]: LSS + VoxelNet
- **Camera-to-BEV View Transform**参照LSS的做法，加速BEV pooling
- 参照TransFusion，**在BEV下融合**Camera和LiDAR feature
- nuScenes NDS: 0.761

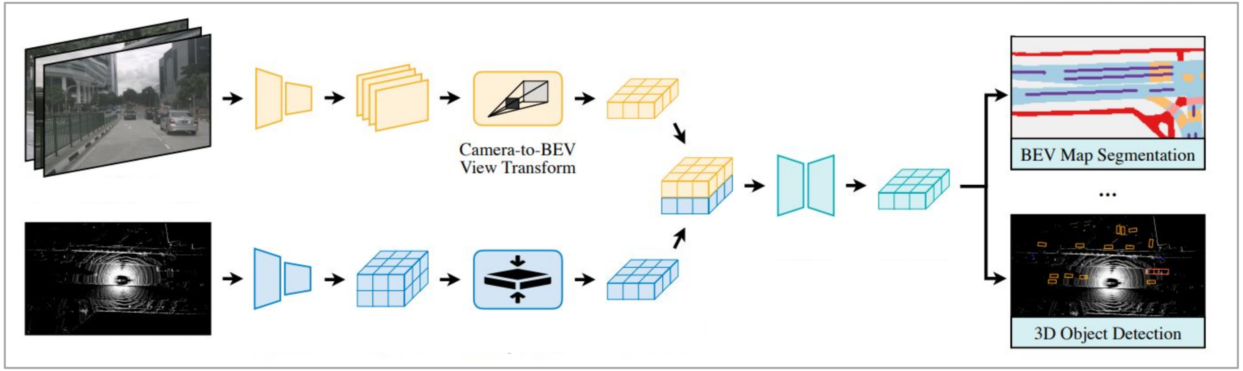
Current SOTA
Any Modality

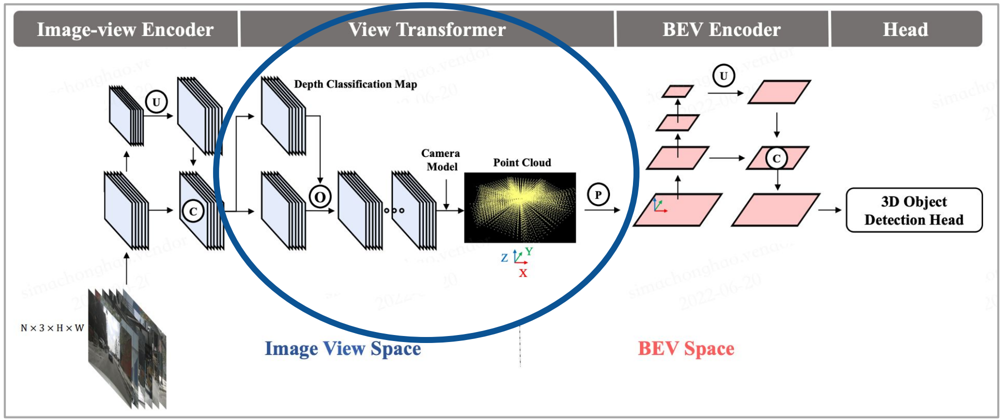
- BEVDet [2]: LSS + CenterPoint

[1] BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation, *arxiv:2205.13542.*

[2] BEVDet: High-Performance Multi-Camera 3D Object Detection in Bird-Eye-View, *arxiv:2112.11790.*

2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant



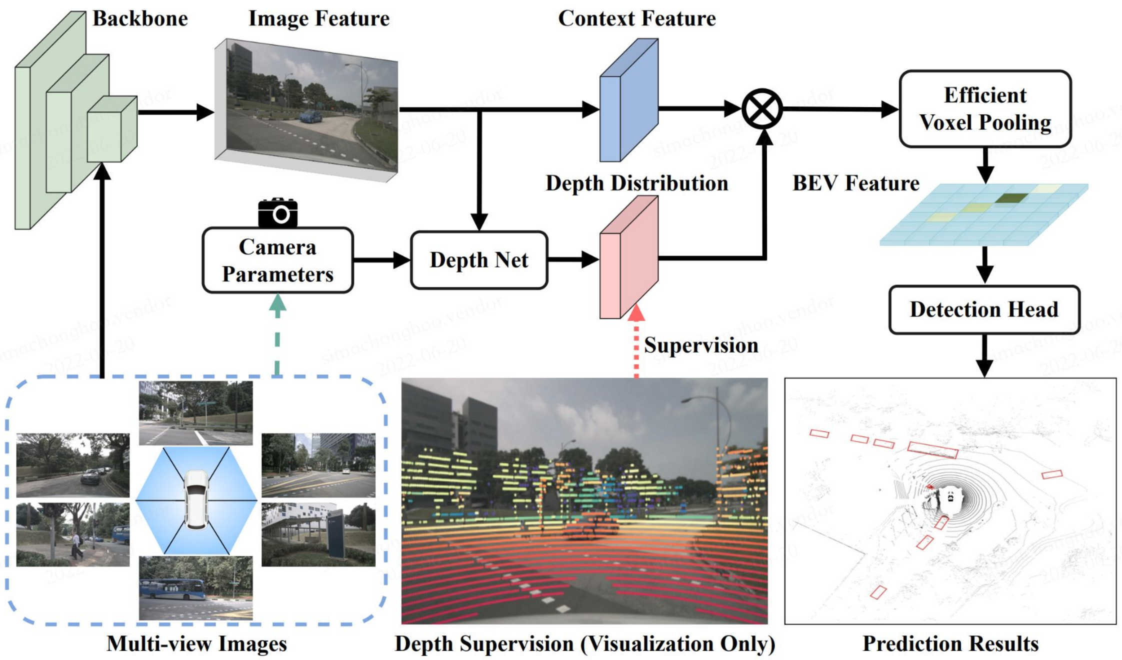


- BEVFusion [1]: LSS + VoxelNet
- **Camera-to-BEV View Transform** accelerates BEV pooling based on LSS
- **Fuse** Camera and LiDAR feature in BEV based on TransFusion
- +
- **Current SOTA Any Modality** 1
- BEVDet [2]: LSS + CenterPoint
- **View TransFormer** improved on the basis of LSS and added data augmentation in the BEV space
- nuScenes NDS: 0.569

[1] BEVFusion: Multi-Task Multi-Sensor Fusion with Unified Bird's-Eye View Representation, *arxiv:2205.13542.*

[2] BEVDet: High-Performance Multi-Camera 3D Object Detection in Bird-Eye-View, *arxiv:2112.11790.*

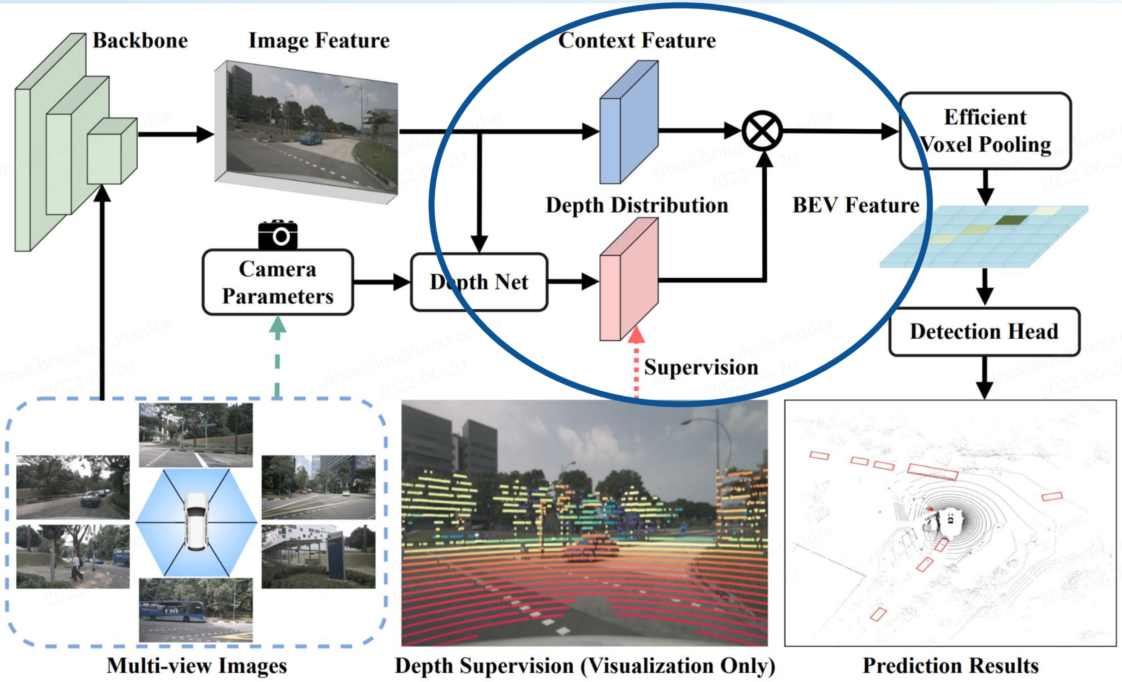
2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant



- BEVDepth [1] : LSS + Depth supervision

[1] BEVDepth: Acquisition of Reliable Depth for Multi-view 3D Object Detection, *arXiv:2206.10092.*

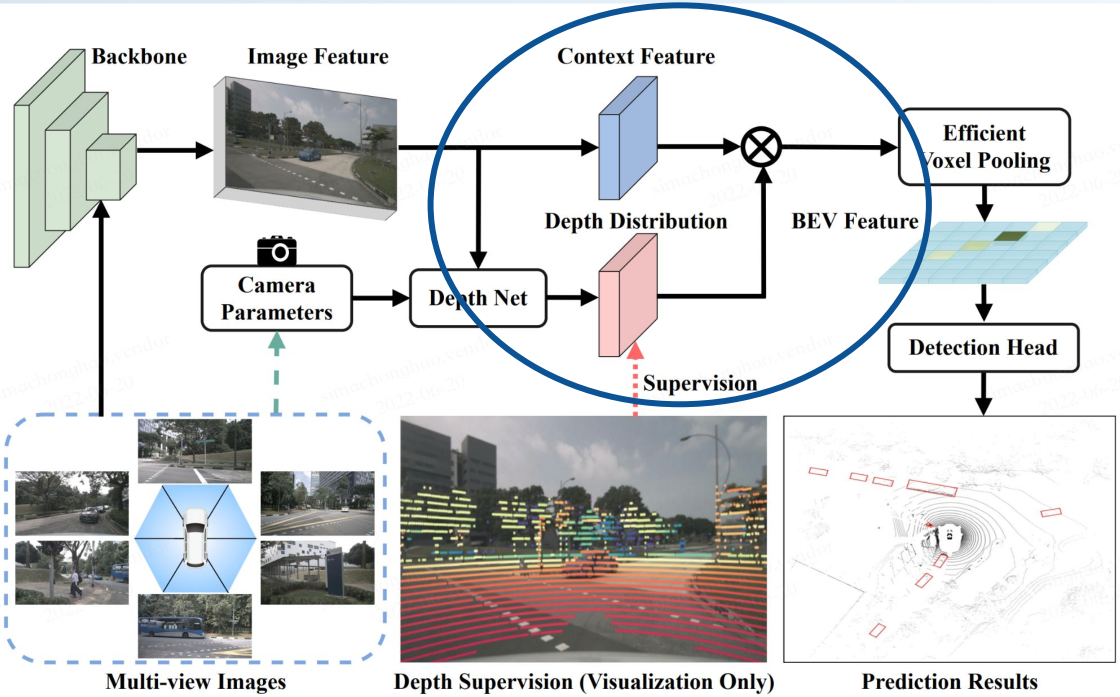
2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant



- BEVDepth [1] : LSS + Depth supervision
- **Based on the method used in LSS,** LiDAR is added as the supervision signal for depth distribution of BEV feature

[1] BEVDepth: Acquisition of Reliable Depth for Multi-view 3D Object Detection, *arXiv:2206.10092.*

2D to 3D: Lift-Splat-Shoot (LSS) and its Derivant

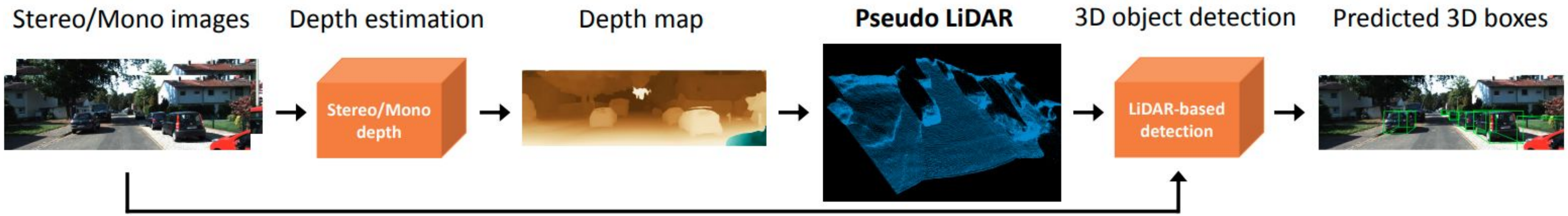


- BEVDepth [1] : LSS + Depth supervision
- **Based on the method used in LSS,** LiDAR is added as the supervision signal for depth distribution of BEV feature
- nuScenes NDS: 0.600

Current SOTA Camera-only

[1] BEVDepth: Acquisition of Reliable Depth for Multi-view 3D Object Detection, *arXiv:2206.10092.*

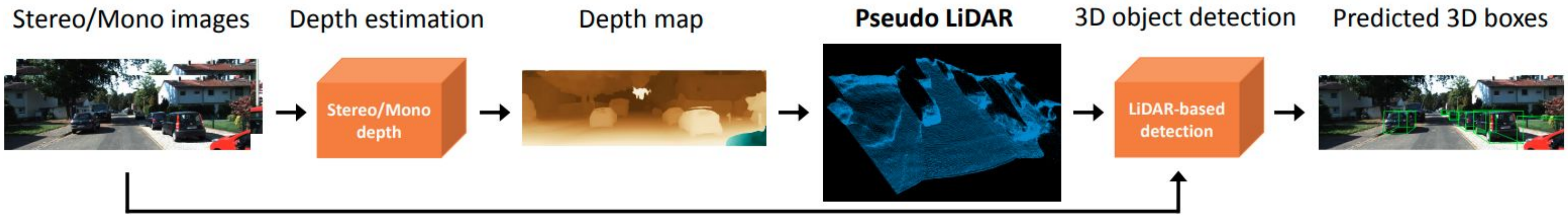
2D to 3D: Pseudo Lidar Family



- Pseudo-LiDAR [1] : Using pixel-level depth estimation to extend image to pseudo point cloud

[1] Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving Representation, *CVPR 2019.*

2D to 3D: Pseudo Lidar Family



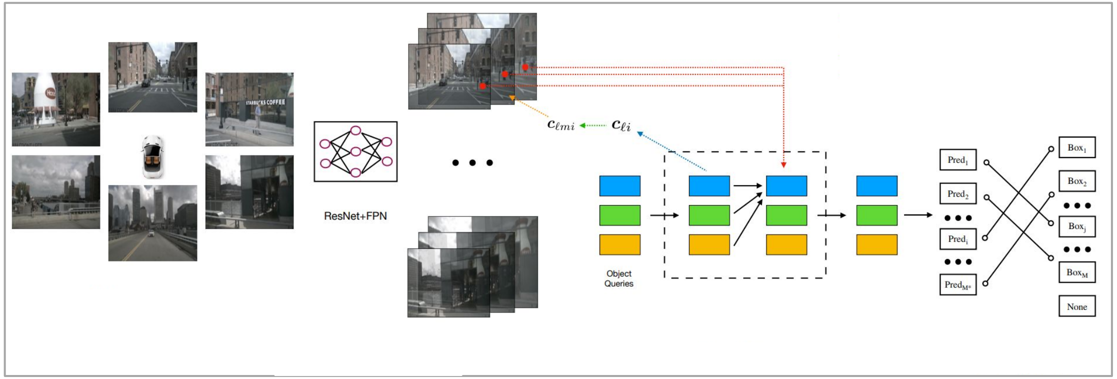
- Pseudo-LiDAR [1] : Using pixel-level depth estimation to extend image to pseudo point cloud
- **Pros** :
  - Depth map is continuous, friendly for point cloud detection
- **Cons** :
  - Detection quality strongly depends on depth estimation which is usually inaccurate up till now.
  - The absolute pixel-level depth value in outdoor scenes is difficult to acquire
- **Following works** :
  - Pseudo-LiDAR++ [2]
  - Patch-Net [3]

[1] Pseudo-LiDAR from Visual Depth Estimation: Bridging the Gap in 3D Object Detection for Autonomous Driving Representation, *CVPR 2019.*

[2] Pseudo-LiDAR++: Accurate Depth for 3D Object Detection in Autonomous Driving, *ICLR 2020.*

[3] Rethinking Pseudo-LiDAR Representation, *ECCV 2020.*

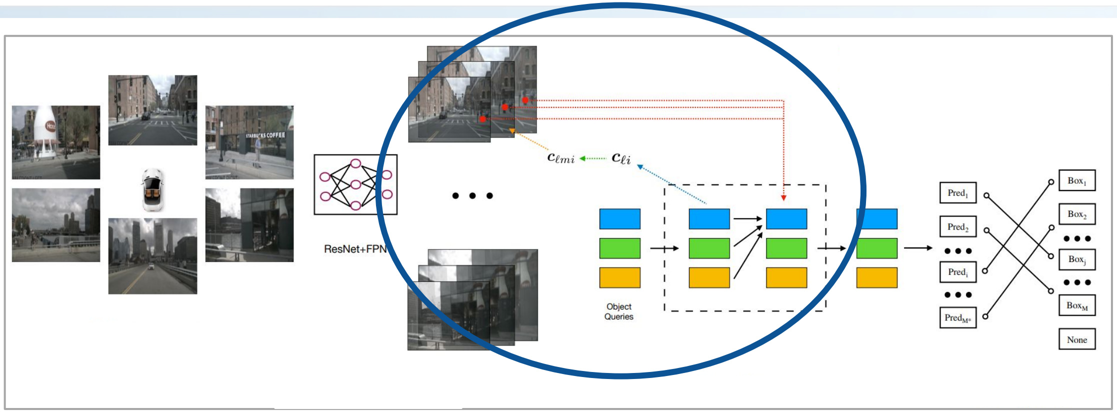
3D to 2D: DETR3D and its Derivant



- DETR3D [1] : Sample front view features based on the relationship between BEV and front view on the panoramic camera feature, and output 3D object detection

[1] DETR3D: 3D Object Detection from Multi-view Images via 3D-to-2D Queries, *CoRL 2021.*

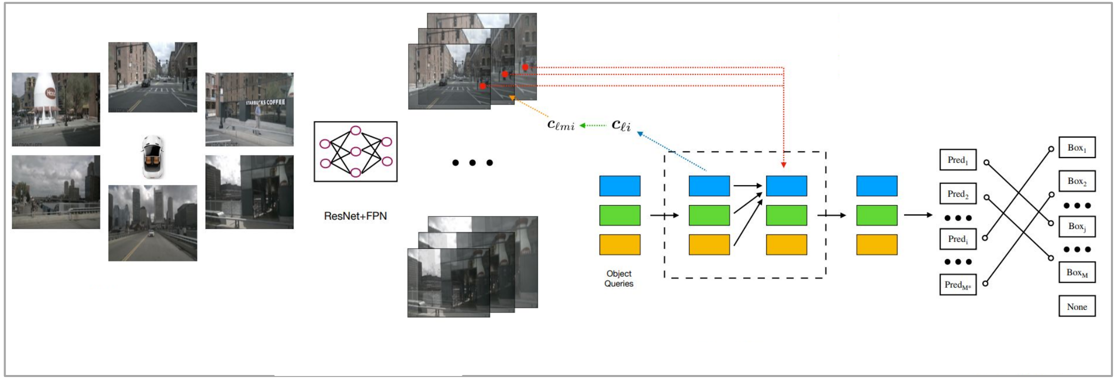
3D to 2D: DETR3D and its Derivant

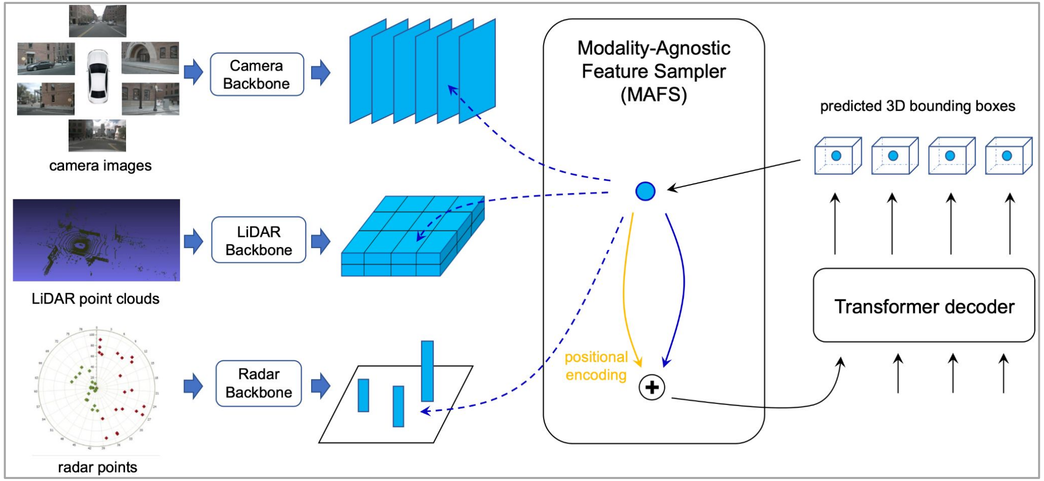


- DETR3D [1] : Sample front view features based on the relationship between BEV and front view on the panoramic camera feature, and output 3D object detection
- **Feature Transformation**: Project 3D query to 2D(front view) space, look up 2D feature, and decode into object bbox
- nuScenes NDS: 0.479

[1] DETR3D: 3D Object Detection from Multi-view Images via 3D-to-2D Queries, *CoRL 2021*.

3D to 2D: DETR3D and its Derivant





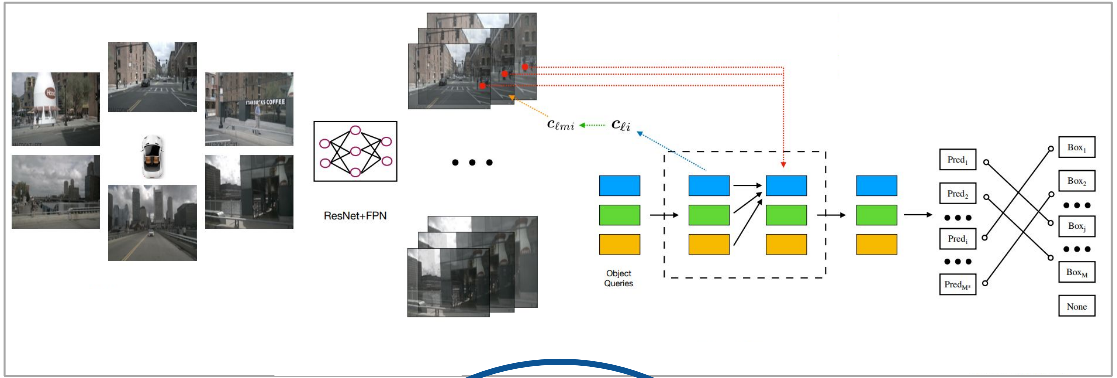
- DETR3D [1] : Sample front view features based on the relationship between BEV and front view on the panoramic camera feature, and output 3D object detection
- **Feature Transformation**: Project 3D query to 2D(front view) space, look up 2D feature, and decode into object bbox
- nuScenes NDS: 0.479

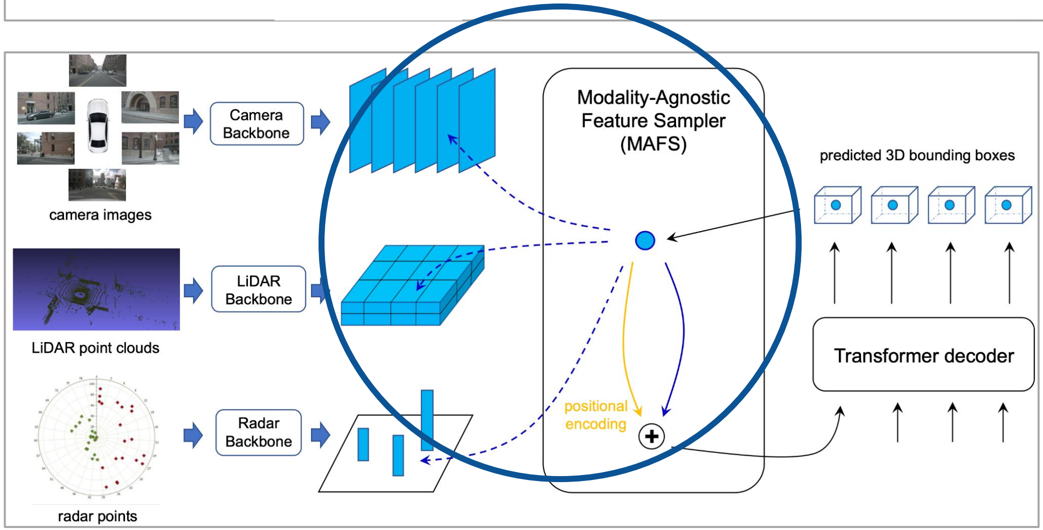
- FUTR3D [2] : sensor fusion of DETR3D

[1] DETR3D: 3D Object Detection from Multi-view Images via 3D-to-2D Queries, *CoRL 2021*.

[2] FUTR3D: A Unified Sensor Fusion Framework for 3D Detection, *arXiv:2203.10642*.

3D to 2D: DETR3D and its Derivant



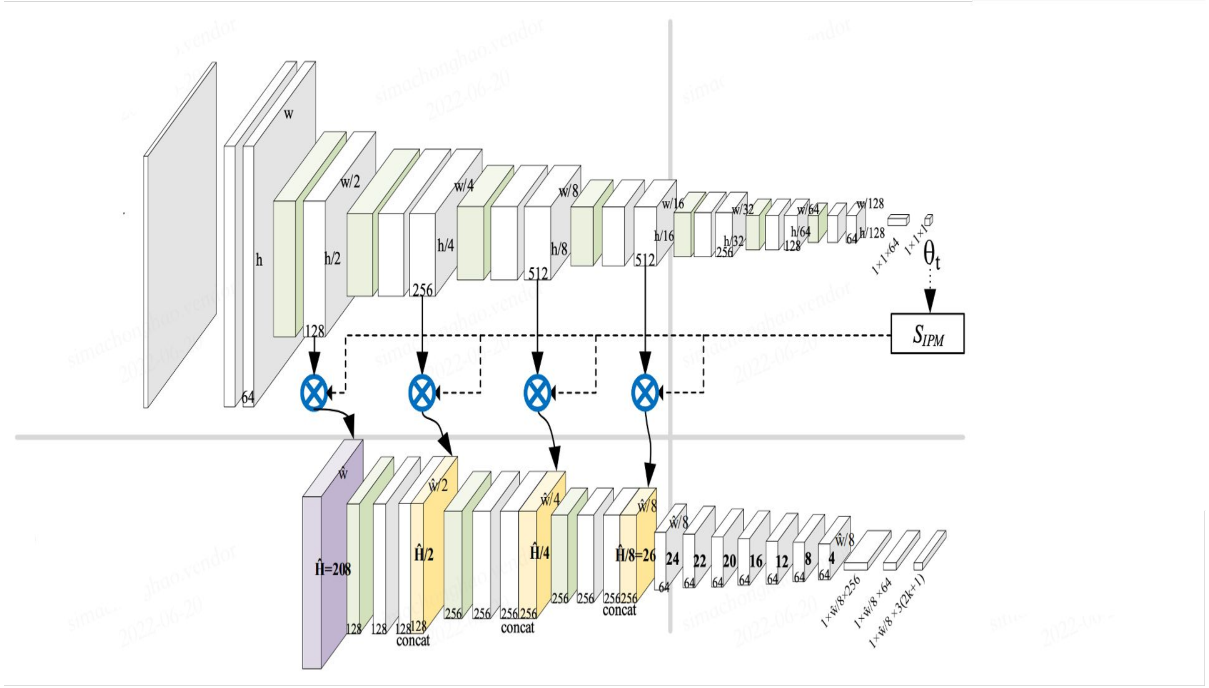


- DETR3D [1] : Sample front view features based on the relationship between BEV and front view on the panoramic camera feature, and output 3D object detection
- **Feature Transformation**: Project 3D query to 2D(front view) space, look up 2D feature, and decode into object bbox
- nuScenes NDS: 0.479
-
- FUTR3D [2] : sensor fusion of DETR3D
- **MAFS** : 3D query are projected to 2D plane respectively，voxel，radar are used to look up feature，and then decode into object bbox
- nuScenes NDS: 0.680

[1] DETR3D: 3D Object Detection from Multi-view Images via 3D-to-2D Queries, *CoRL 2021*.

[2] FUTR3D: A Unified Sensor Fusion Framework for 3D Detection, *arXiv:2203.10642*.

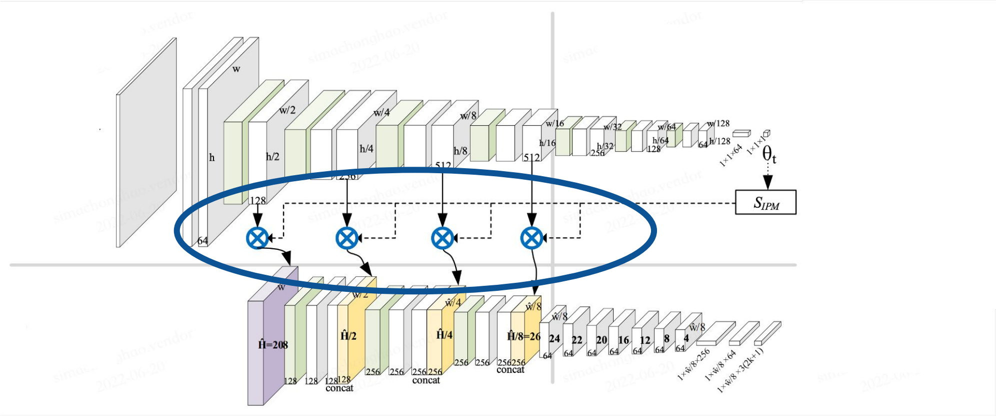
3D to 2D: Explicit BEV Feature



3D-LaneNet [1] : 3D lane detection in BEV

[1] 3D-LaneNet: End-to-End 3D Multiple Lane Detection, *ICCV 2019.*

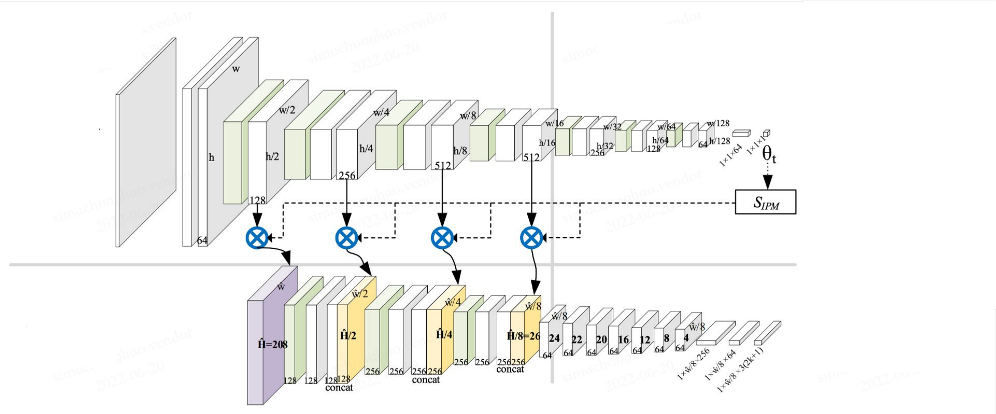
3D to 2D: Explicit BEV Feature

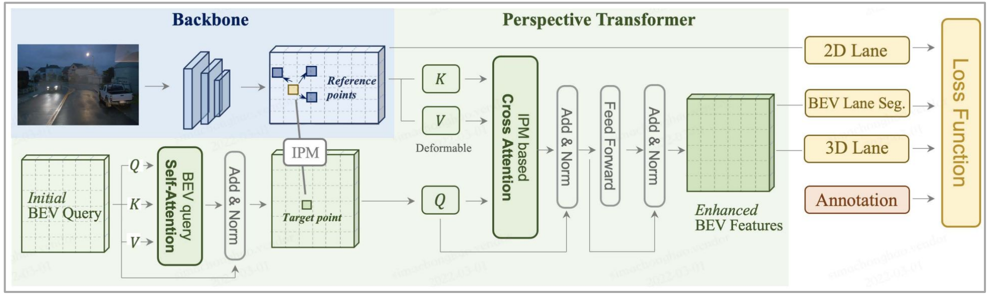


- 3D-LaneNet [1] : 3D lane detection in BEV
- **Projection to Top view** : feature is projected from front view to BEV ***based on IPM***, grid sampler is then used to obtain BEV feature
- OpenLane F1: 40.2

[1] 3D-LaneNet: End-to-End 3D Multiple Lane Detection, *ICCV 2019.*

3D to 2D: Explicit BEV Feature





- 3D-LaneNet [1] : 3D lane detection in BEV
- **Projection to Top view** : feature is projected from front view to BEV ***based on IPM***, grid sampler is then used to obtain BEV feature
- OpenLane F1: 40.2

PerceptionX https://github.com/OpenPerceptionX/OpenLane

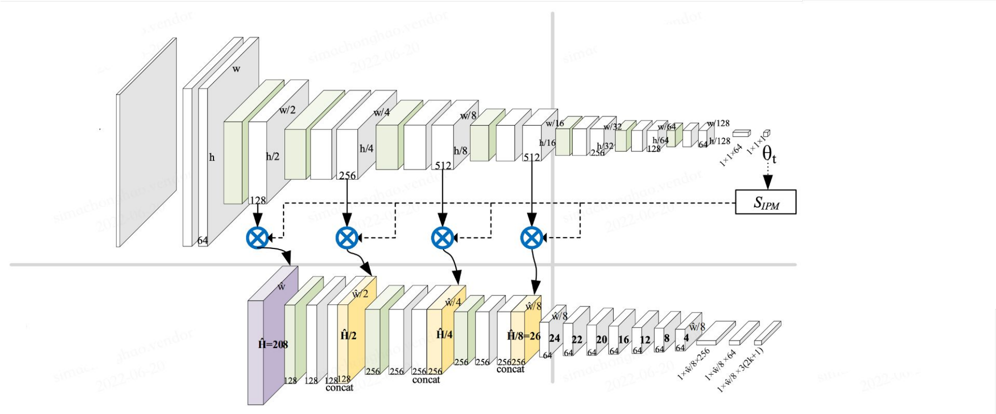
- PersFormer [2] : Joint detection of 2D-3D lane

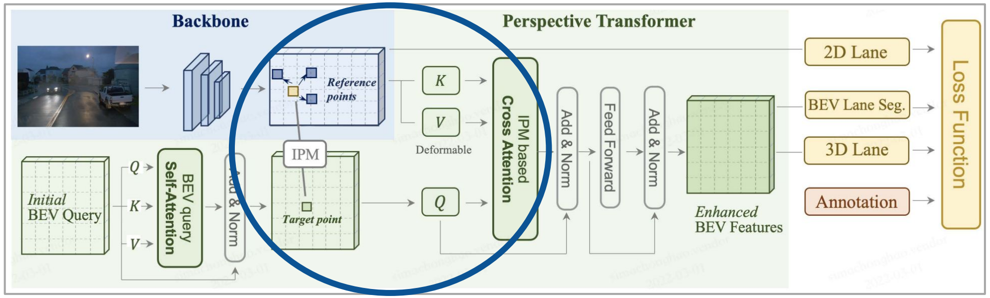
[1] 3D-LaneNet: End-to-End 3D Multiple Lane Detection, *ICCV 2019.*

[2] PersFormer: 3D Lane Detection via Perspective Transformer and the OpenLane Benchmark, *arXiv:2203.11089.*

3D to 2D: Explicit BEV Feature

上海人工智能实验室
Shanghai Artificial Intelligence Laboratory





- 3D-LaneNet [1]: 3D lane detection in BEV
- **Projection to Top view** : feature is projected from front view to BEV ***based on IPM***, grid sampler is then used to obtain BEV feature
- OpenLane F1: 40.2

Current SOTA

PerceptionX

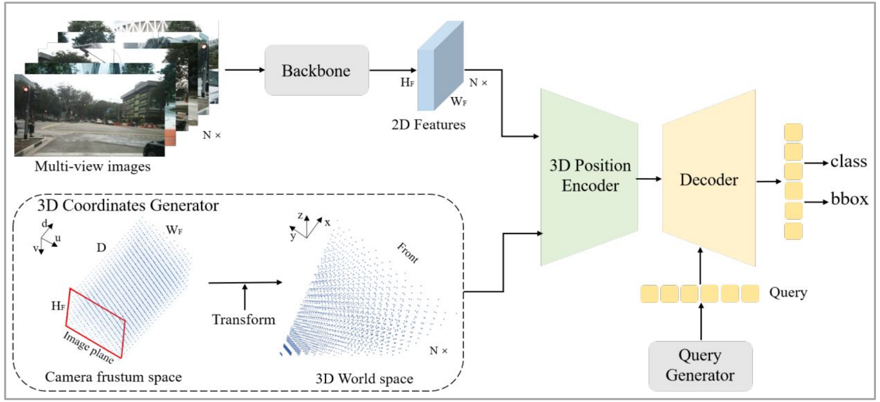
https://github.com/OpenPerceptionX/OpenLane

- PersFormer [2]: Joint detection of both 2D-3D lane
- **IPM-based Cross Attention** : front view feature is used as keyand value, reference points of 2D-BEV are acquired via ***IPM***, use BEV query to retrieve a BEV feature
- OpenLane F1: 49.0

[1] 3D-LaneNet: End-to-End 3D Multiple Lane Detection, *ICCV 2019.*

[2] PersFormer: 3D Lane Detection via Perspective Transformer and the OpenLane Benchmark, *arXiv:2203.11089.*

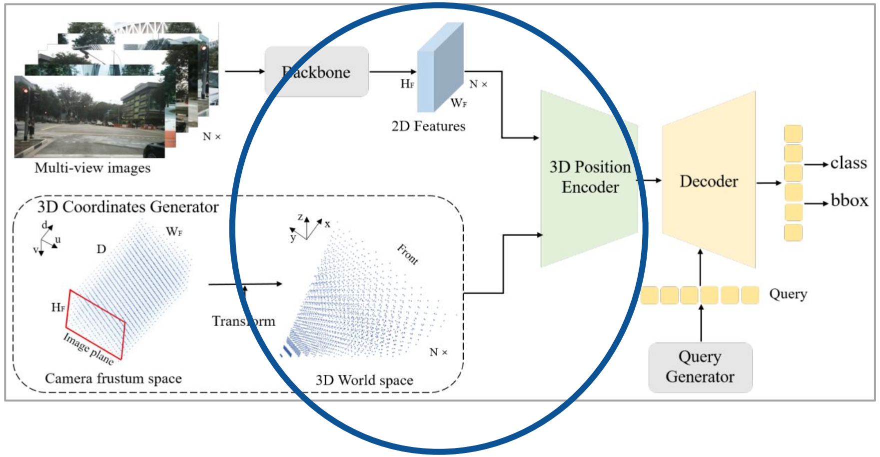
3D to 2D: Implicit 3D Positional Embedding



- PETR [1] : 3D object detection based on 3D position encoding

[1] PETR: Position Embedding Transformation for Multi-View 3D Object Detection, *arxiv:2203.05625.*

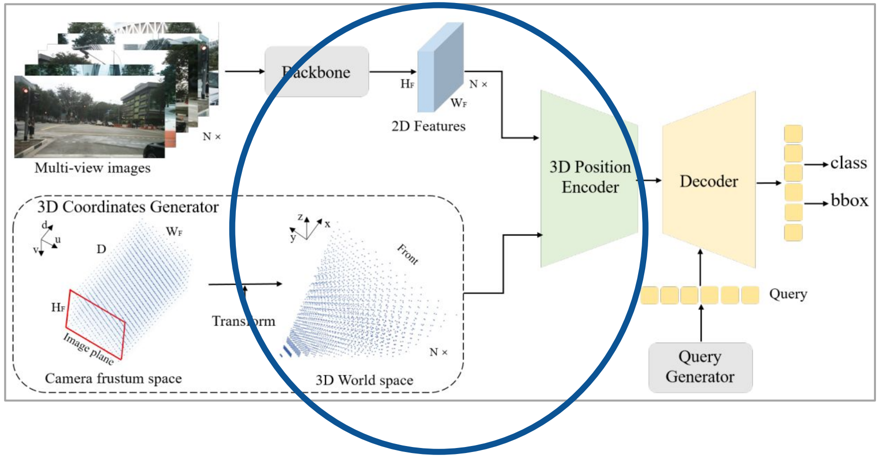
3D to 2D: Implicit 3D Positional Embedding



- PETR [1] : 3D object detection based on 3D position encoding
- **3D Coordinates Generator & 3D Position Encoder** : generate position encoding based on 3D coordinates, encode 3D features and input them into the encoder to output features with 3D spatial information
- nuScenes NDS: 0.481

[1] PETR: Position Embedding Transformation for Multi-View 3D Object Detection, *arxiv:2203.05625*.

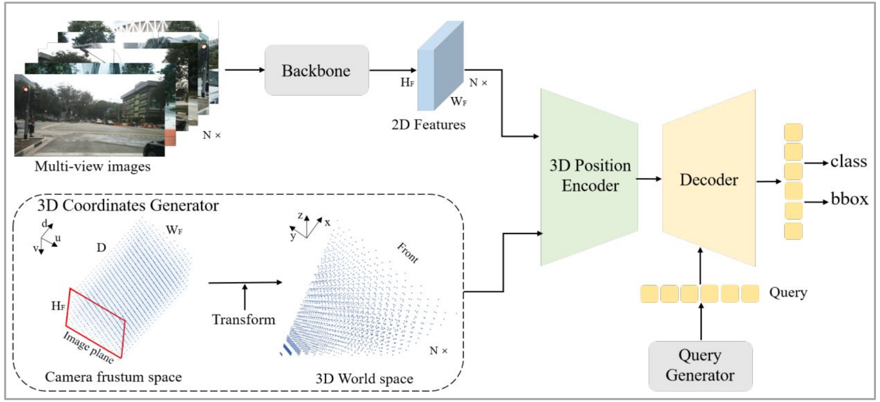
3D to 2D: Implicit 3D Positional Embedding

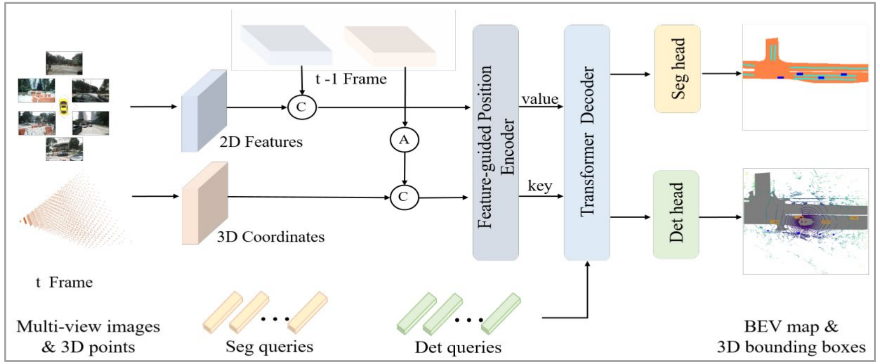


- PETR [1] : 3D object detection based on 3D position encoding
- **3D Coordinates Generator & 3D Position Encoder** : generate position encoding based on 3D coordinates, encode 3D features and input them into the encoder to output features with 3D spatial information
- nuScenes NDS: 0.481

[1] PETR: Position Embedding Transformation for Multi-View 3D Object Detection, *arxiv:2203.05625.*

3D to 2D: Implicit 3D Positional Embedding



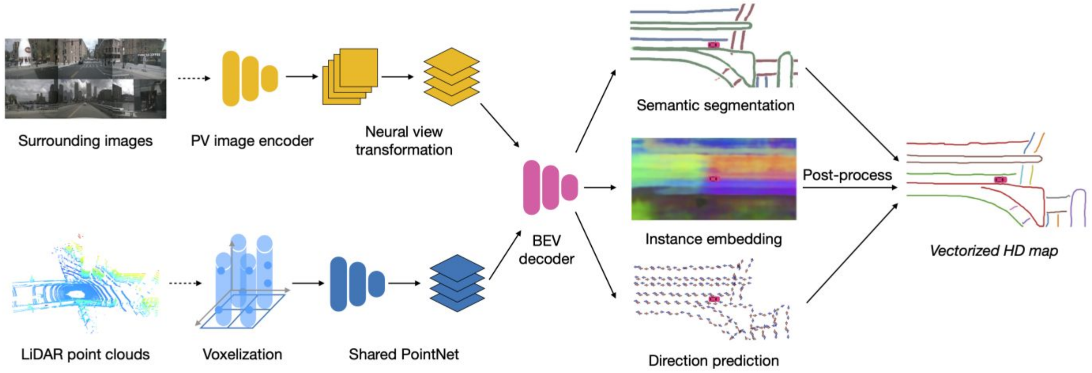


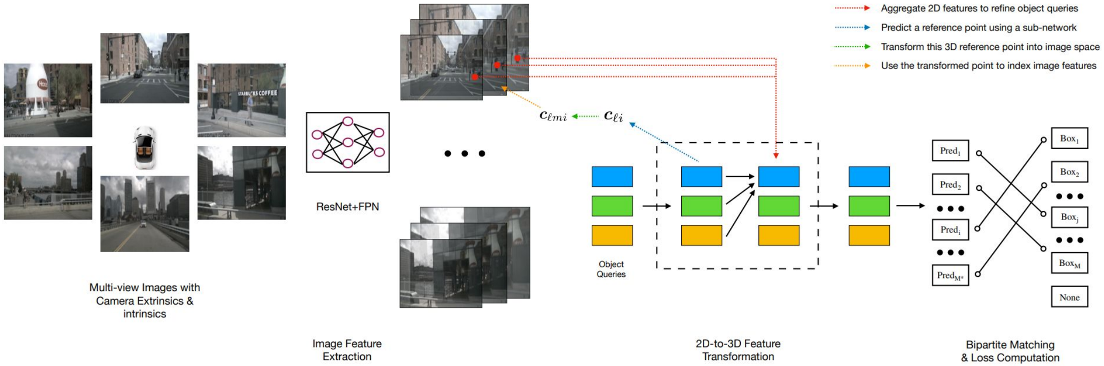
- PETR [1] : 3D object detection based on 3D position encoding
- **3D Coordinates Generator & 3D Position Encoder** : generate position encoding based on 3D coordinates, encode 3D features and input them into the encoder to output features with 3D spatial information
- nuScenes NDS: 0.481
-
- PETRv2 [2]add temporal information based on PETR

[1] PETR: Position Embedding Transformation for Multi-View 3D Object Detection, *arxiv:2203.05625.*

[2] PETRv2: A Unified Framework for 3D Perception from Multi-Camera Images, *arxiv:2206.01256.*

Implicit Query-based BEV: HDMapNet-DETR3D





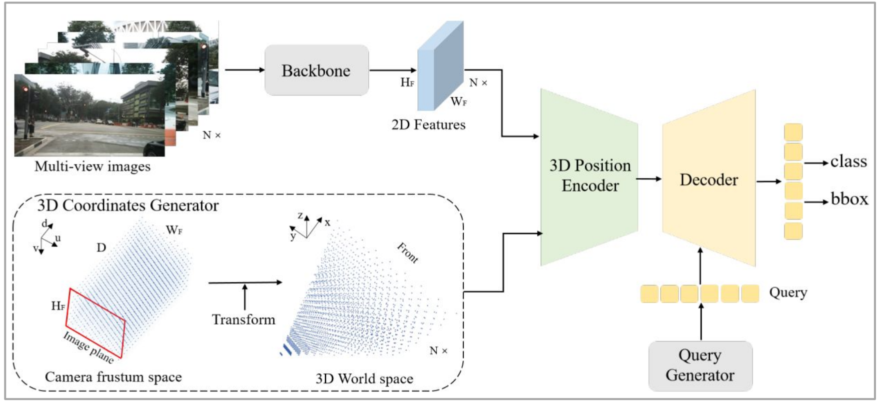
- HDMapNet [1]：Fuse camera and LiDAR feature in BEV, output HD map in BEV
- **Neural view transformation**:Transformation from front view to BEV based on the simple projection and stitching from camera intrinsics and extrinsics, similar to panoramic IPM

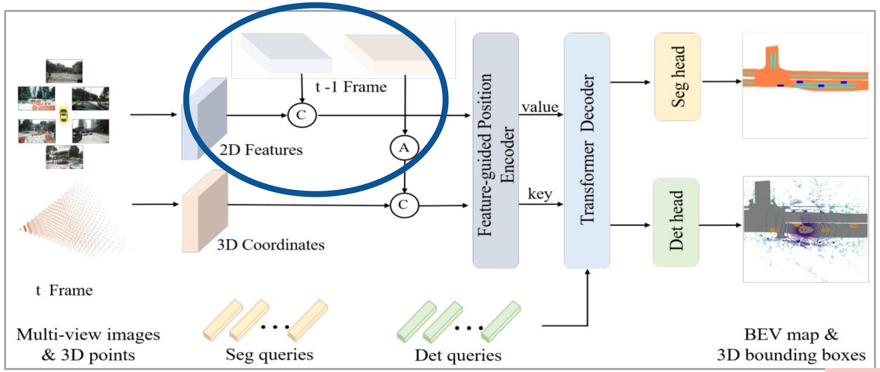
- DETR3D [2]：Sample front view feature according to the BEV-front view relationship based on panoramic camera feature
- **2D-to-3D Feature Transformation** ：project 3D query to 2D（front view）plane，retrieve 2D feature，decode into object bbox

[1] HDMapNet: An Online HD Map Construction and Evaluation Framework, *ICRA 2022*

[2] DETR3D: 3D Object Detection from Multi-view Images via 3D-to-2D Queries, *CoRL 2021*

3D to 2D: Implicit 3D Positional Embedding





- PETR [1]: : 3D object detection based on 3D position encoding
- **3D Coordinates Generator & 3D Position Encoder** : : generate position encoding based on 3D coordinates, encode 3D features and input them into the encoder to output features with 3D spatial information
- nuScenes NDS: 0.481

- PETRv2 [2]add temporal information based on PETR
- **Temporal operation:** fuse historical frame's information in image space
- nuScenes NDS: 0.582

MEGVII 旷视

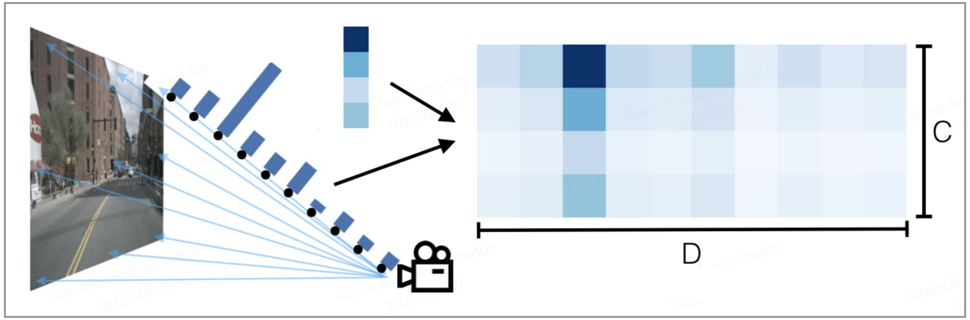
Second Best Camera-only

[1] PETR: Position Embedding Transformation for Multi-View 3D Object Detection, *arxiv:2203.05625.*

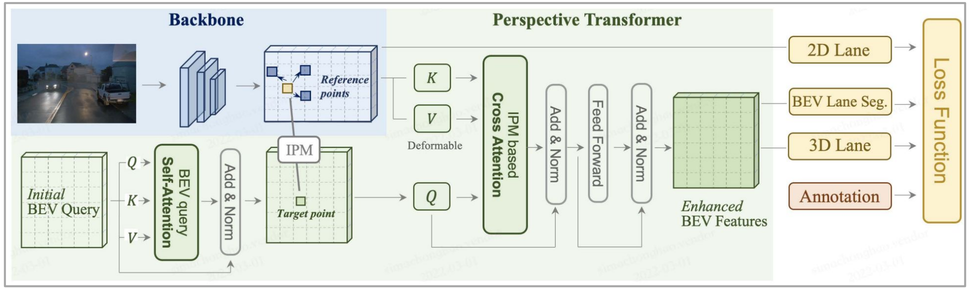
[2] PETRv2: A Unified Framework for 3D Perception from Multi-Camera Images, *arxiv:2206.01256.*

Summary of BEV Perception in Academia

- **From 2D-to-3D prior**
  - Depth estimation
    1. Lift, Splat, Shoot and its derivant
    2. Pseudo-LiDAR Family



- **From 3D-to-2D prior**
  - Index local feature based on the projection from 3D to 2D
    1. DETR3D and its variant
    2. Explicit BEV feature
  - Implicit 3D positional encoding



Both perspectives are promising on nuScenes / Waymo leaderboard

Dataset

Dataset	Task	Sensor Config	Region	Scale	Quality	Influence (Leaderboard, workshop)
KITTI (2009)	• 2D/3D object detection • nV • stereo • depth • Lidar segmentation	• 1L(Velodyne HDL-64E,10Hz) • 4C(90°, 10Hz)	Germany Karlsruhe	• 1.5h • 7.5k frames • 1M images • 47K 3d bbox	★★★★	★★★★★
Waymo	• 2D/3D object detection • nV • Object Tracking • motion forecast • domain gap	• 5L(10Hz) • 5C(50.4°, 10Hz)	America San Francisco Phoenix Mountain View	• 5.5h • 200k frame • 1M images • 1.4M 3d bbox	★★★★★	★★★★
nuScenes	• 3D object detection • nV • Object Tracking • motion forecast	• 1L(20Hz) • 6C(70°,rear cam 110°, 12hz)	America Boston Singapore	• 5.5h • 40K frame • 1.4M images • 1.4M 3d bbox	★★★★	★★★★
Argoverse	• 3D object detection • nV • Object Tracking • motion forecast	• 2L(10Hz) • 9C(69.3°, 30Hz)	America Miami Pittsburgh	• 0.6h • 22K frame • 490K images • 993K 3d bbox	★★★	★★
Lyft L5	• 3D object detection • nV • Object Tracking • motion forecast	• 3L(10Hz) • 7C(87.1°, 10Hz)	America Palo Alto	• 2.5h • 46K frame • 323K images • 1.3M 3d bbox	★★	★★
BDD						

3 BEVFormer: A Shanghai AI Lab Approach

BEVFormer and its Variant

BEVFormer: Demo

BEVFormer: Learning Bird's-Eye-View Representation from Multi-Camera Images via Spatiotemporal Transformers

Zhiqi Li*, Wenhai Wang*, Hongyang Li*, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, Jifeng Dai

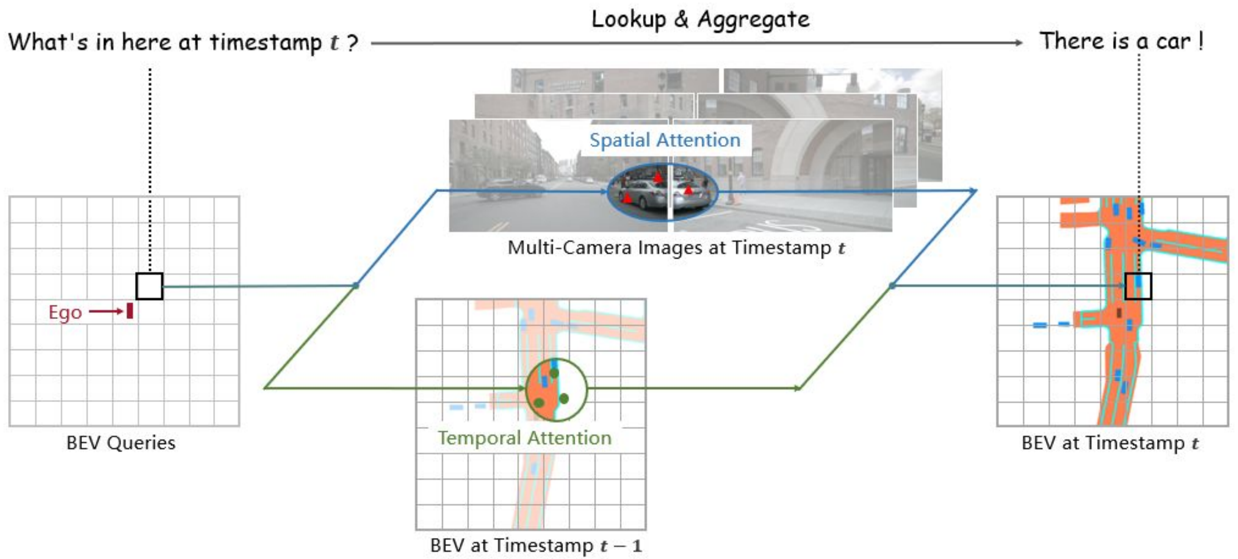
Nanjing University Shanghai AI Laboratory The University of Hong Kong

BEVFormer: Motivation

BEVFormer

A unified End-to-End framework which fuses multi-camera and temporal feature based on Deformable Attention and is suitable for various kinds of perception tasks in AD

Key Module



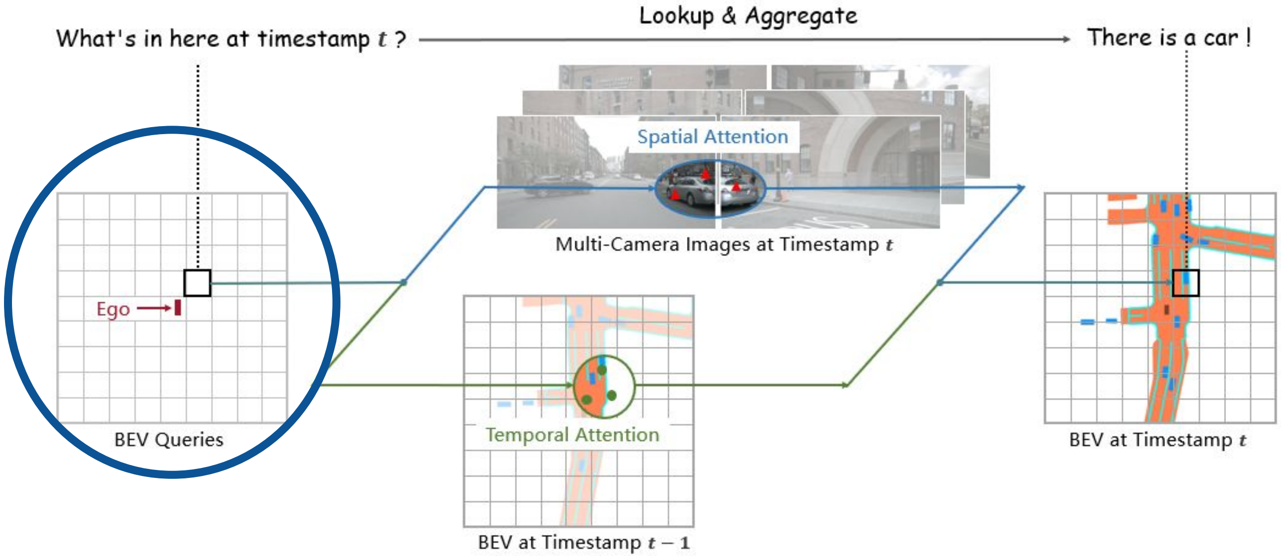
BEVFormer: Motivation

BEVFormer

A unified End-to-End framework which fuses multi-camera and temporal feature based on Deformable Attention and is suitable for various kinds of perception tasks in AD

Key Module

- **BEV Queries Q**: used for lookup to obtain BEV feature map



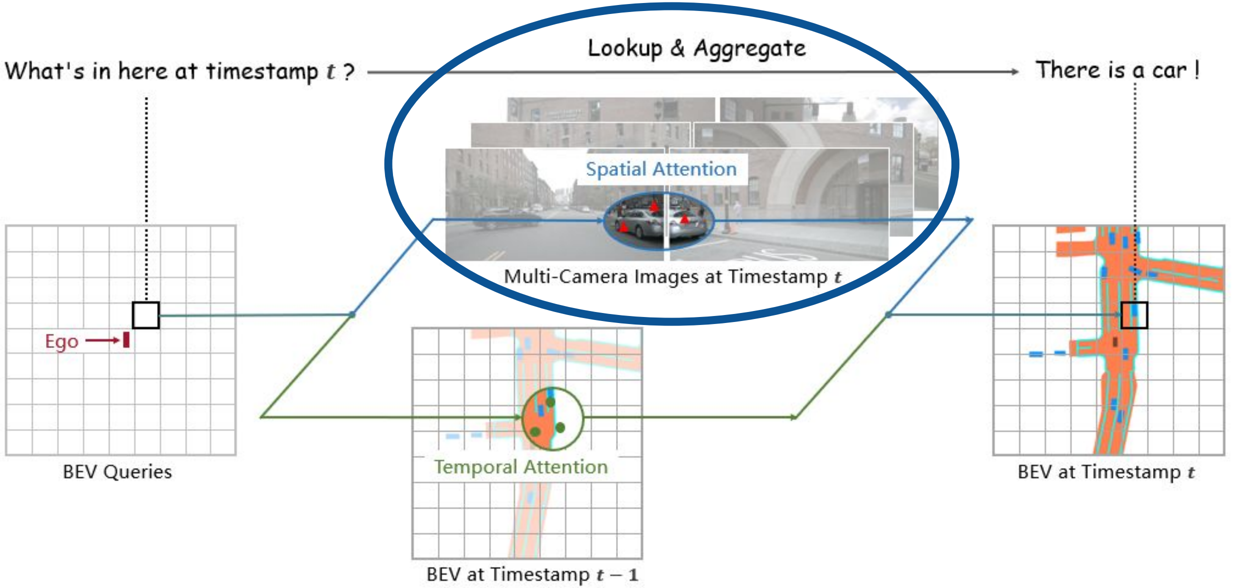
BEVFormer: Motivation

BEVFormer

A unified End-to-End framework which fuses multi-camera and temporal feature based on Deformable Attention and is suitable for various kinds of perception tasks in AD

Key Module

- **BEV Queries Q**: lookup to obtain BEV feature map
- **Spatial Cross-Attention**: fuse multi-camera feature



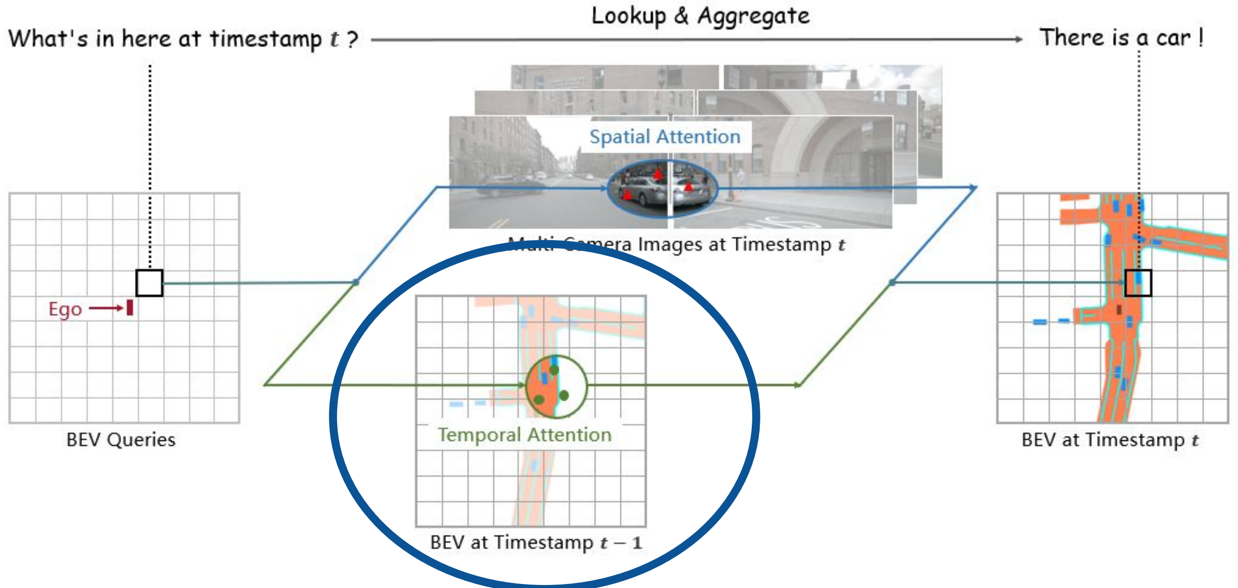
BEVFormer: Motivation

BEVFormer

A unified End-to-End framework which fuses multi-camera and temporal feature based on Deformable Attention and is suitable for various kinds of perception tasks in AD

Key Module

- **BEV Queries Q**: lookup to obtain BEV feature map
- **Spatial Cross-Attention**: fuse multi-camera feature
- **Temporal Self-Attention**: aggregate temporal BEV feature



BEVFormer: Motivation

BEVFormer

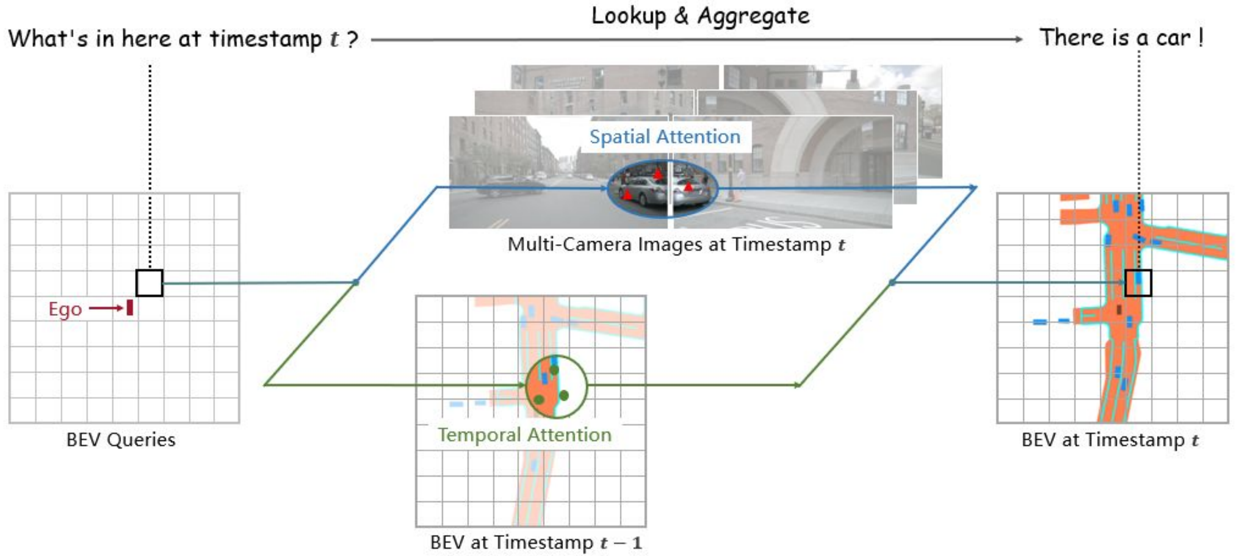
A unified End-to-End framework which fuses multi-camera and temporal feature based on Deformable Attention and is suitable for various kinds of perception tasks in AD

Key Module

- **BEV Queries Q** : lookup to obtain BEV feature map
- **Spatial Cross-Attention** : fuse multi-camera feature
- **Temporal Self-Attention** : aggregate temporal BEV feature

Keypoint

- Using **learnable** queries to represent real world from BEV view
- Look up spatial features in images and temporal features in previous BEV map, aka **Spatial-temporal**



BEVFormer: Overall Architecture

BEVFormer

A unified End-to-End framework which fuses multi-camera and temporal feature based on Deformable Attention and is suitable for various kinds of perception tasks in AD

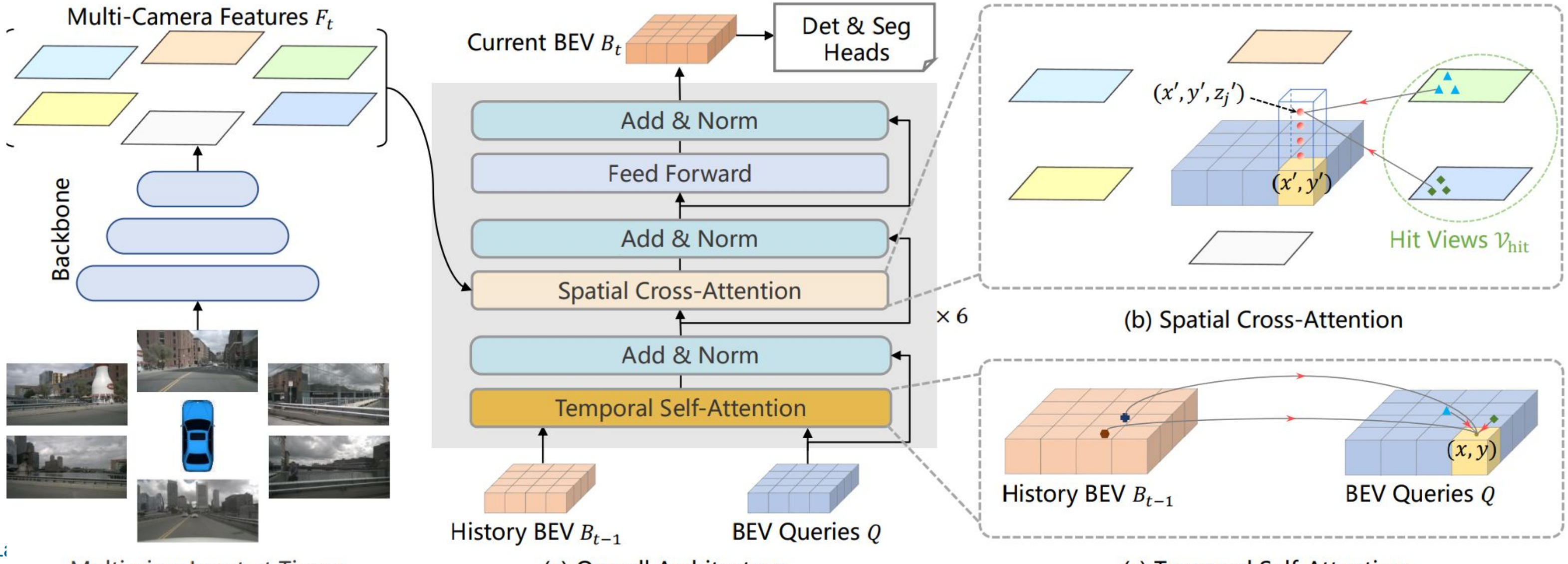
Key Module

- **BEV Queries Q**: lookup to obtain BEV feature map
- **Spatial Cross-Attention**: fuse multi-camera feature
- **Temporal Self-Attention**: aggregate temporal BEV feature

Keypoint

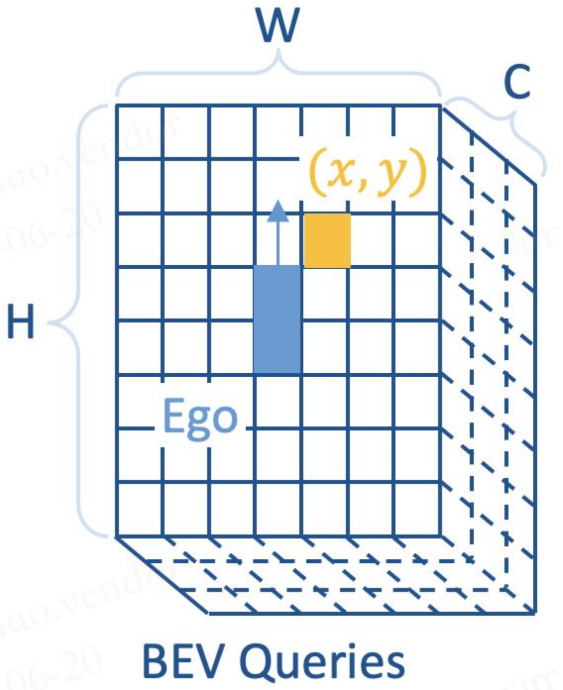
Using **learnable** queries to represent real world from BEV view

Look up spatial features in images and temporal features in previous BEV map, aka **Spatial-temporal**

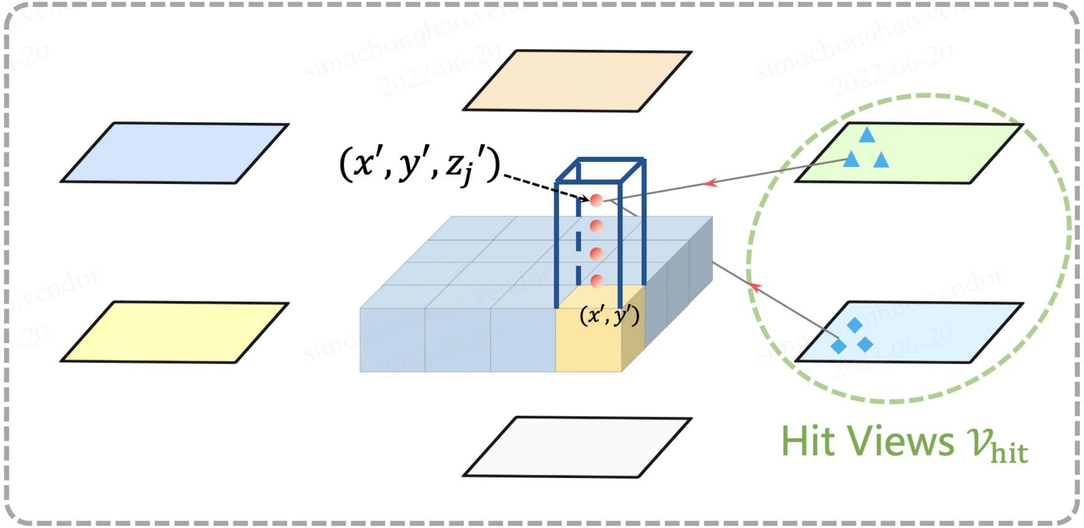


BEVFormer: BEV Queries

- BEV queries are H*W*C ***learnable parameter***, which are used to capture BEV feature surrounding *ego car*.
- Every query locating at position *(x, y)* is responsible for ***representing its corresponding small range of area***.
- Take turns to look up ***spatial*** and ***temporal information*** to generate BEV feature map.



BEVFormer: Spatial Cross-Attention



Spatial Cross-Attention

Look up spatial information

Concrete Steps

【Step 1】Lift each BEV query to be a ***pillar***

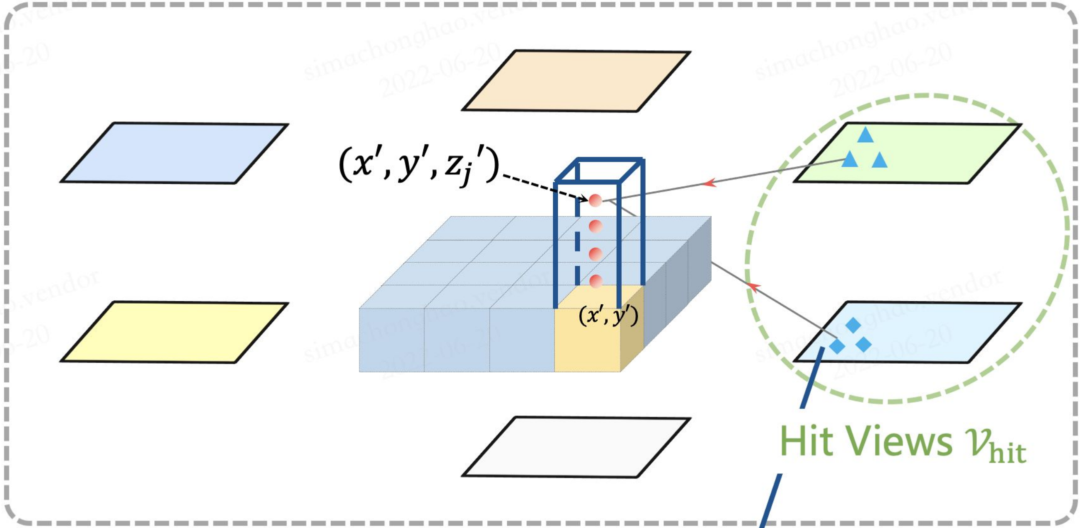
【Step 2】Project the ***3D points*** in pillar to ***2D points*** in views

【Step 3】Sample features from regions in *hit views*

【Step 4】Fuse by weight

[1] Deformable DETR: Deformable Transformers for End-to-End Object Detection, *ICLR 2021 Oral*

BEVFormer: Spatial Cross-Attention



Spatial Cross-Attention

Sparse Attention, e.g., Deformable Attention [1]

Look up spatial information

Concrete Steps

【Step 1】Lift each BEV query to be a ***pillar***

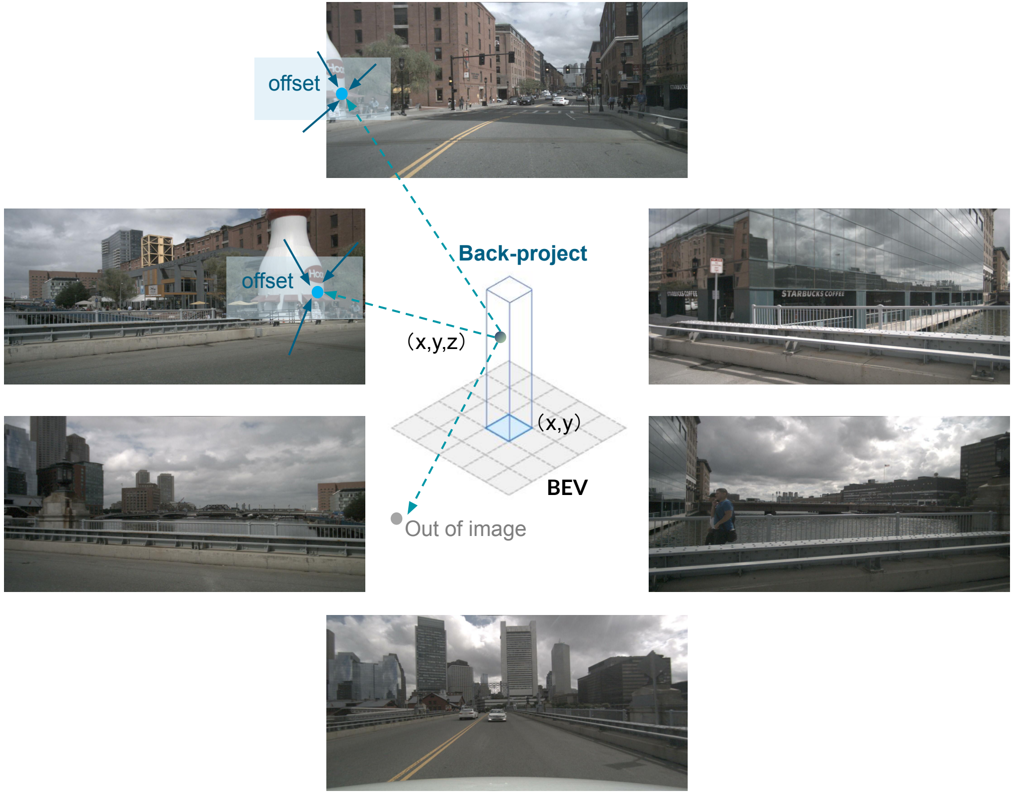
【Step 2】Project the ***3D points*** in pillar to ***2D points*** in views

【Step 3】Sample features from regions in *hit views*

【Step 4】Fuse by weight

[1] Deformable DETR: Deformable Transformers for End-to-End Object Detection, *ICLR 2021 Oral*

BEVFormer: Spatial Cross-Attention



Look up spatial information

Concrete Steps

【Step 1】Lift each BEV query to be a ***pillar***

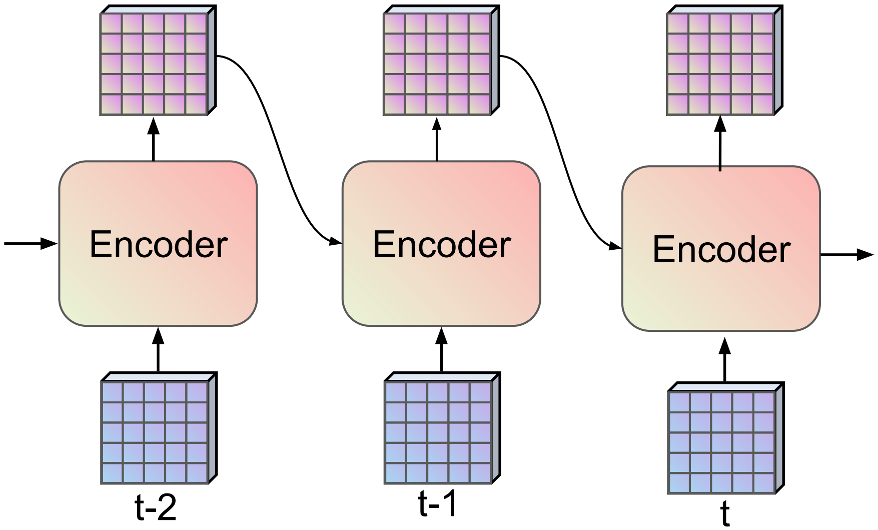
【Step 2】Project the ***3D points*** in pillar to ***2D points*** in views

【Step 3】Sample features from regions in *hit views*

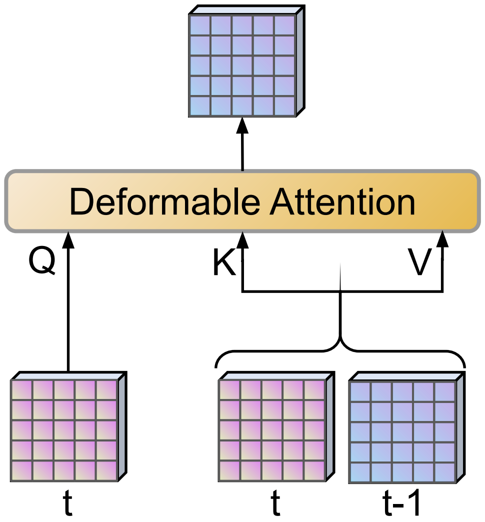
【Step 4】Fuse by weight

BEVFormer: Temporal Self-Attention

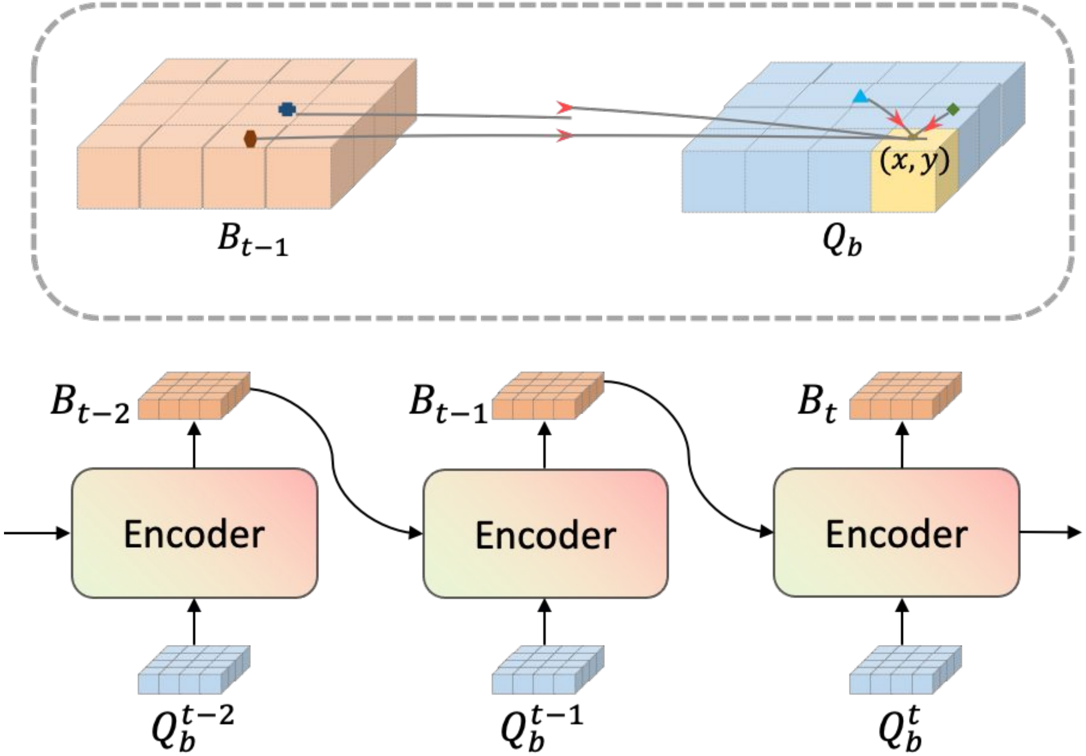
(Temporal feature modeling) recurrently model temporal BEV feature in the similar way to RNN. Every timestamp only requires last timestamp feature, which leads to lower computational cost.



(Temporal feature aggregation) current timestamp's BEV feature is the query while both current and last timestamp's BEV feature is the key and value based on Deformable Attention to aggregate temporal feature



BEVFormer: Temporal Self-Attention



Look up temporal information

Concrete Steps

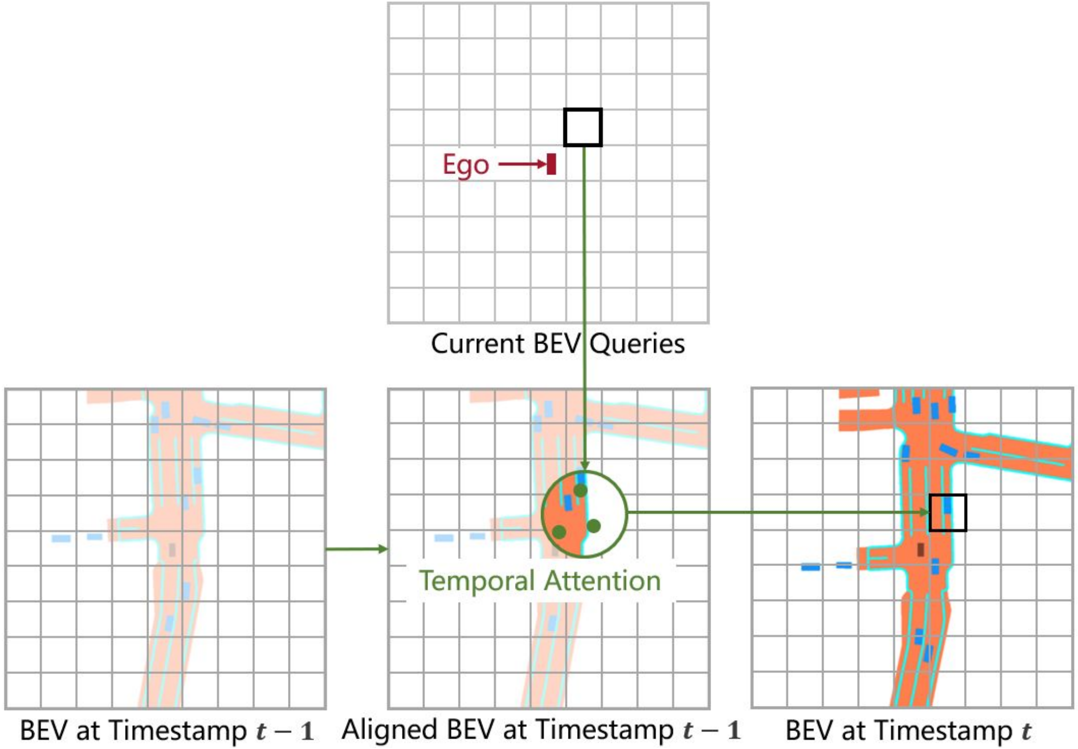
【Step 1】***align two bev feature map*** according to ego car's motion

【Step 2】sample feature ***from current timestamp and the past***

【Step 3】***compute the weighted sum*** of the sampled BEV feature

【Step 4】***Recursively collect*** historical BEV feature

BEVFormer: Temporal Self-Attention



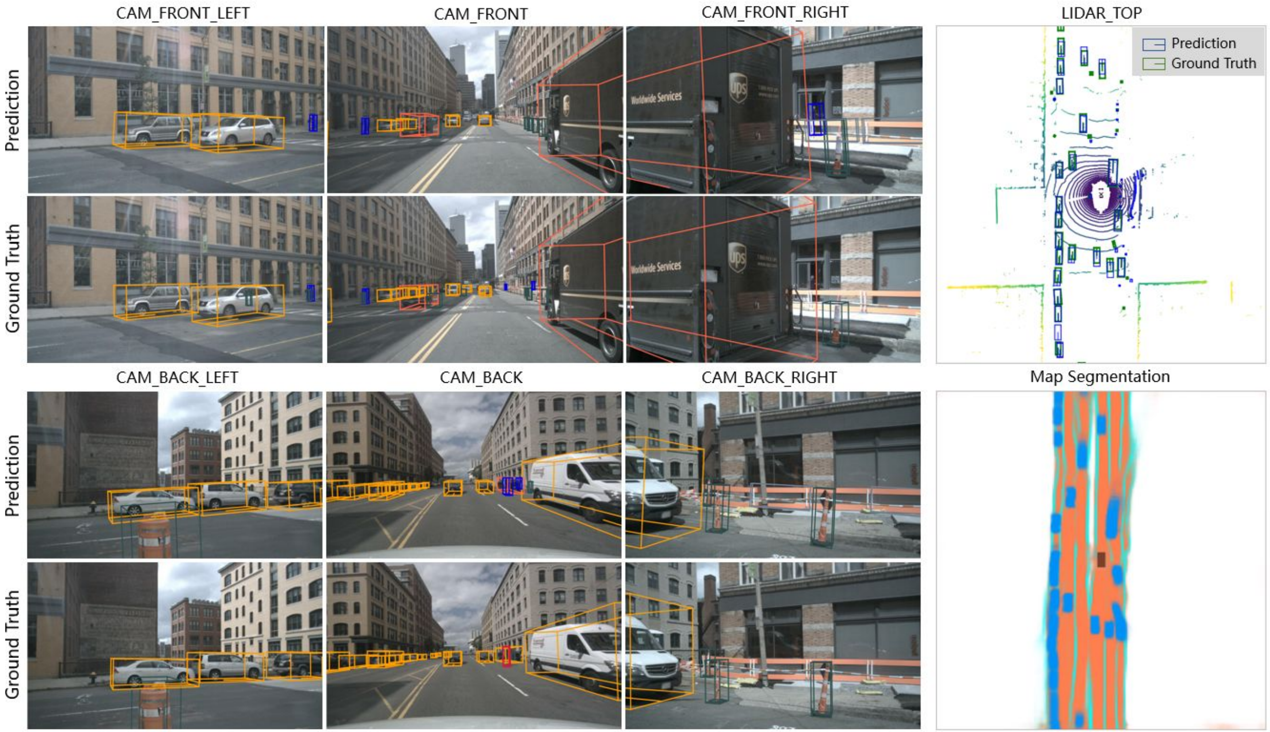
Concrete Steps

【Step 1】Given BEV feature map at timestamp t-1 B_{t-1}, based on ego-motion, align it to current BEV coordinates and denote it B'_{t-1}

【Step 2】Use Deformable Attention to perform cross attention on B'_{t-1} and current BEV query

BEVFormer: Explicit BEV feature

- Multi-task learning: 3D object detection and map semantic segmentation
- Transferability: commonly used 2D detection head can be transferred to 3D detection with minor modification



BEVFormer: Performance on nuScenes & Waymo 1.2

nuScenes *test set*, NDS : ***56.9 v.s. 47.9***

Table 1: **3D Detection Results on nuScenes `test` set.** ∗ notes that VoVNet-99 (V2-99) [21] was pre-trained on the depth estimation task with extra data [31]. “BEVFormer-S” does not leverage temporal information in the BEV encoder. “L” and “C” indicate LiDAR and Camera, respectively.

Method	Modality	Backbone	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
SSN [54]	L	-	0.569	0.463	-	-	-	-	-
CenterPoint-Voxel [51]	L	-	0.655	0.580	-	-	-	-	-
PointPainting [43]	L&C	-	0.581	0.464	0.388	0.271	0.496	0.247	0.111
FCOS3D [45]	C	R101	0.428	0.358	0.690	0.249	0.452	1.434	**0.124**
PGD [44]	C	R101	0.448	0.386	**0.626**	**0.245**	0.451	1.509	0.127
BEVFormer-S	C	R101	0.462	0.409	0.650	0.261	0.439	0.925	0.147
BEVFormer	C	R101	**0.535**	**0.445**	0.631	0.257	**0.405**	**0.435**	0.143
DD3D [31]	C	V2-99*	0.477	0.418	**0.572**	**0.249**	**0.368**	1.014	**0.124**
DETR3D [47]	C	V2-99*	0.479	0.412	0.641	0.255	0.394	0.845	0.133
BEVFormer-S	C	V2-99*	0.495	0.435	0.589	0.254	0.402	0.842	0.131
BEVFormer	C	V2-99*	**0.569**	**0.481**	0.582	0.256	0.375	**0.378**	0.126

Waymo 1.2 *val set*, L1/APH : ***28.0 v.s. 22.0***

Table 3: **3D Detection Results on Waymo `val` set under Waymo evaluation metric and nuScenes evaluation metric.** “L1” and “L2” refer “LEVEL_1” and “LEVEL_2” difficulties of Waymo [40]. *: Only use the front camera and only consider object labels in the front camera’s field of view (50.4°). †: We compute the NDS score by setting ATE and AAE to be 1. “L” and “C” indicate LiDAR and Camera, respectively.

Method	Modality	Waymo Metrics				Nuscenes Metrics				
		IoU=0.5		IoU=0.7						
		L1/APH	L2/APH	L1/APH	L2/APH	NDS†↑	AP↑	ATE↓	ASE↓	AOE↓
PointPillars [20]	L	0.866	0.801	0.638	0.557	0.685	0.838	0.143	0.132	0.070
DETR3D [47]	C	0.220	0.216	0.055	0.051	0.394	0.388	0.741	**0.156**	0.108
BEVFormer	C	**0.280**	**0.241**	**0.061**	**0.052**	**0.426**	**0.440**	**0.679**	0.157	**0.101**
CaDNN* [34]	C	0.175	0.165	0.050	0.045	-	-	-	-	-
BEVFormer*	C	0.308	0.277	0.077	0.069	-	-	-	-	-

BEVFormer: Performance on nuScenes & Waymo 1.2

nuScenes *test set*, NDS : ***56.9 v.s. 47.9***

Table 1: **3D Detection Results on nuScenes** test **set.** ∗ notes that VoVNet-99 (V2-99) [21] was pre-trained on the depth estimation task with extra data [31]. "BEVFormer-S" does not leverage temporal information in the BEV encoder. "L" and "C" indicate LiDAR and Camera, respectively.

Method	Modality	Backbone	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
SSN [54]	L	-	0.569	0.463	-	-	-	-	-
CenterPoint-Voxel [51]	L	-	0.655	0.580	-	-	-	-	-
PointPainting [43]	L&C	-	0.581	0.464	0.388	0.271	0.496	0.247	0.111
FCOS3D [45]	C	R101	0.428	0.358	0.690	0.249	0.452	1.434	**0.124**
PGD [44]	C	R101	0.448	0.386	**0.626**	**0.245**	0.451	1.509	0.127
BEVFormer-S	C	R101	0.462	0.409	0.650	0.261	0.439	0.925	0.147
BEVFormer	C	R101	**0.535**	**0.445**	0.631	0.257	**0.405**	**0.435**	0.143
DD3D [31]	C	V2-99*	0.477	0.418	**0.572**	**0.249**	**0.368**	1.014	**0.124**
DETR3D [47]	C	V2-99*	0.479	0.412	0.641	0.255	0.394	0.845	0.133
BEVFormer-S	C	V2-99*	0.495	0.435	0.589	0.254	0.402	0.842	0.131
BEVFormer	C	V2-99*	**0.569**	**0.481**	0.582	0.256	0.375	**0.378**	0.126

Waymo 1.2 *val set*, L1/APH : ***28.0 v.s. 22.0***

Table 3: **3D Detection Results on Waymo** val **set under Waymo evaluation metric and nuScenes evaluation metric.** "L1" and "L2" refer "LEVEL_1" and "LEVEL_2" difficulties of Waymo [40]. *: Only use the front camera and only consider object labels in the front camera's field of view (50.4°). †: We compute the NDS score by setting ATE and AAE to be 1. "L" and "C" indicate LiDAR and Camera, respectively.

Method	Modality	Waymo Metrics IoU=0.5 L1/APH	Waymo Metrics IoU=0.5 L2/APH	Waymo Metrics IoU=0.7 L1/APH	Waymo Metrics IoU=0.7 L2/APH	Nuscenes Metrics NDS†↑	AP↑	ATE↓	ASE↓	AOE↓
PointPillars [20]	L	0.866	0.801	0.638	0.557	0.685	0.838	0.143	0.132	0.070
DETR3D [47]	C	0.220	0.216	0.055	0.051	0.394	0.388	0.741	**0.156**	0.108
BEVFormer	C	**0.280**	**0.241**	**0.061**	**0.052**	**0.426**	**0.440**	**0.679**	0.157	**0.101**
CaDNN* [34]	C	0.175	0.165	0.050	0.045	-	-	-	-	-
BEVFormer*	C	0.308	0.277	0.077	0.069	-	-	-	-	-

Conclusion of ablation study

- ***A strong backbone network*** is important.

BEVFormer: Performance on nuScenes & Waymo 1.2

nuScenes *test set*, NDS : ***56.9 v.s. 47.9***

Waymo 1.2 *val set*, L1/APH : ***28.0 v.s. 22.0***

Table 1: **3D Detection Results on nuScenes test set.** ∗ notes that VoVNet-99 (V2-99) [21] was pre-trained on the depth estimation task with extra data [31]. "BEVFormer-S" does not leverage temporal information in the BEV encoder. "L" and "C" indicate LiDAR and Camera, respectively.

Method	Modality	Backbone	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
SSN [54]	L	-	0.569	0.463	-	-	-	-	-
CenterPoint-Voxel [51]	L	-	0.655	0.580	-	-	-	-	-
PointPainting [43]	L&C	-	0.581	0.464	0.388	0.271	0.496	0.247	0.111
FCOS3D [45]	C	R101	0.428	0.358	0.690	0.249	0.452	1.434	**0.124**
PGD [44]	C	R101	0.448	0.386	**0.626**	**0.245**	0.451	1.509	0.127
BEVFormer-S	C	R101	0.462	0.409	0.650	0.261	0.439	0.925	0.147
BEVFormer	C	R101	**0.535**	**0.445**	0.631	0.257	**0.405**	**0.435**	0.143
DD3D [31]	C	V2-99*	0.477	0.418	**0.572**	**0.249**	**0.368**	1.014	**0.124**
DETR3D [47]	C	V2-99*	0.479	0.412	0.641	0.255	0.394	0.845	0.133
BEVFormer-S	C	V2-99*	0.495	0.435	0.589	0.254	0.402	0.842	0.131
BEVFormer	C	V2-99*	**0.569**	**0.481**	0.582	0.256	0.375	**0.378**	0.126

Table 3: **3D Detection Results on Waymo val set under Waymo evaluation metric and nuScenes evaluation metric.** "L1" and "L2" refer "LEVEL_1" and "LEVEL_2" difficulties of Waymo [40]. *: Only use the front camera and only consider object labels in the front camera's field of view (50.4°). †: We compute the NDS score by setting ATE and AAE to be 1. "L" and "C" indicate LiDAR and Camera, respectively.

Method	Modality	Waymo Metrics IoU=0.5 L1/APH	Waymo Metrics IoU=0.5 L2/APH	Waymo Metrics IoU=0.7 L1/APH	Waymo Metrics IoU=0.7 L2/APH	Nuscenes Metrics NDS†↑	AP↑	ATE↓	ASE↓	AOE↓
PointPillars [20]	L	0.866	0.801	0.638	0.557	0.685	0.838	0.143	0.132	0.070
DETR3D [47]	C	0.220	0.216	0.055	0.051	0.394	0.388	0.741	**0.156**	0.108
BEVFormer	C	**0.280**	**0.241**	**0.061**	**0.052**	**0.426**	**0.440**	**0.679**	0.157	**0.101**
CaDNN* [34]	C	0.175	0.165	0.050	0.045	-	-	-	-	-
BEVFormer*	C	0.308	0.277	0.077	0.069	-	-	-	-	-

Conclusion of ablation study

- ***A strong backbone network*** is important.
- ***Local attention*** is better than global attention(~4.4 in NDS)
- ***Temporal cues*** are important(yields higher recall rate and more accurate speed estimation)
- ***Multi-task learning*** improves the performance of 3D object detection but decreases the performance of BEV map segmentation on the other hand

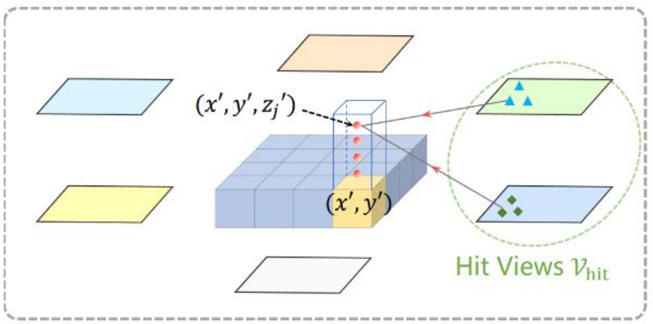
BEVFormer: Ablation on Attention Module

Table 5: The detection results of different methods with **various BEV encoders** on nuScenes val set.

Method	Attention	NDS↑	mAP↑	mATE↓	mAOE↓	#Param.	FLOPs	Memory
VPN* [30]	-	0.334	0.252	0.926	0.598	111.2M	924.5G	~20G
List-Splat* [32]	-	0.397	0.348	0.784	0.537	74.0M	1087.7G	~20G
BEVFormer-S†	Global	0.404	0.325	0.837	0.442	62.1M	1245.1G	~36G
BEVFormer-S‡	Points	0.423	0.351	0.753	0.442	68.1M	1264.3G	~20G
BEVFormer-S	Local	**0.448**	**0.375**	**0.725**	**0.391**	68.7M	1303.5G	~20G

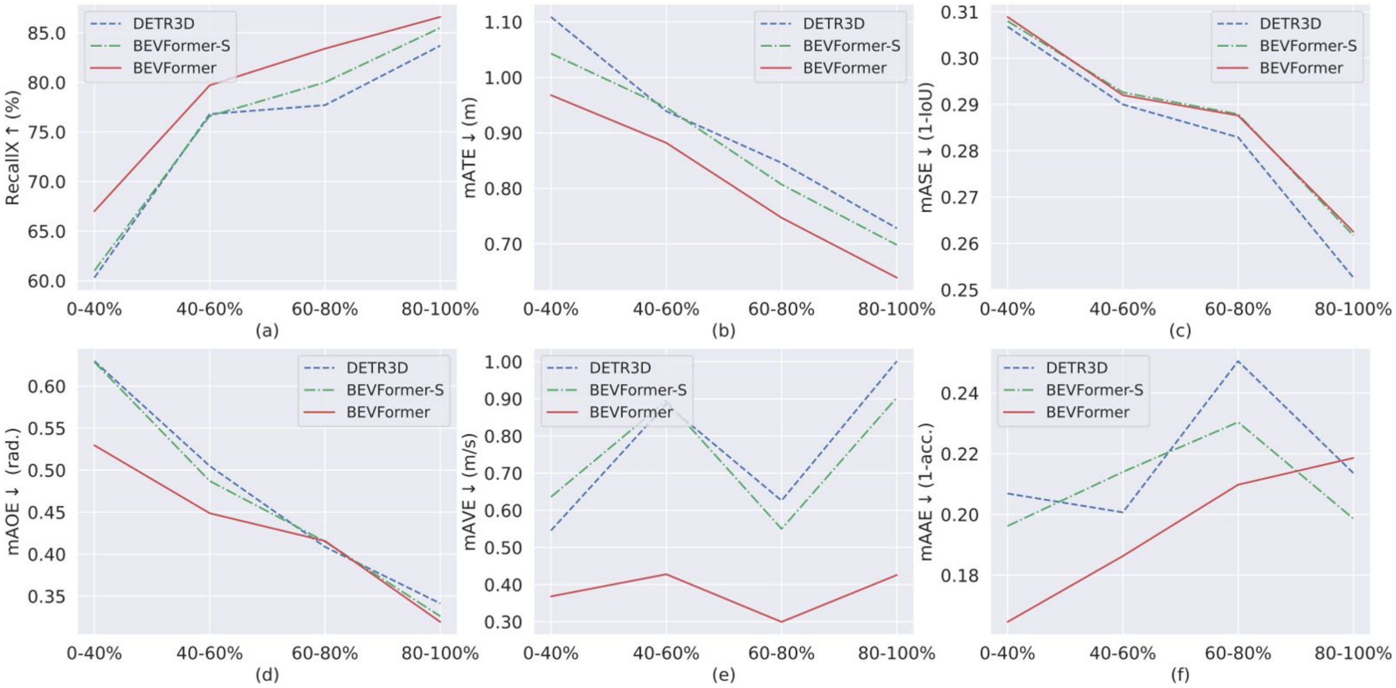
Conclusion of ablation study

- Global attention requires more computational resources
- The receptive field of point interaction is limited
- Deformable attention can strike the balance between computational cost and receptive field.



(b) Spatial Cross-Attention

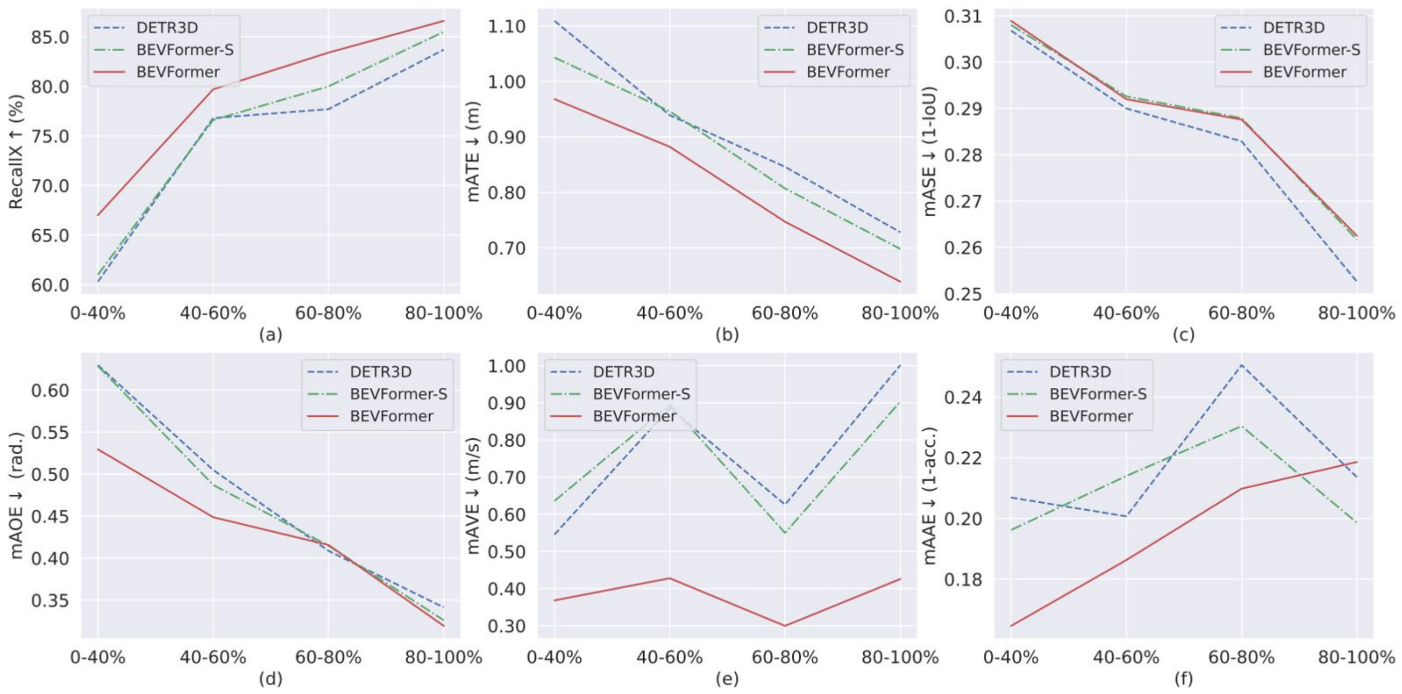
BEVFormer: Ablation on Temporal Clues



visibility that {0-40%, 40-60%, 60-80%, 80-100%} of objects can be visible

- mATE: mean Average Translation Error
- mASE: mean Average Size Error
- mAOE: mean Average Orientation Error
- mAVE: mean Average Velocity Error
- mAAE: mean Average Attribute Error

BEVFormer: Ablation on Temporal Clues



Via temporal cues,

We have:

- **Higher recall rate,** especially for those object with **low visibility**
- More accurate **position estimation**
- More accurate **speed estimation**

visibility that {0-40%, 40-60%, 60-80%, 80-100%} of objects can be visible

BEVFormer: Ablation on Multi-task Learning

Table 4: **The Results on 3D detection and map segmentation task.** Comparison of training segmentation and detection tasks jointly or not. *: We use VPN [30] and Lift-Splat [32] to replace our BEV encoder for comparison, and the task heads are the same. †: Results from their paper.

Method	Task Head		3D Detection		BEV Segmentation (IoU)			
	Det	Seg	NDS↑	mAP↑	Car	Vehicles	Road	Lane
Lift-Splat† [32]	✗	✓	-	-	32.1	32.1	72.9	20.0
FIERY† [18]	✗	✓	-	-	-	38.2	-	-
VPN* [30]	✓	✗	0.333	0.253	-	-	-	-
VPN*	✗	✓	-	-	31.0	31.8	76.9	19.4
VPN*	✓	✓	0.334	0.257	36.6	37.3	76.0	18.0
Lift-Splat*	✓	✗	0.397	0.348	-	-	-	-
Lift-Splat*	✗	✓	-	-	42.1	41.7	77.7	20.0
Lift-Splat*	✓	✓	0.410	0.344	43.0	42.8	73.9	18.3
BEVFormer-S	✓	✗	0.448	0.375	-	-	-	-
BEVFormer-S	✗	✓	-	-	43.1	43.2	**80.7**	21.3
BEVFormer-S	✓	✓	0.453	0.380	44.3	44.4	77.6	19.8
BEVFormer	✓	✗	0.517	**0.416**	-	-	-	-
BEVFormer	✗	✓	-	-	44.8	44.8	80.1	**25.7**
BEVFormer	✓	✓	**0.520**	0.412	**46.8**	**46.7**	77.5	23.9

Through multi-task learning, we have:

- A multi-task head with higher NDS
- Lower IoU of road and lane

Join the BEVFormer **community**!

BEVFormer Public

[ECCV 2022] This is the official implementation of BEVFormer, a camera-only framework for autonomous driving perception, e.g., 3D object detection and semantic map segmentation.

Python ☆ 1.2k 143

zhiqi-li Zhiqi Li
whai362 Wenhai Wang
hli2020 Hongyang Li

Github | Just **Google** BEVFormer
Zhihu Discussions | 自动驾驶BEV感知的下一步是什么
自动驾驶BEV感知的下一步是什么？

Chat with us in a professional way.

微信扫一扫