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

Zhiqi Li[1,2*], **Wenhai Wang**[2*], **Hongyang Li**[2*], **Enze Xie**[3], **Chonghao Sima**[2], **Tong Lu**[1], **Yu Qiao**[2], **Jifeng Dai**[2✉]

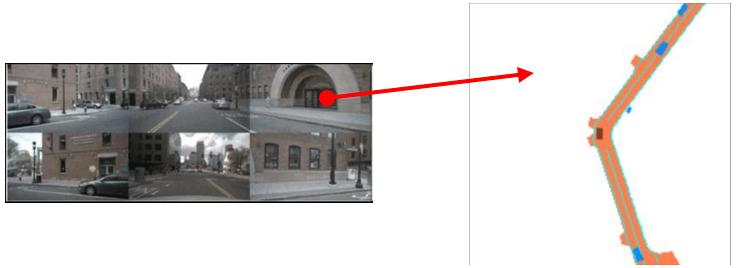
[1]Nanjing University [2]Shanghai AI Laboratory [3]The University of Hong Kong

ECCV 2022

세미나 발표자 : 김형균 (PseudoLab 10th)

Relationship

Bottom-Up (2D → 3D)



LSS
(Philion et al.)
2020-08

2021

2022

BEVDet
(Huang et al.)
2022-06

BEVFormer
(Li et al.)
2022-07

Simple-BEV
(Adam et al.)
2022-09

BEVDepth
(Yinhao et al.)
2022-11

PETRv2
(Liu et al.)
2022-11

2023

FB-BEV
(Zhiqi et al.)
2023-08

2024

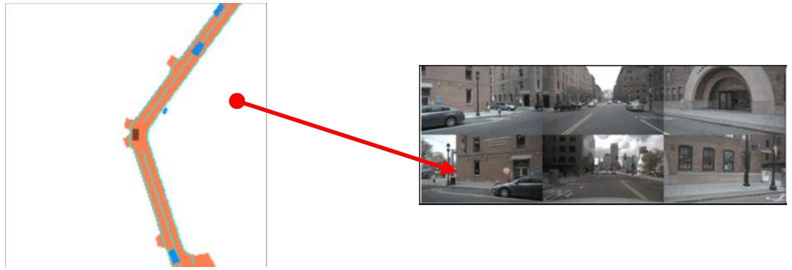
PointBeV
(Loick et al.)
2024-05

BEVFusion
(Zhijian et al.)
2024-09

GaussianBeV
(Florian et al.)
2024-12

2025

Top-Down (3D → 2D)



Colored : Camera-only
Bold : Attention mechanism

Ref. Ma, Yuexin, et al. "Vision-centric bev perception: A survey." IEEE Transactions on Pattern Analysis and Machine Intelligence (2024).

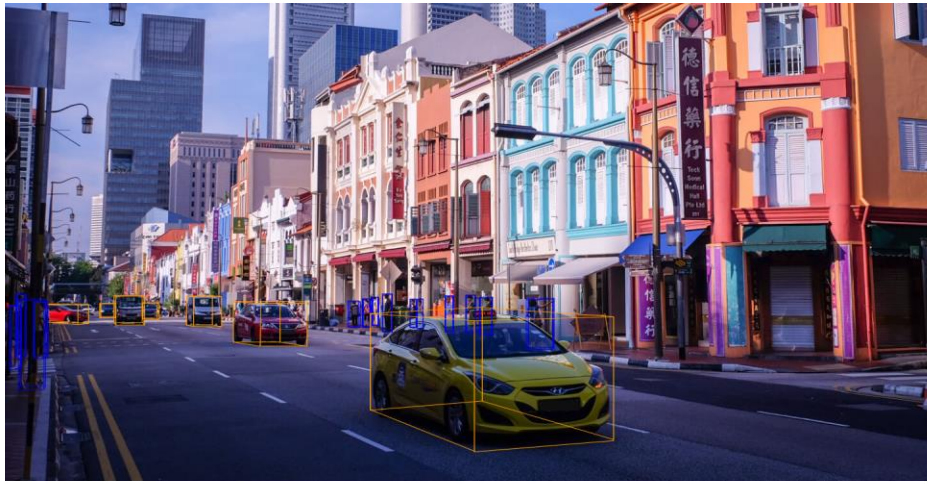
Agenda

- Problem Definition
- Methodology: *BEVFormer*
- Experiment Results
- Key takeaways

Problem Definition

Problem Definition

- **Multi-Camera** 기반 3D Perception: **Bird's-Eye View (BEV) Representation**
  - Lidar 센서에 비해 저렴한 가격
  - 원거리 및 고해상도 정보 습득 가능
  - 주행에 필요한 Vision-based 이미지 감지 가능
    (e.g., 주행 신호, 정지선, 횡단보도 등)



Camera-view 이미지를 통한 Object detection 과 Segmentation task

Figure : Caesar, Holger, et al. "nuscenes: A multimodal dataset for autonomous driving." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

Problem Definition

기존의 Multi-Camera BEV Representation Models

1. Depth Information 기반
   - e.g., *Lift-Splat-Shoot, BEVDet, …*
   - Bottom-Up Method (2D → 3D)
   - Depth Estimation에서 누적된 오차가 Downstream Task 성능 저하에 큰 영향

2. Temporal Information 사용 X
   - Temporal Information의 활용 가능성
     - 움직이는 물체 추적
     - 가려진 물체 추론
     - 물체의 속도 추론

Problem Definition

***BEVFormer**: Multi-Camera BEV Representation via Spatiotemporal Transformer*

1. Depth Information 기반
   - e.g., *Lift-Splat-Shoot, BEVDet, and so on*
   - Bottom-Up Method (2D → 3D)
   - Depth Estimation에서 누적된 오차가 Downstream Task 성능 저하에 큰 영향

Spatial Cross-Attention

2. Temporal Information 사용 X
   - Temporal Information의 활용 가능성
     - 움직이는 물체 추적
     - 가려진 물체 추론
     - 물체의 속도 추론

Problem Definition

***BEVFormer**: Multi-Camera BEV Representation via Spatiotemporal Transformer*

1. Depth Information 기반
   - e.g., *Lift-Splat-Shoot, BEVDet, and so on*
   - Bottom-Up Method (2D -> 3D)
   - Depth Estimation에서 누적된 오차가 Downstream Task 성능 저하에 큰 영향

Spatial Cross-Attention

2. Temporal Information 사용 X
   - Temporal Information의 활용 가능성
     - 움직이는 물체 추적
     - 가려진 물체 추론
     - 물체의 속도 추론

Temporal Self-Attention

Problem Definition

***BEVFormer**: Multi-Camera BEV Representation via Spatiotemporal Transformer*

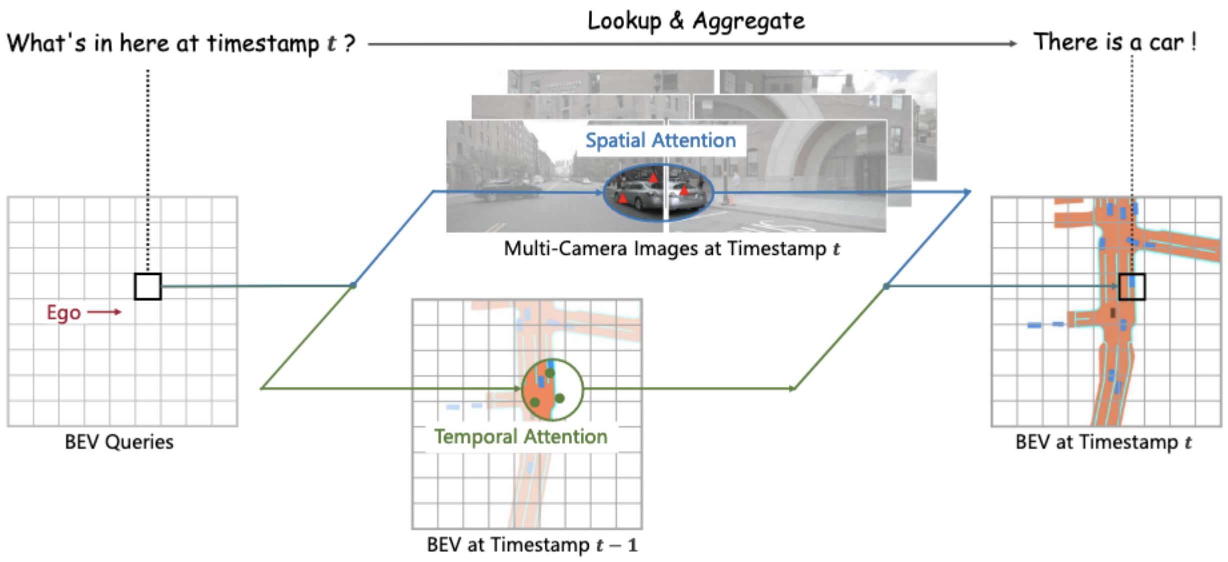
1. Depth Information 기반
   - e.g., *Lift-Splat-Shoot, BEVDet, and so on*
   - Bottom-Up Method (2D → 3D)
   - Depth Estimation에서 누적된 오차가 Downstream Task 성능 저하에 큰 영향

Spatial Cross-Attention

2. Temporal Information 사용 X
   - Temporal Information의 활용 가능성
     - 움직이는 물체 추적
     - 가려진 물체 추론
     - 물체의 속도 추론

Temporal Self-Attention

Grid-shaped BEV Queries

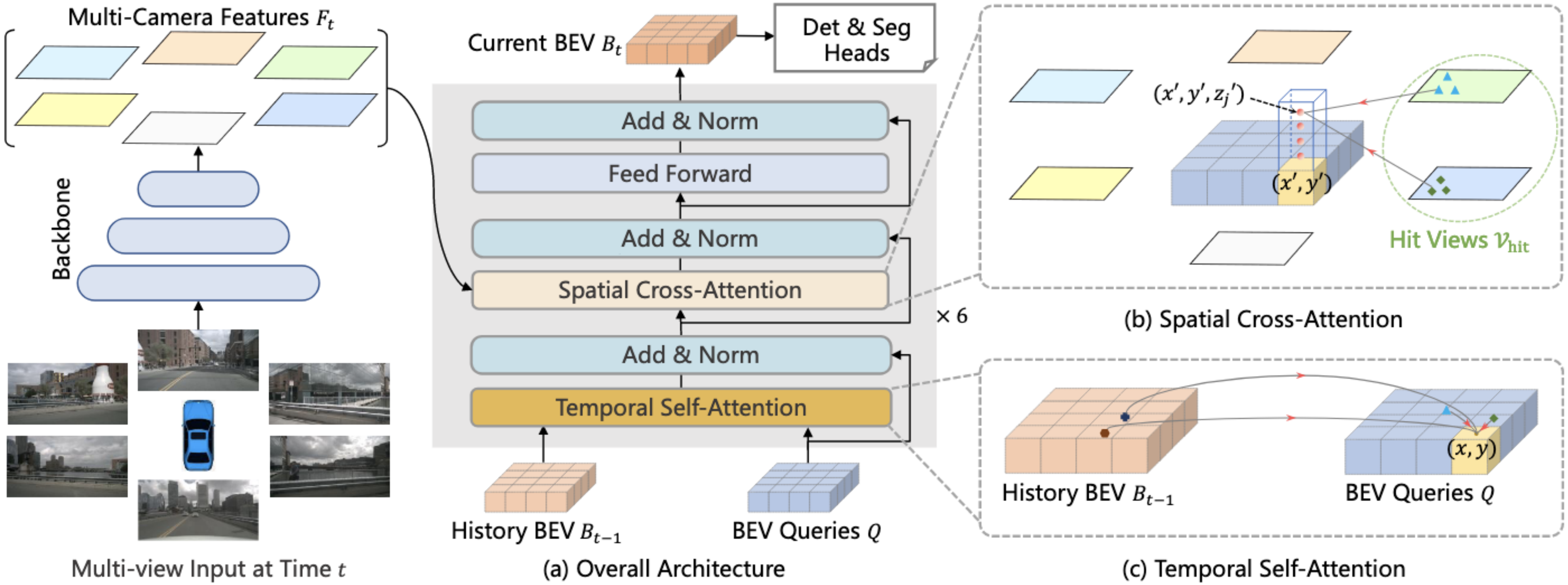


Methodology

BEVFormer: Multi-Camera BEV Representation via Spatiotemporal Transformer

Methodology - BEVFormer

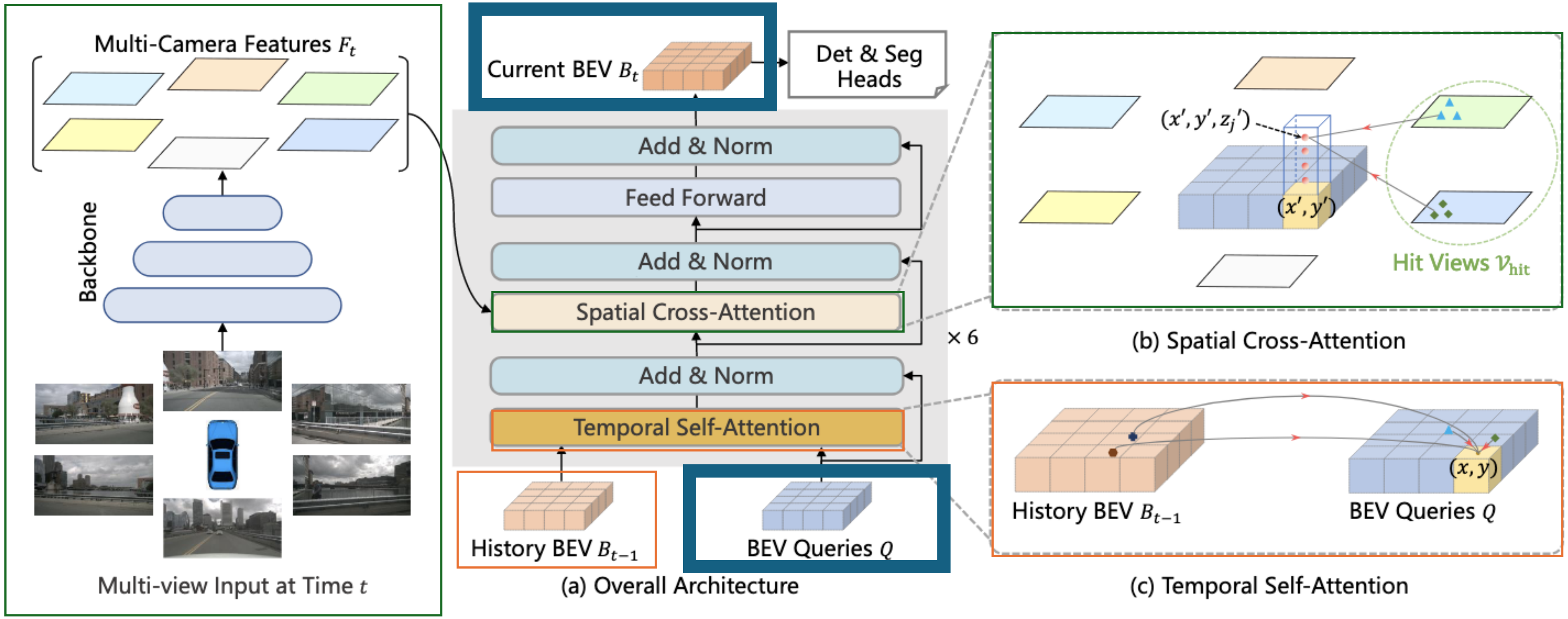
- Overview



Methodology - BEVFormer

- Overview

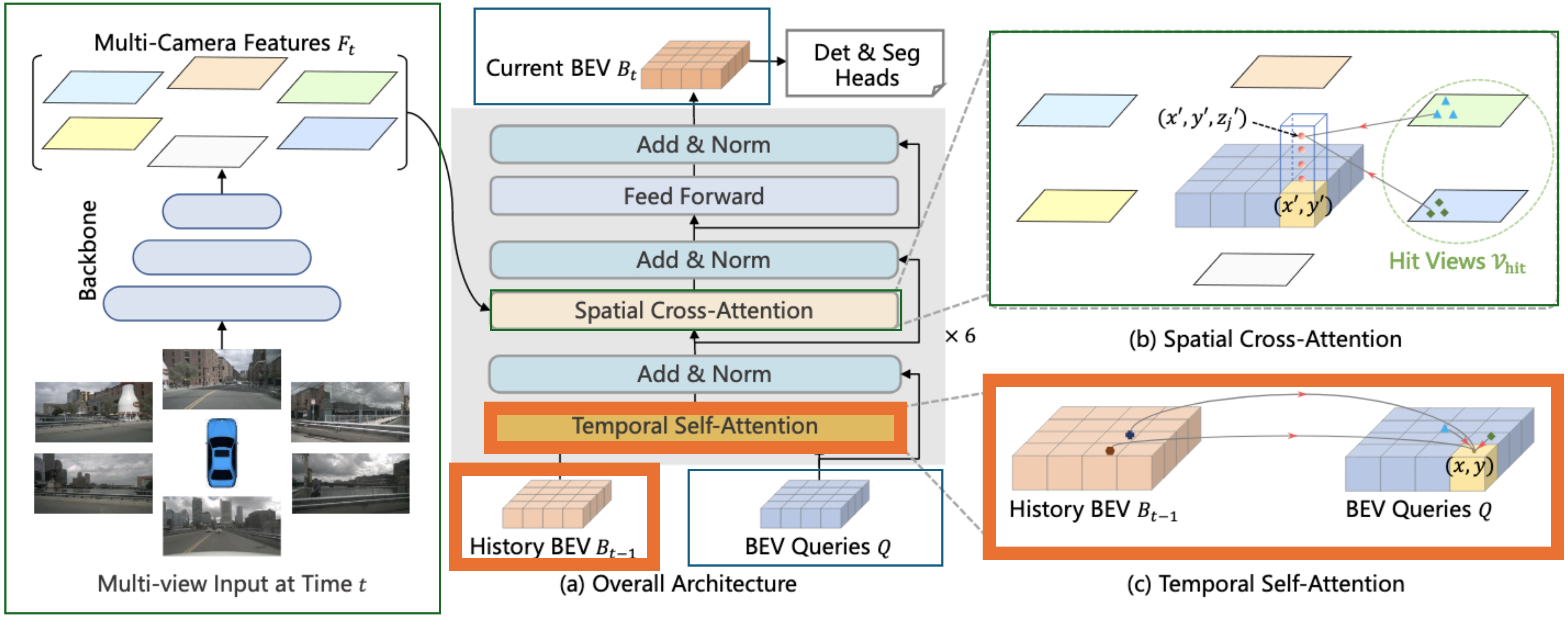
Grid-shaped BEV queries | Temporal Self-Attention | Spatial Cross-Attention



Methodology - BEVFormer

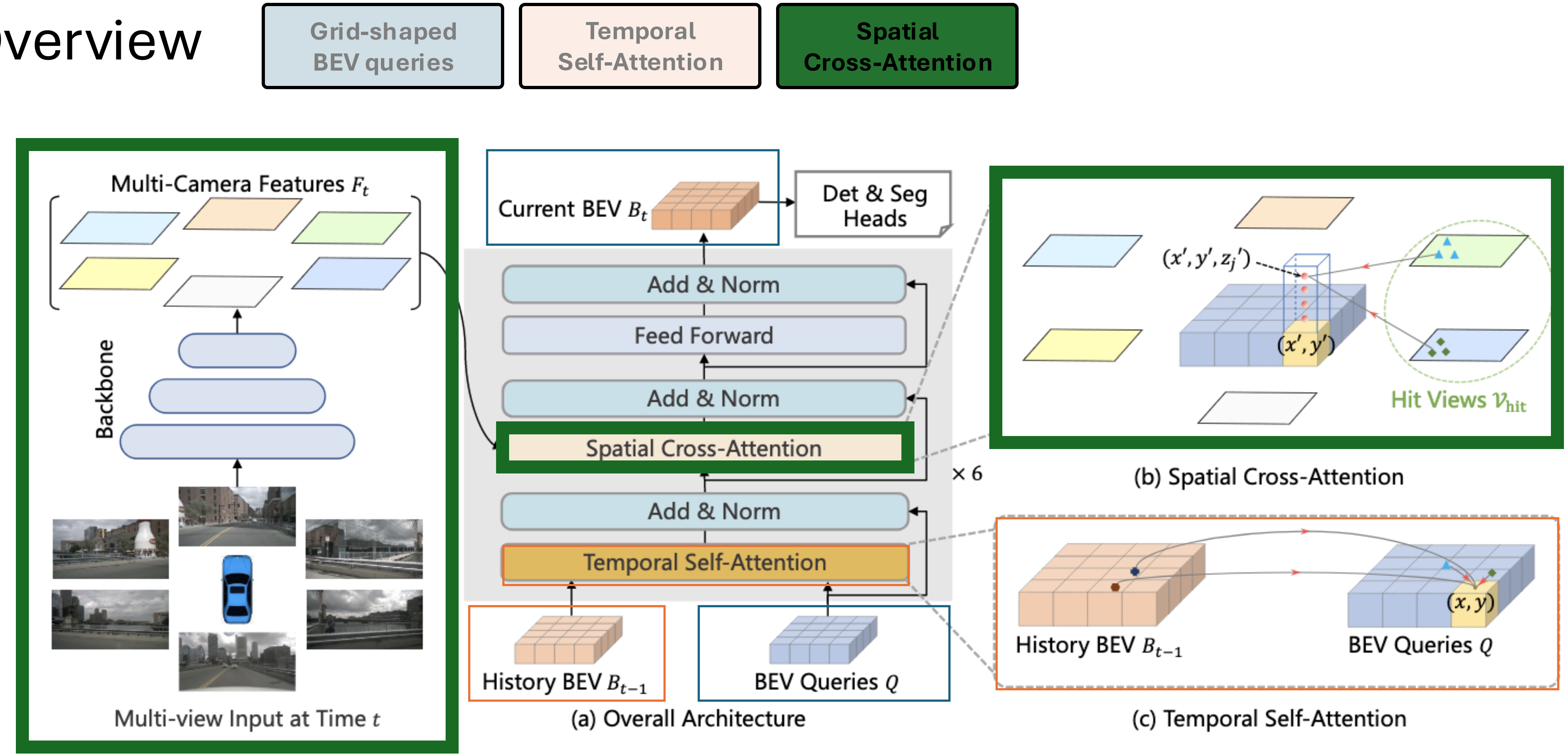
- Overview

Grid-shaped BEV queries	**Temporal Self-Attention**	Spatial Cross-Attention

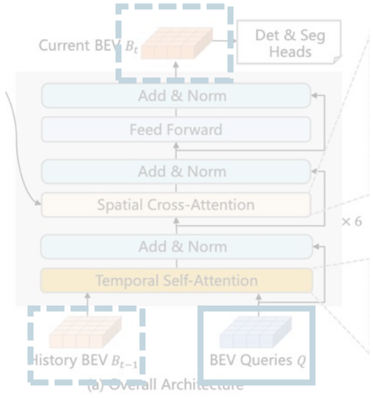


Methodology - BEVFormer

- Overview



Methodology - BEVFormer

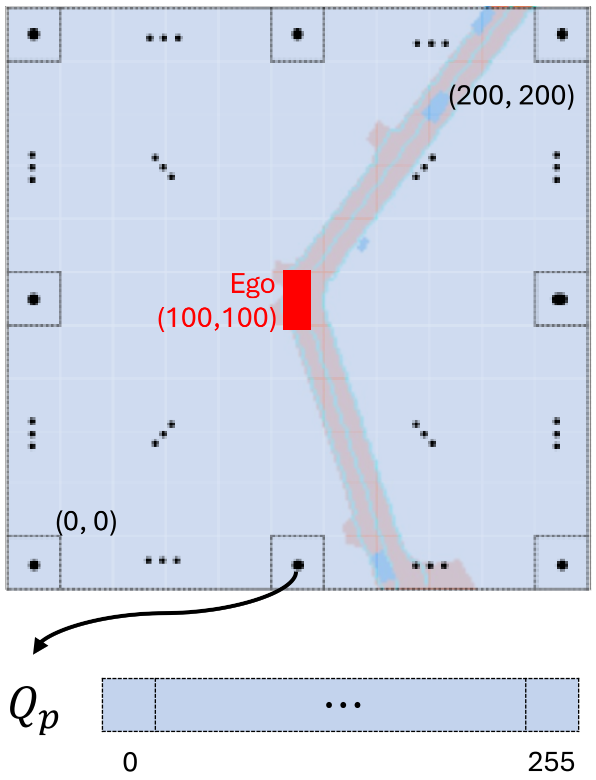


- Grid-shaped BEV Queries
  - BEV Queries(Q) 와 BEV Features(B)의 크기는 서로 동일
  - $Q \in \mathbb{R}^{H\times W\times C}(H : Grid\ height, W: Grid\ width, C: Features)$
    각 Grid point P=(x, y)에서의 Query feature : $Q_p \in \mathbb{R}^{1\times C}$
  - 각 Grid point와 Ego Vehicle 간의 실제 거리

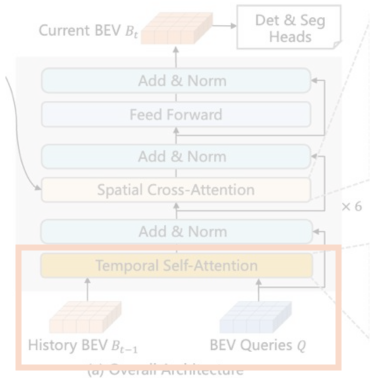
    $$x' = \left(x - \frac{W}{2}\right) \times s,\ y' = \left(y - \frac{H}{2}\right) \times s$$

    e.g., W = 200, H = 200, s = 0.512m,
    Grid 좌표 (100,100) → 실제 (0m, 0m)
    Grid 좌표 (90,120) → 실제 (-5.12m, 10.24m)

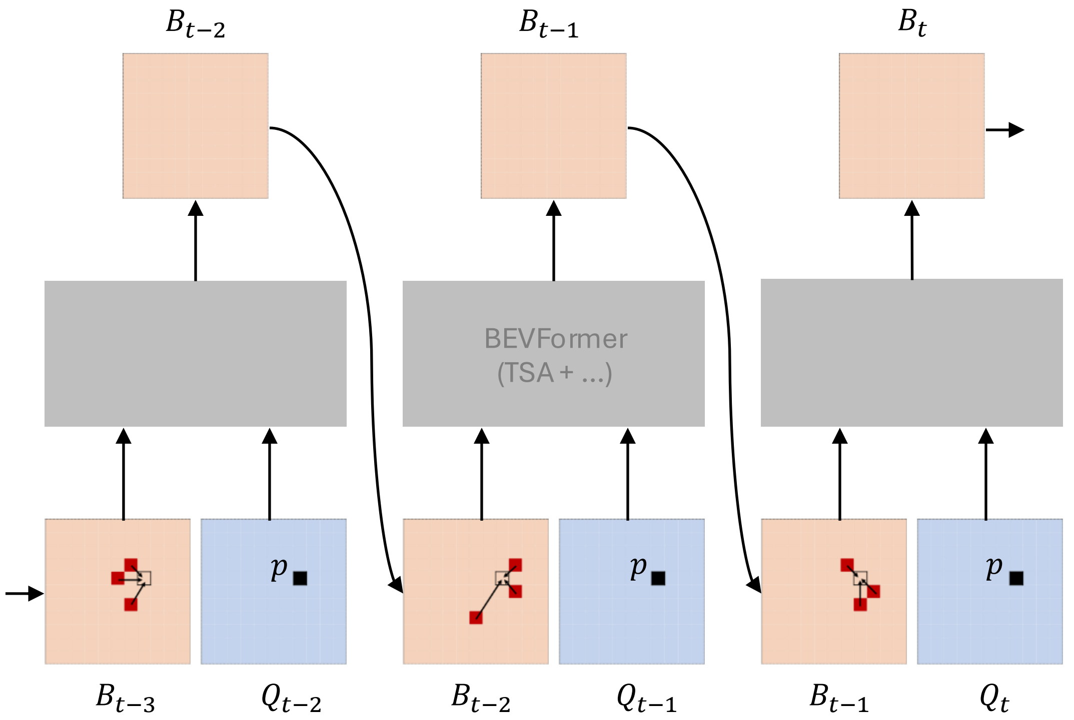
BEV Queries / Features



Methodology - BEVFormer



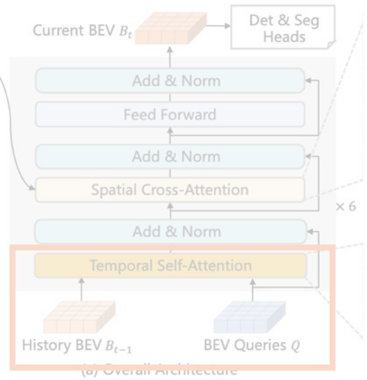
- Temporal Self-Attention (TSA)
  - RNN 과 Self-Attention 메커니즘 컨셉을 적용
  - RNN
    이전 타임스탬프의 BEV Feature (B_{t-1})의 정보를 현재 타임스탬프의 BEV Queries (Q)에 반영
  - Self-Attention
    BEV Query (Q)에서 각각의 위치 Q_p 마다, 관련이 높을 것으로 예상되는 주변 위치의 BEV Feature (B_{t-1}) 정보를 B_t에 반영



NOTE! Training과 Inference의 효율을 높이기 위해, ***Deformable DETR(Zhu et al., 2020) [1]*** 의 Attention 메커니즘을 활용.

[1] Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



- Temporal Self-Attention (TSA) ← Deformable Attention

B_{t-1}

Ego Vehicle의 움직임 정보를 바탕으로
t-1의 BEV Feature 들의 위치 조정*

$$\mathrm{TSA}(Q_p, \{Q, B'_{t-1}\}) = \sum_{V \in \{Q, B'_{t-1}\}} \mathrm{DeformAttn}(Q_p, p, V),$$

$$\mathrm{DeformAttn}(q, p, x) = \sum_{i=1}^{N_{\mathrm{head}}} \mathcal{W}_i \sum_{j=1}^{N_{\mathrm{key}}} \mathcal{A}_{ij} \cdot \mathcal{W}'_i x(p + \Delta p_{ij}),$$

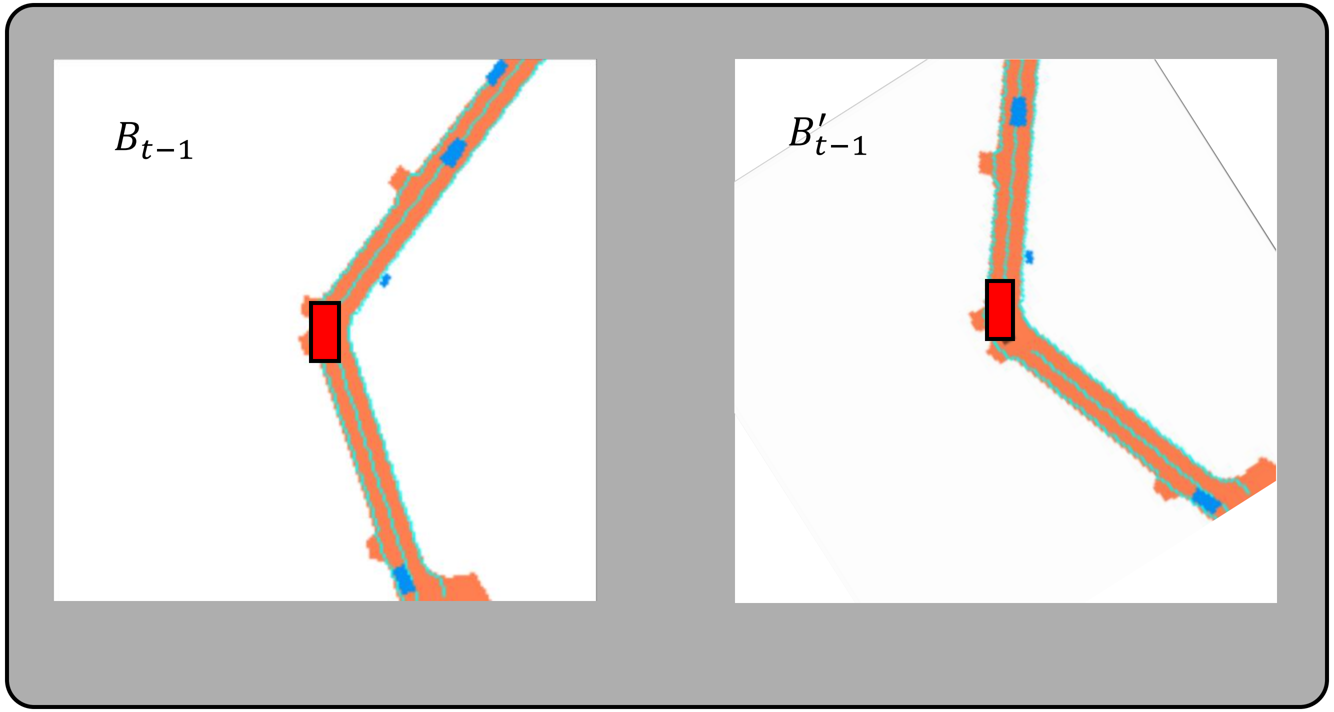
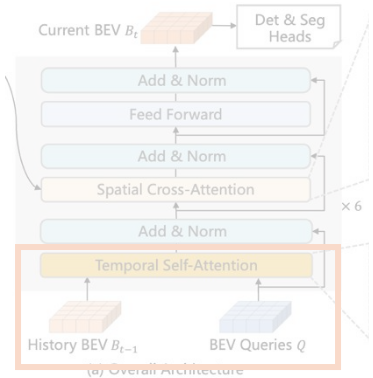
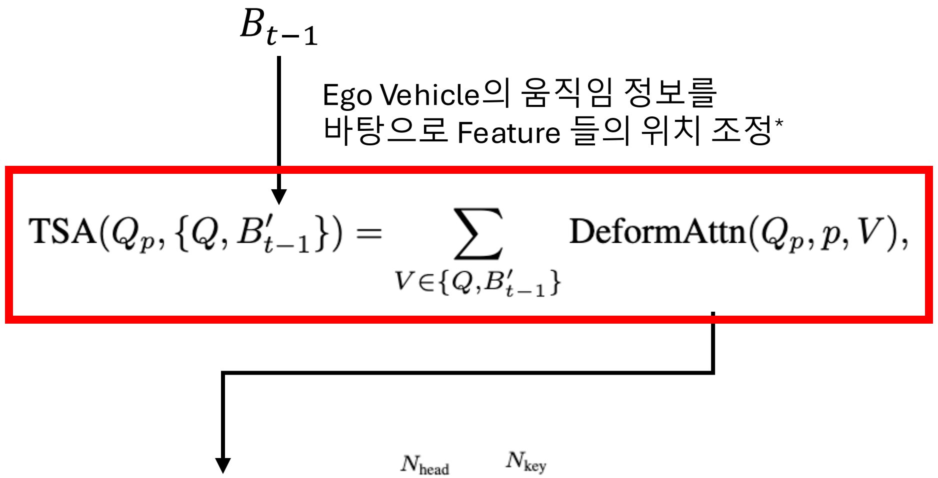


Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



- Temporal Self-Attention (TSA) ← Deformable Attention



$$\text{TSA}(Q_p, \{Q, B'_{t-1}\}) = \sum_{V \in \{Q, B'_{t-1}\}} \text{DeformAttn}(Q_p, p, V),$$

$$\text{DeformAttn}(q, p, x) = \sum_{i=1}^{N_{\text{head}}} \mathcal{W}_i \sum_{j=1}^{N_{\text{key}}} \mathcal{A}_{ij} \cdot \mathcal{W}'_i x(p + \Delta p_{ij}),$$

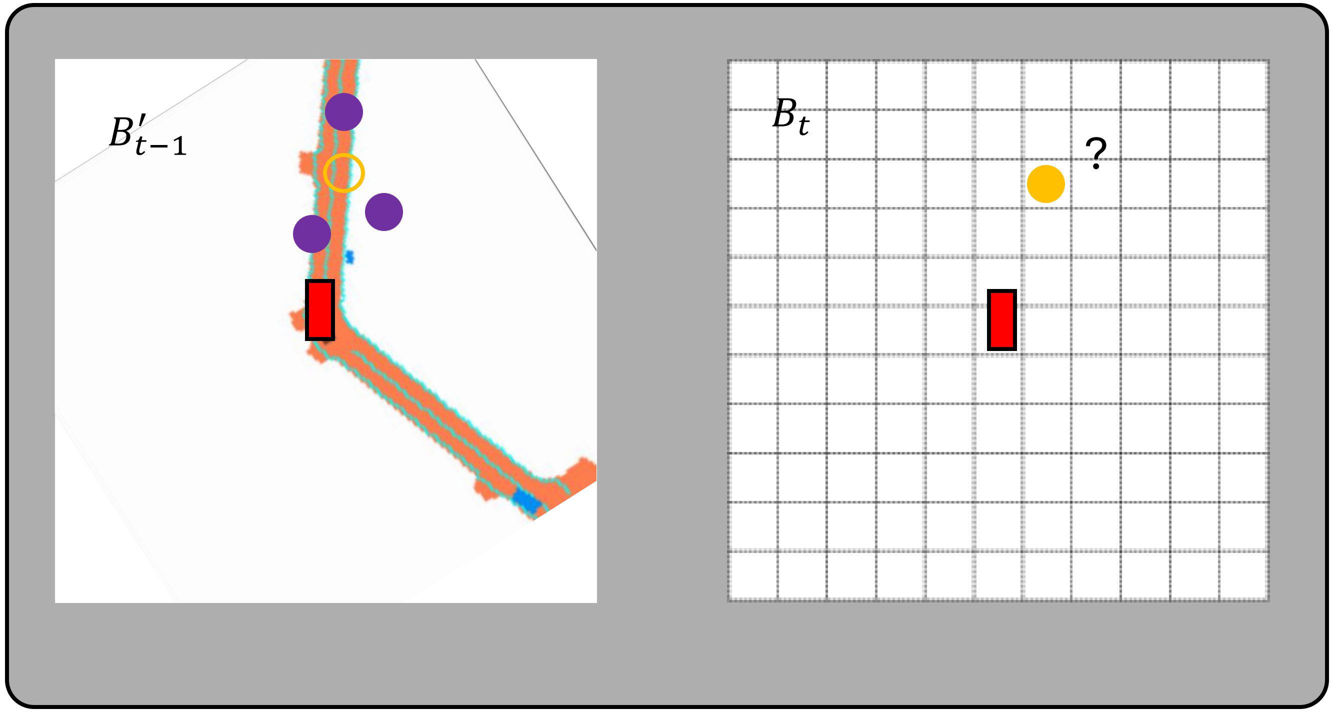
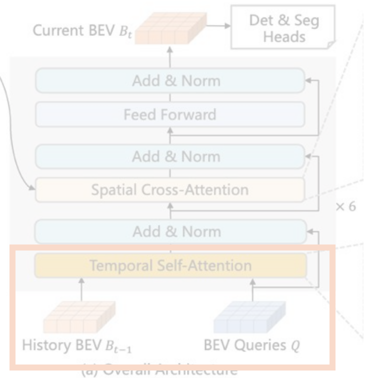


Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



- Temporal Self-Attention (TSA) ← Deformable Attention

B_{t-1}

Ego Vehicle의 움직임 정보를
바탕으로 Feature 들의 위치 조정*

$$\boxed{\mathrm{TSA}(Q_p, \{Q, B'_{t-1}\}) = \sum_{V \in \{Q, B'_{t-1}\}} \mathrm{DeformAttn}(Q_p, p, V),}$$

$$\mathrm{DeformAttn}(q, p, x) = \sum_{i=1}^{N_{\mathrm{head}}} \mathcal{W}_i \sum_{j=1}^{N_{\mathrm{key}}} \mathcal{A}_{ij} \cdot \mathcal{W}'_i x(p + \Delta p_{ij}),$$

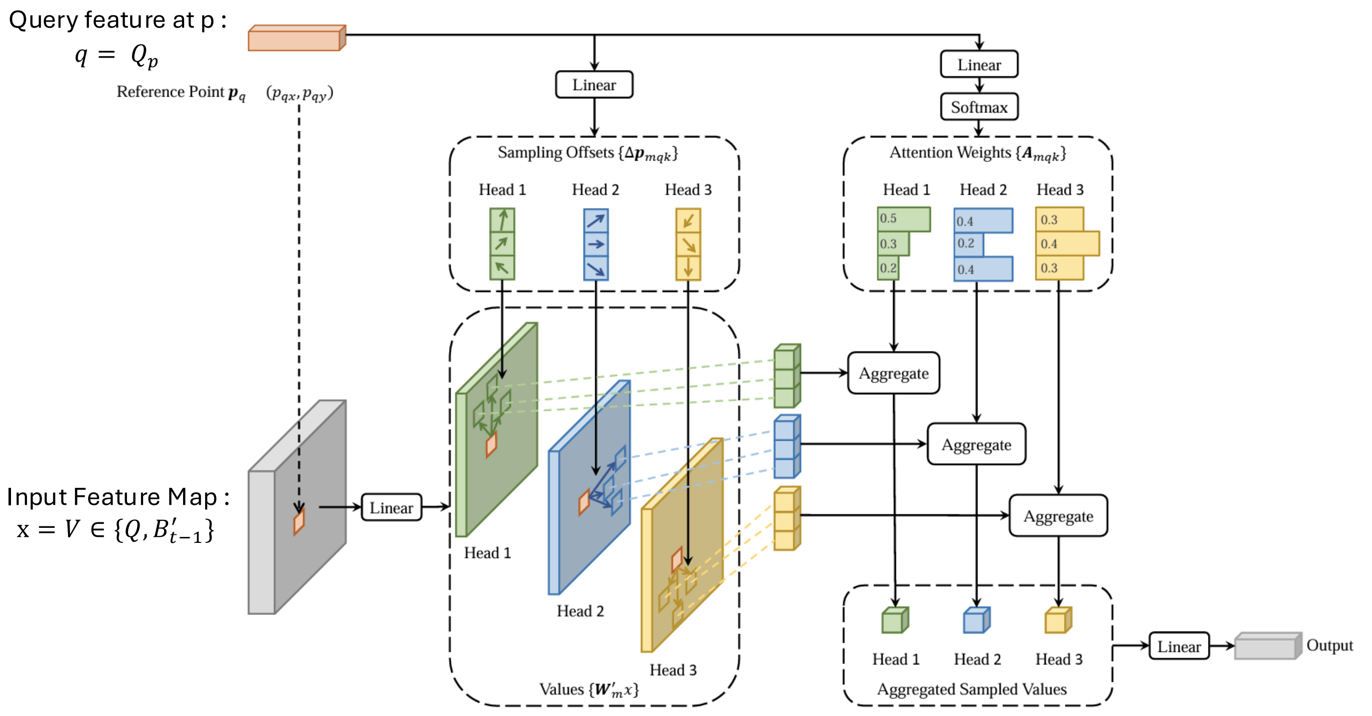
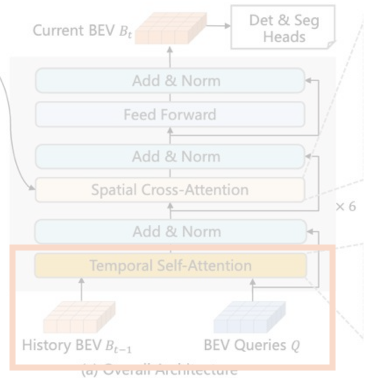


Figure 2: Illustration of the proposed deformable attention module.

Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).



Methodology - BEVFormer

- Temporal Self-Attention (TSA) ← Deformable Attention

B_{t-1}

Ego Vehicle의 움직임 정보를 바탕으로 Feature 들의 위치 조정*

$$\mathrm{TSA}(Q_p, \{Q, B'_{t-1}\}) = \sum_{V \in \{Q, B'_{t-1}\}} \mathrm{DeformAttn}(Q_p, p, V),$$

$$\mathrm{DeformAttn}(q, p, x) = \sum_{i=1}^{N_{\mathrm{head}}} \mathcal{W}_i \sum_{j=1}^{N_{\mathrm{key}}} \mathcal{A}_{ij} \cdot \boxed{\mathcal{W}'_i x(p + \Delta p_{ij}),}$$

논문에선 $N_{head} = 8, N_{key} = 4$ 사용

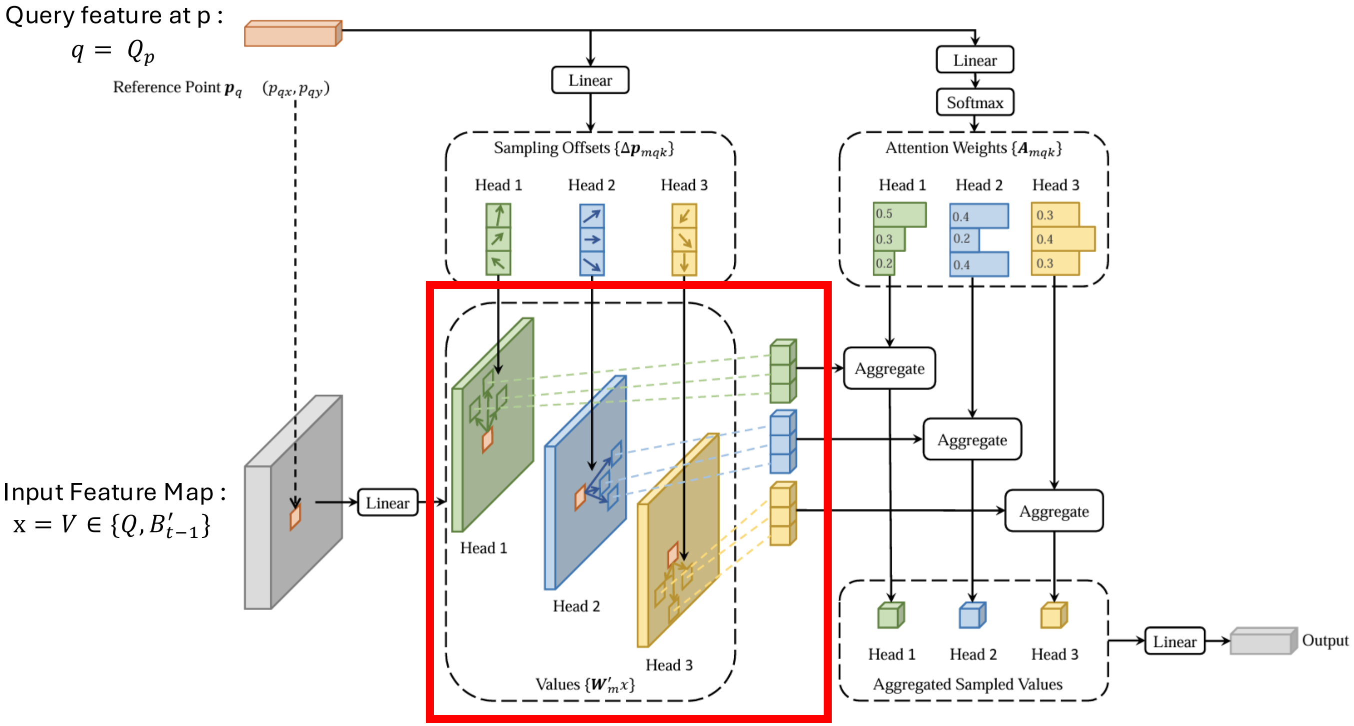
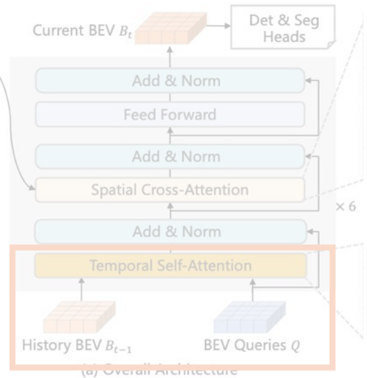


Figure 2: Illustration of the proposed deformable attention module.

Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



- Temporal Self-Attention (TSA) ← Deformable Attention

B_{t-1}

Ego Vehicle의 움직임 정보를
바탕으로 Feature 들의 위치 조정*

$$\mathrm{TSA}(Q_p, \{Q, B'_{t-1}\}) = \sum_{V \in \{Q, B'_{t-1}\}} \mathrm{DeformAttn}(Q_p, p, V),$$

$$\mathrm{DeformAttn}(q, p, x) = \sum_{i=1}^{N_{\text{head}}} \mathcal{W}_i \boxed{\sum_{j=1}^{N_{\text{key}}} \mathcal{A}_{ij} \cdot \mathcal{W}'_i x(p + \Delta p_{ij}),}$$

논문에선 $N_{head} = 8, N_{key} = 4$ 사용

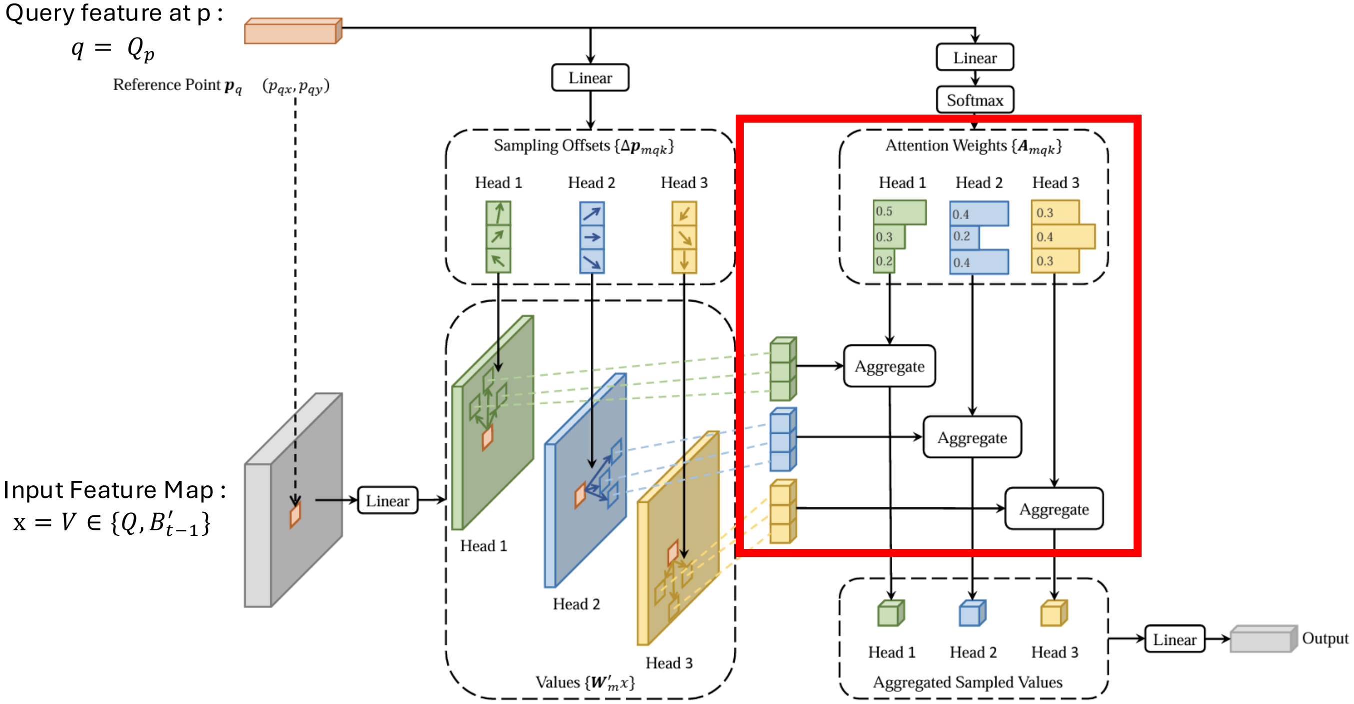
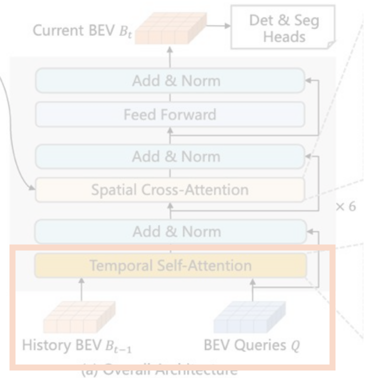


Figure 2: Illustration of the proposed deformable attention module.

Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



- Temporal Self-Attention (TSA) ← Deformable Attention

B_{t-1}

Ego Vehicle의 움직임 정보를
바탕으로 Feature 들의 위치 조정*

$$\mathrm{TSA}(Q_p, \{Q, B'_{t-1}\}) = \sum_{V \in \{Q, B'_{t-1}\}} \mathrm{DeformAttn}(Q_p, p, V),$$

$$\mathrm{DeformAttn}(q, p, x) = \boxed{\sum_{i=1}^{N_{\mathrm{head}}} \mathcal{W}_i \sum_{j=1}^{N_{\mathrm{key}}} \mathcal{A}_{ij} \cdot \mathcal{W}'_i x(p + \Delta p_{ij}),}$$

논문에선 $N_{head} = 8, N_{key} = 4$ 사용

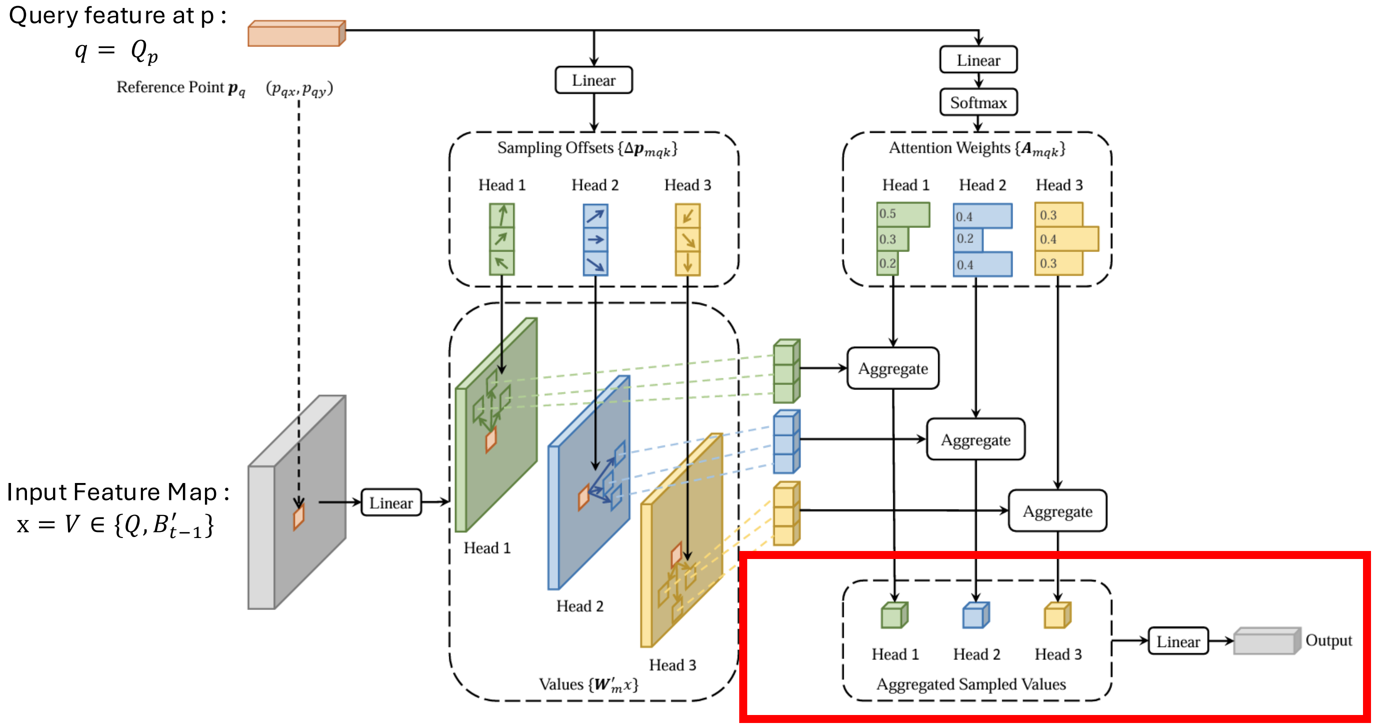
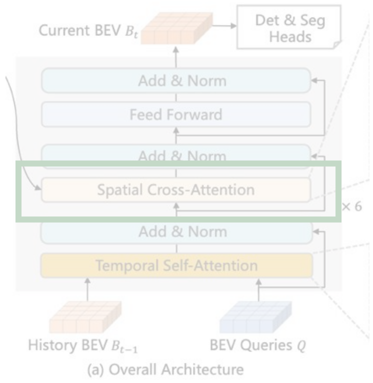


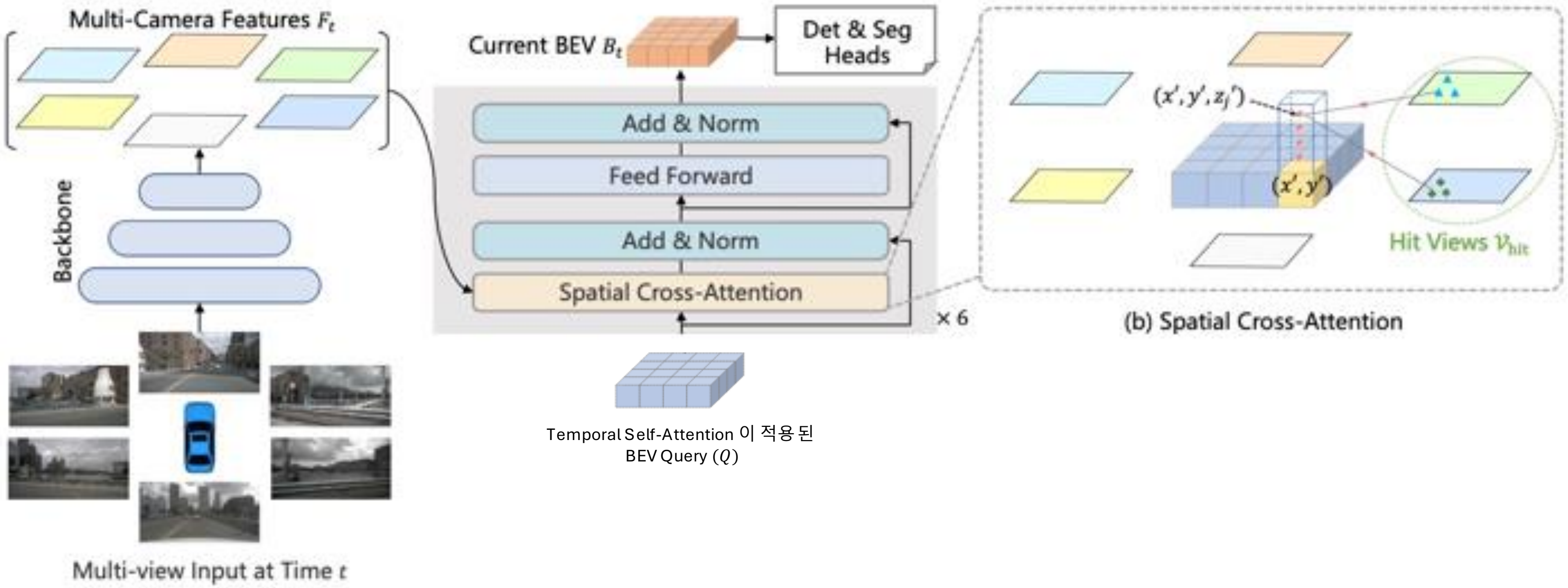
Figure 2: Illustration of the proposed deformable attention module.

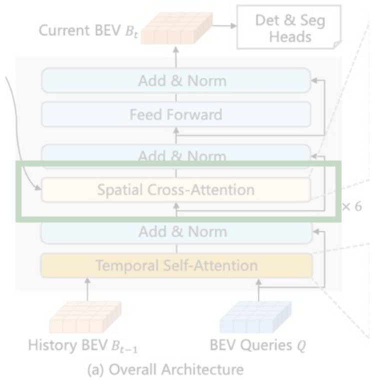
Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



- Spatial Cross-Attention (SCA)

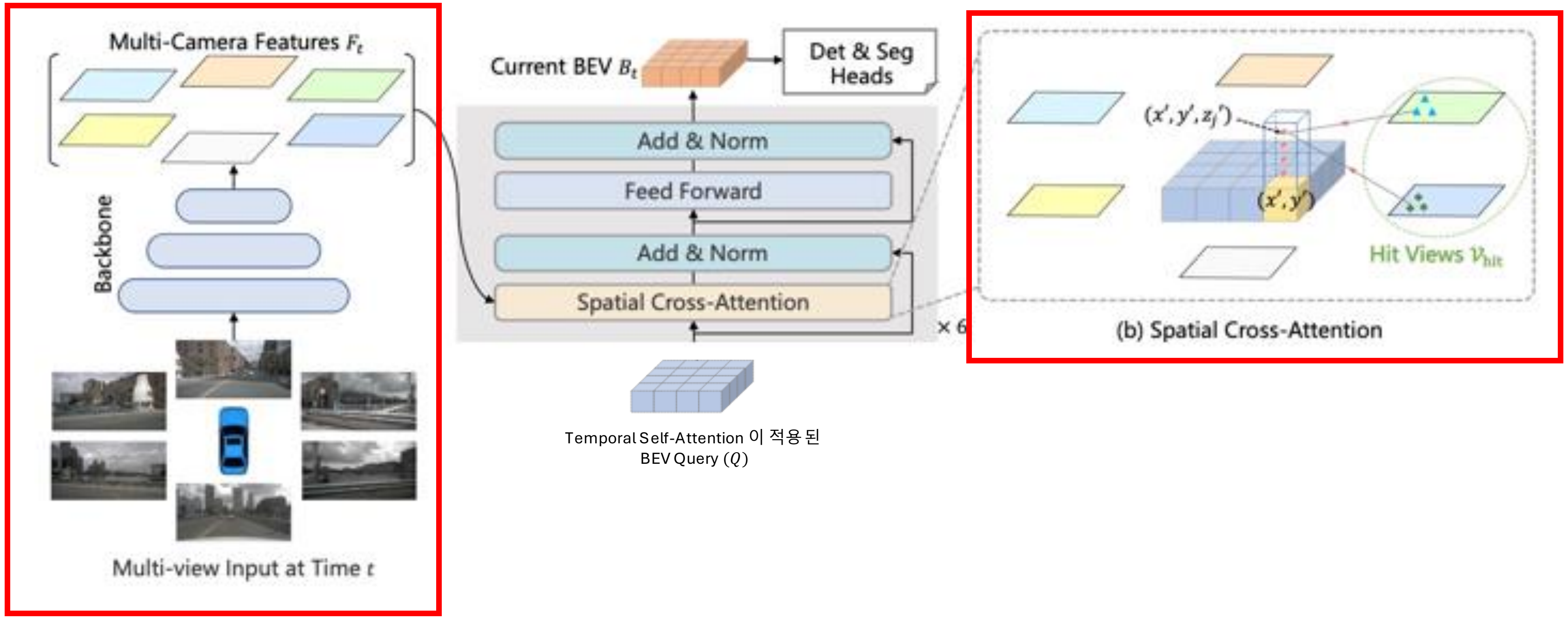




Methodology - BEVFormer

- Spatial Cross-Attention (SCA)

1. Hit Views $\mathcal{V}_{hit}$ 선정



2. BEV Query와 Multi-Camera Feature 간의 Cross-Attention
(Multi-Scale Deformable Attention)

Methodology - BEVFormer

- Spatial Cross-Attention (SCA)

    1. **Hit Views $\mathcal{V}_{hit}$ 선정**

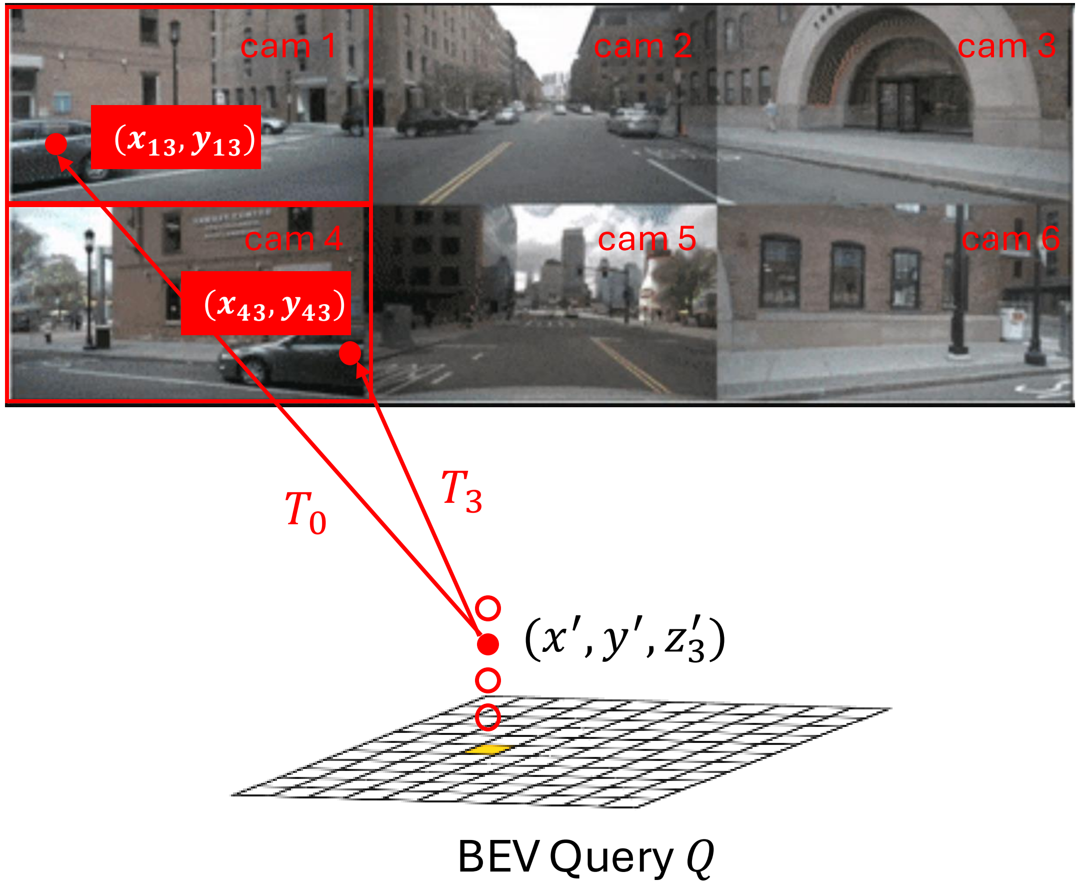
       BEV Query Q 의 각 Point 에 대응되는 Camera Image 의 Feature 만을 사용하기 위함

       a. Grid 내 Point (x, y) 의 실제 거리 (x', y')

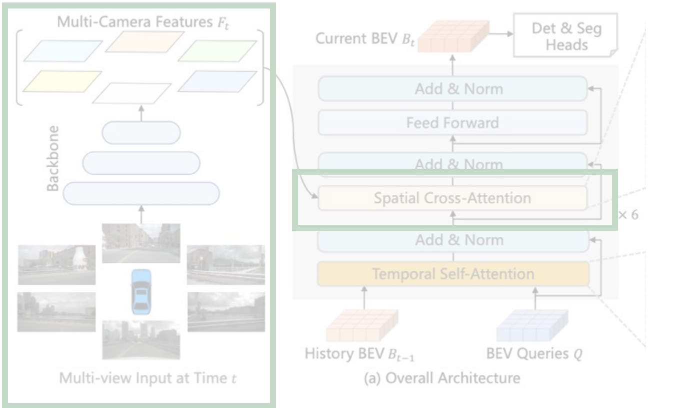
       b. 높이 정보를 고려하기 위해 $\{z_j'\}_{j=1}^{N_{ref}}$ 를 추가

       c. Camera의 Extrinsic & Intrinsic 정보를 바탕으로 계산된 **Projection Matrix T_i** 를 사용하여 각 Camera의 2D 이미지에 투영

$$\mathcal{P}(p, i, j) = (x_{ij}, y_{ij})$$
$$\text{where } z_{ij} \cdot \begin{bmatrix} x_{ij} & y_{ij} & 1 \end{bmatrix}^T = T_i \cdot \begin{bmatrix} x' & y' & z_j' & 1 \end{bmatrix}^T.$$



Methodology - BEVFormer



- Spatial Cross-Attention (SCA)
  2. **BEV Query와 Multi-Camera Feature 간의 Cross-Attention**

$$\mathrm{SCA}(Q_p, F_t) = \frac{1}{|\mathcal{V}_{hit}|} \sum_{i \in \mathcal{V}_{hit}} \sum_{j=1}^{N_{ref}} \mathrm{MSDeformAttn}(Q_p, \mathcal{P}(p,i,j), F_t^i)$$

$$\mathrm{MSDeformAttn}(q, \hat{\mathcal{P}}, \{x^l\}_{l=1}^{L}) = \sum_{i=1}^{N_{head}} \mathcal{W}_i [\sum_{j=1}^{N_{key}} \sum_{l=1}^{L} \mathcal{A}_{ijl} \cdot \mathcal{W}_i' x^l (\phi_l(\hat{\mathcal{P}}) + \Delta p_{ijl})]$$

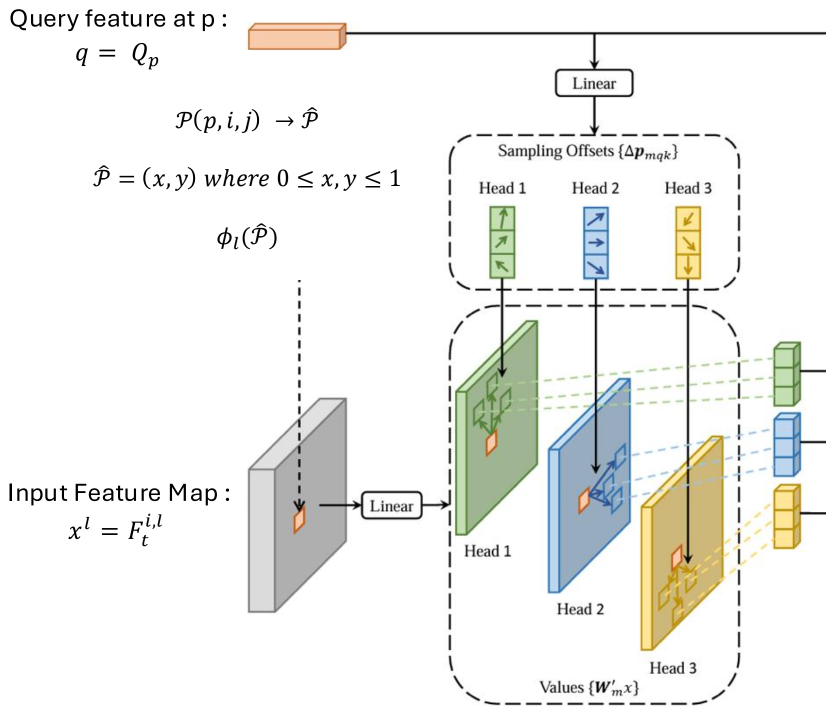
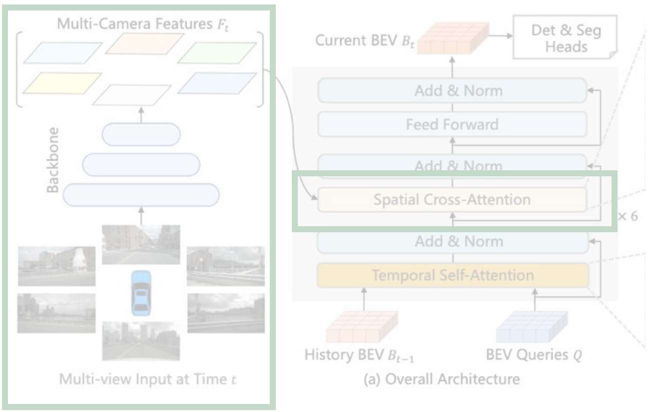


Figure 2: Illustration of the proposed def

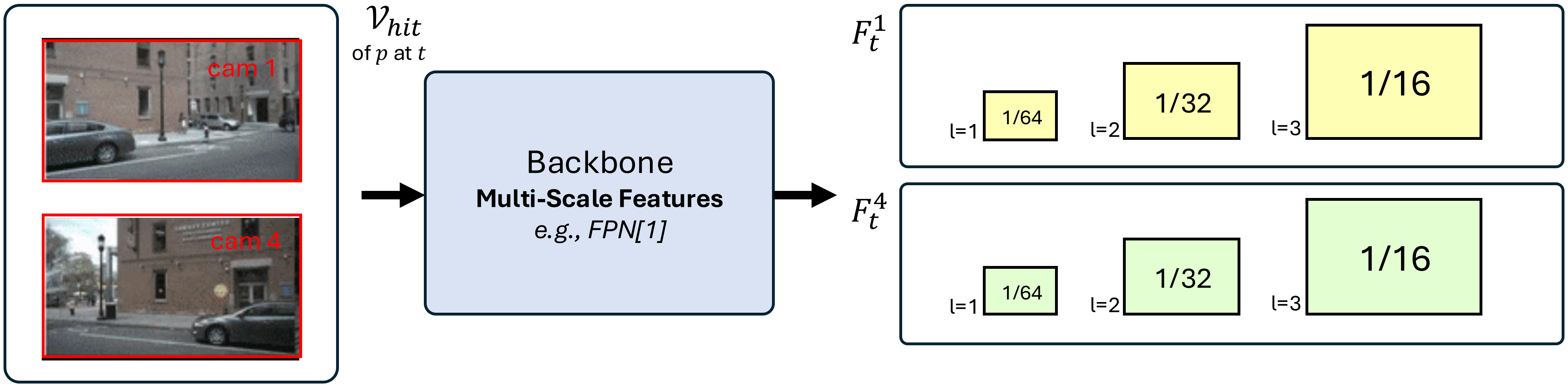
Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



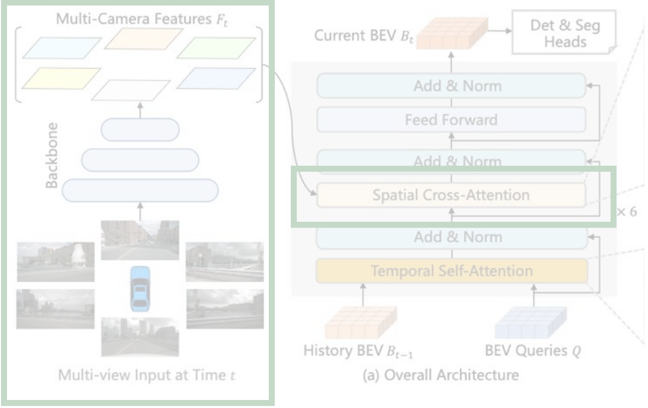
- Spatial Cross-Attention (SCA)
    2. **BEV Query와 Multi-Camera Feature 간의 Cross-Attention**

$$\mathrm{SCA}(Q_p, F_t) = \boxed{\frac{1}{|\mathcal{V}_{hit}|}\sum_{i\in\mathcal{V}_{hit}}} \sum_{j=1}^{N_{ref}} \mathrm{MSDeformAttn}(Q_p, \mathcal{P}(p,i,j), F_t^i)$$



[1] Lin, Tsung-Yi, et al. "Feature pyramid networks for object detection." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

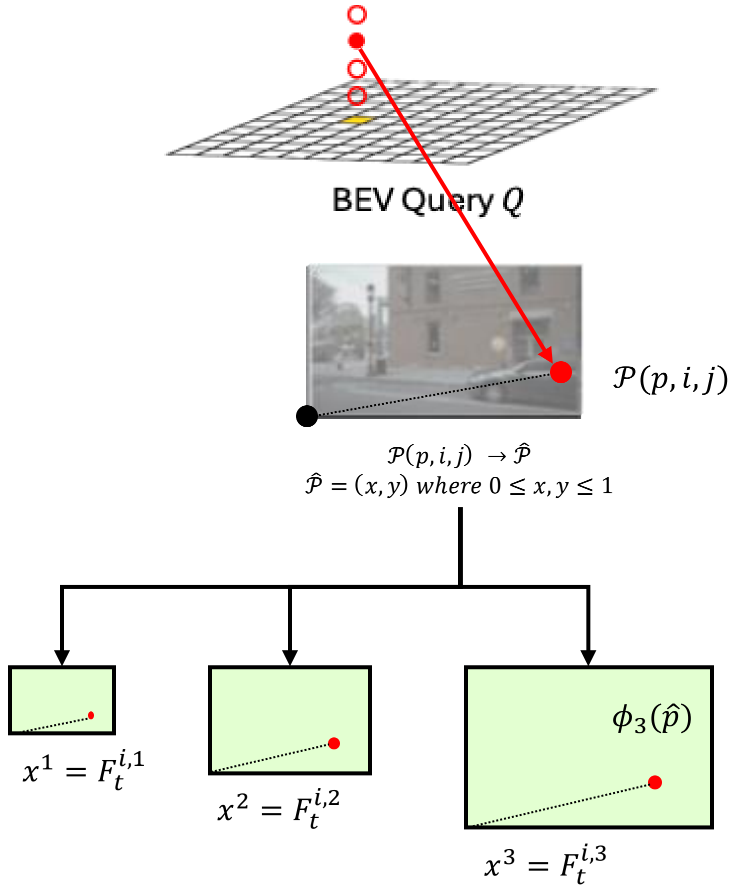
Methodology - BEVFormer



- Spatial Cross-Attention (SCA)
    2. **BEV Query와 Multi-Camera Feature 간의 Cross-Attention**

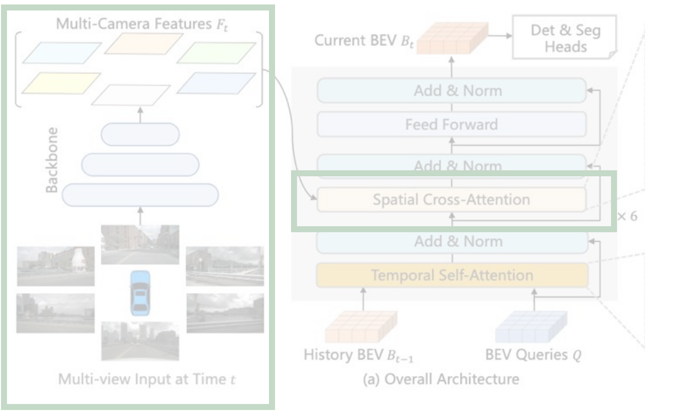
$$\mathrm{SCA}(Q_p, F_t) = \frac{1}{|\mathcal{V}_{hit}|}\sum_{i\in\mathcal{V}_{hit}}\sum_{j=1}^{N_{ref}} \mathrm{MSDeformAttn}(Q_p, \boxed{\mathcal{P}(p,i,j)}, F_t^i)$$

$$\mathrm{MSDeformAttn}(q, \boxed{\hat{\mathcal{P}}, \{x^l\}_{l=1}^{L}}) = \sum_{i=1}^{N_{head}} \mathcal{W}_i\Big[\sum_{j=1}^{N_{key}} \boxed{\sum_{l=1}^{L}} \mathcal{A}_{ijl} \cdot \mathcal{W}'_i x^l(\boxed{\phi_l(\hat{\mathcal{P}})} + \Delta p_{ijl})\Big]$$



Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer



- Spatial Cross-Attention (SCA)
  1. **Hit Views $\mathcal{V}_{hit}$ 선정**
  2. **BEV Query와 Multi-Camera Feature 간의 Cross-Attention**

$$\mathrm{SCA}(Q_p, F_t) = \frac{1}{|\mathcal{V}_{hit}|} \sum_{i \in \mathcal{V}_{hit}} \sum_{j=1}^{N_{ref}} \mathrm{MSDeformAttn}(Q_p, \mathcal{P}(p,i,j), F_t^i)$$

$$\mathrm{MSDeformAttn}(q, \hat{\mathcal{P}}, \{x^l\}_{l=1}^{L}) = \sum_{i=1}^{N_{head}} \mathcal{W}_i [\sum_{j=1}^{N_{key}} \sum_{l=1}^{L} \mathcal{A}_{ijl} \cdot \mathcal{W}_i' x^l (\phi_l(\hat{\mathcal{P}}) + \Delta p_{ijl})]$$

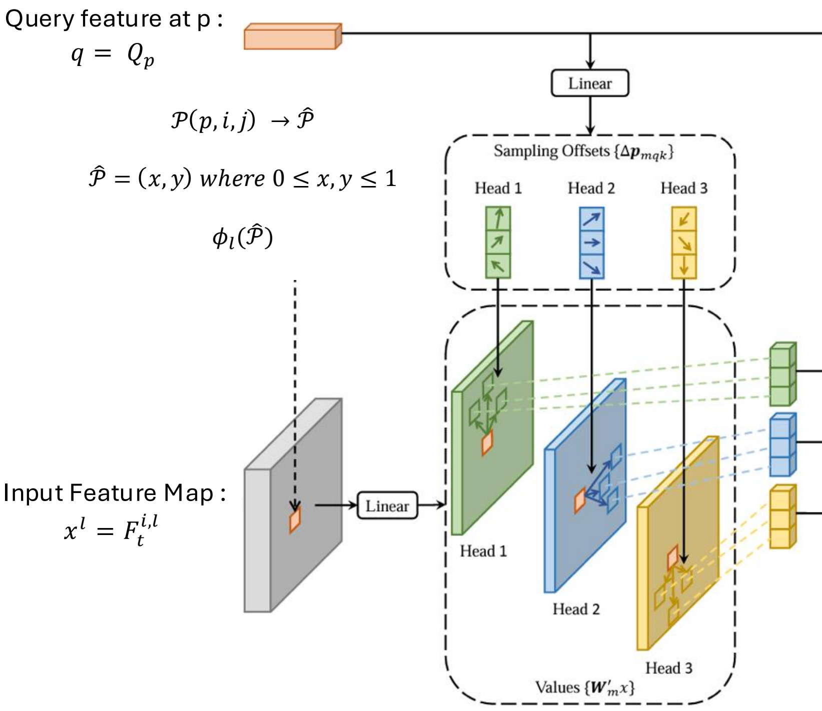
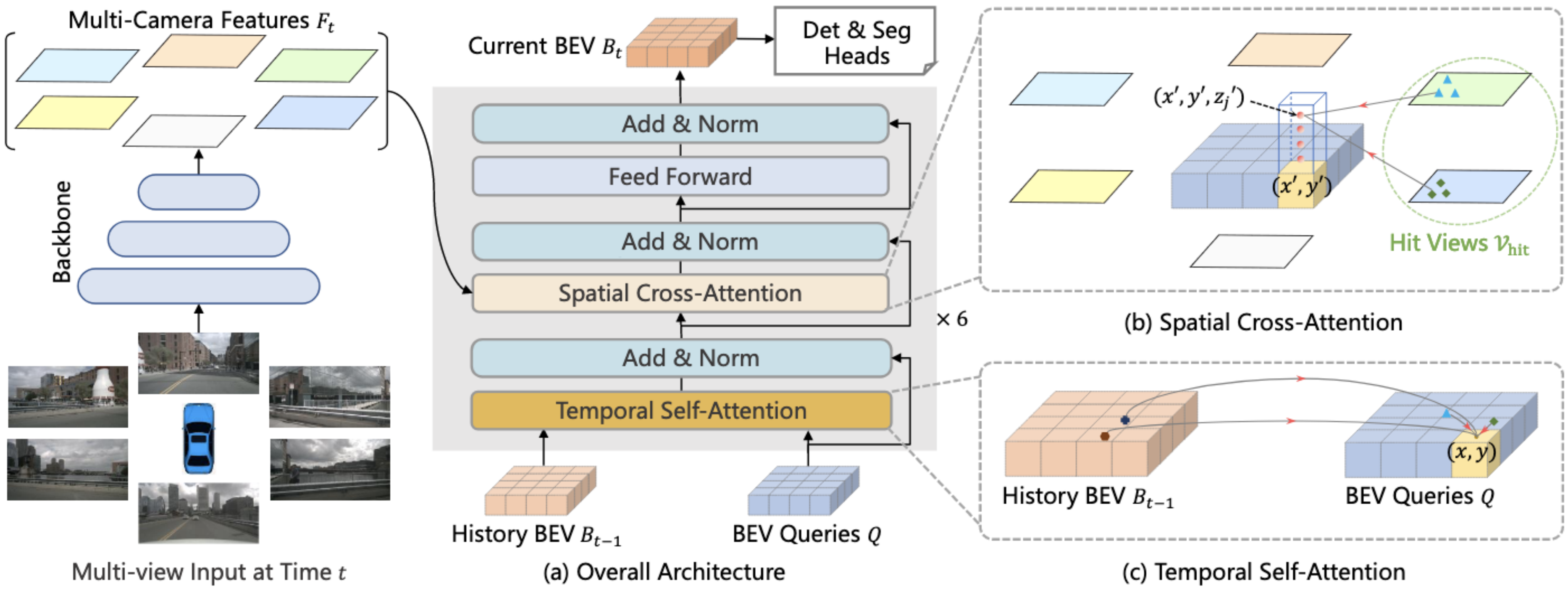


Figure 2: Illustration of the proposed def

Figure : Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." *arXiv preprint arXiv:2010.04159* (2020).

Methodology - BEVFormer

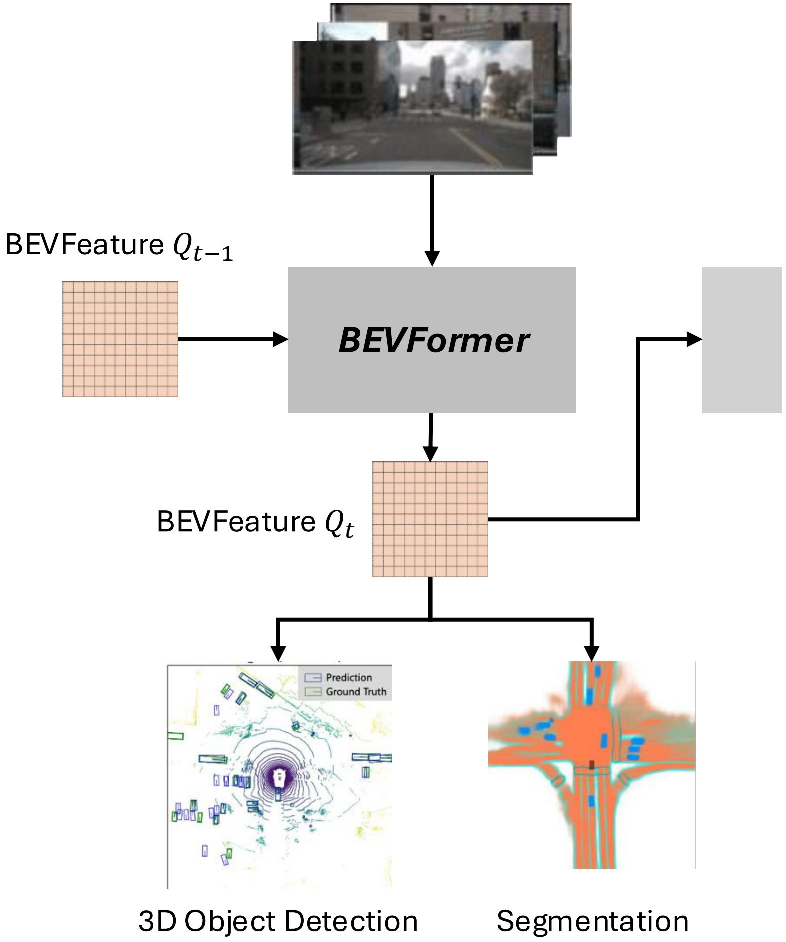
- Recap



Experiments

Experiment Setup

- Training
  - Dataset
    - NuScenes Dataset [1] : 3D OD + Seg
    - Waymo Open Dataset [2] : 3D OD
  - Backbone
    - ResNet101-DCN (FCOS3D [3])
    - VoVnet-99 (DD3D [4])
  - Head
    - 3D OD : Deformable DETR [5]
    - Panoptic SegFormer [6]
  - Loss Function
    - 3D Object Detection Task : L1 loss
    - Segmentation Task : Generalized IoU loss



[1] Caesar, Holger, et al. "nuscenes: A multimodal dataset for autonomous driving." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.
[2] Sun, Pei, et al. "Scalability in perception for autonomous driving: Waymo open dataset." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.
[3] Wang, Tai, et al. "Fcos3d: Fully convolutional one-stage monocular 3d object detection." Proceedings of the IEEE/CVF international conference on computer vision. 2021.
[4] Park, Dennis, et al. "Is pseudo-lidar needed for monocular 3d object detection?." Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021.
[5] Zhu, Xizhou, et al. "Deformable detr: Deformable transformers for end-to-end object detection." arXiv preprint arXiv:2010.04159 (2020).
[6] Li, Zhiqi, et al. "Panoptic segformer: Delving deeper into panoptic segmentation with transformers." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022.

Experiment Results

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 [55]	L	-	0.569	0.463	-	-	-	-	-
CenterPoint-Voxel [52]	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

1. Camera-only Baseline 모델 (Attention X)에 비해, **9.0 Points 높은 NDS 퍼포먼스**와 현저히 **줄어든 Velocity 예측 에러**를 보여줌.
2. Lidar 센서를 사용한 Baseline 모델에 준하는 NDS 퍼포먼스를 보여줌.

Experiment Results

Table 4: **3D detection and map segmentation results on nuScenes `val` set.** 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

* BEVFormer-S
Spatial Cross-Attention만
적용된 BEVFormer

1. 2D → 3D 기반 Baseline 모델인 **VPN과 LSS 보다 전반적으로 향상된 퍼포먼스**를 보여줌.
2. BEVFormer 모델의 **Multi-Task Learning 적용 가능성을 확인**. (예외, Road and Lane Segmentation)

Experiment Results

1. BEVFormer는 Object **Visibility 가 낮은 (0-40 %) 환경에서도 Robust한 퍼포먼스**를 보여줌.

2. 특히, Visibility가 낮은 환경에서, **Temporal information이 3D Object의 Translation, Orientation, 그리고 Velocity 예측 에러를 줄이는 데 효과적**임을 보여줌.

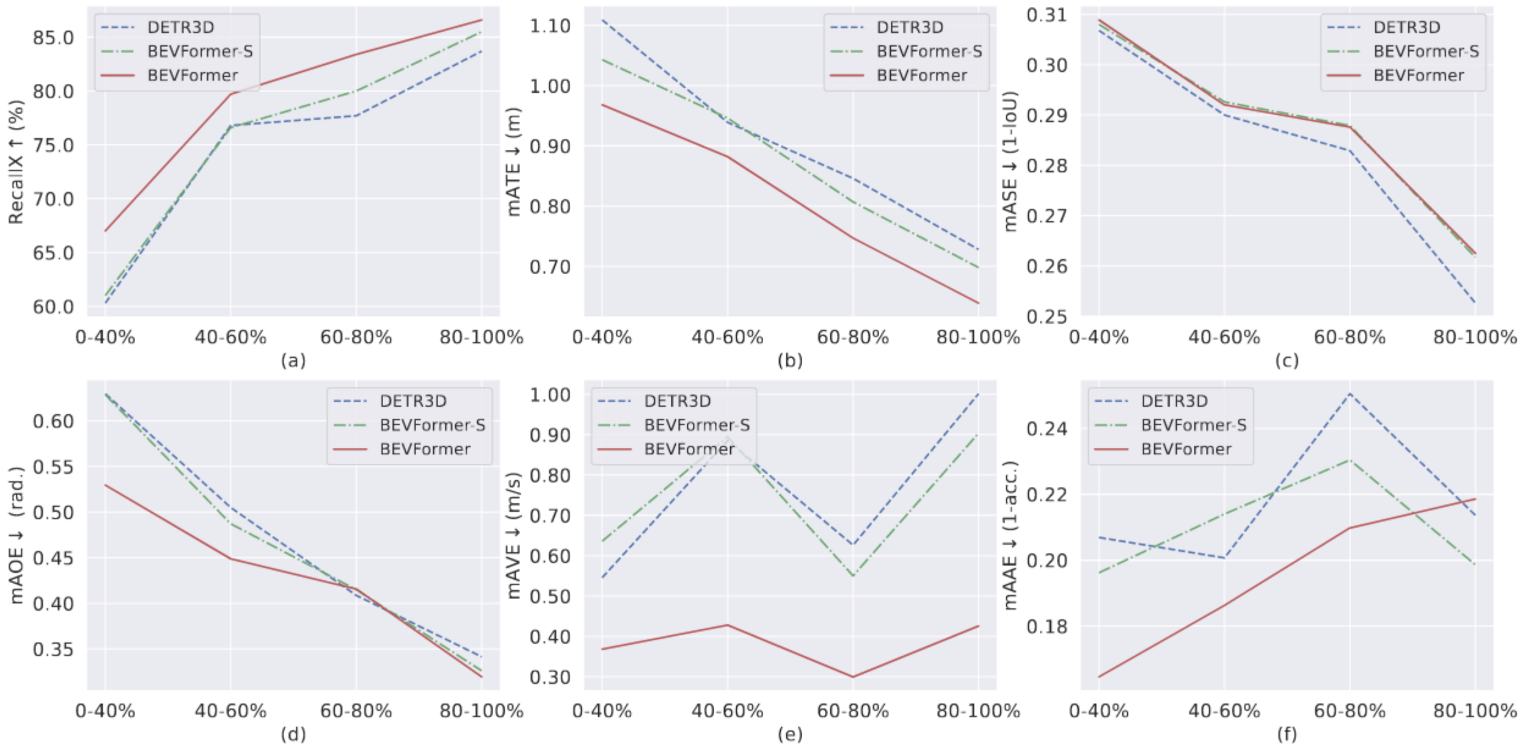
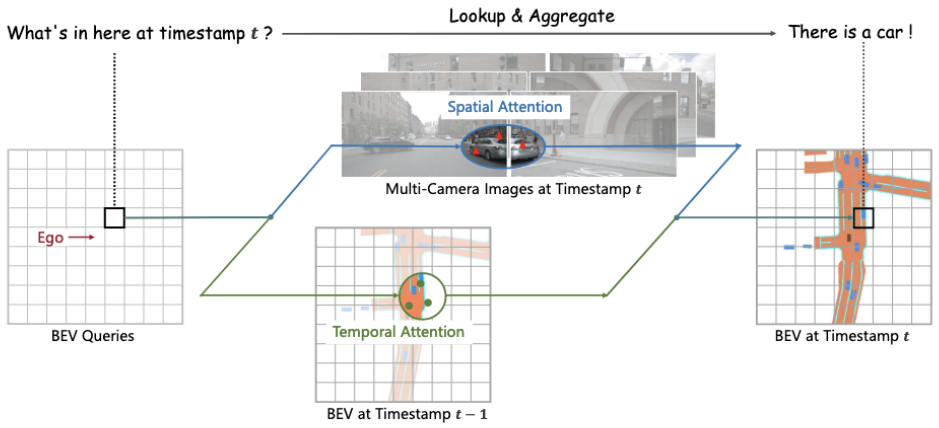


Figure 3: **The detection results of subsets with different visibilities.** We divide the nuScenes `val` set into four subsets based on the visibility that {0-40%, 40-60%, 60-80%, 80-100%} of objects can be visible. (a): Enhanced by the temporal information, BEVFormer has a higher recall on all subsets, especially on the subset with the lowest visibility (0-40%). (b), (d) and (e): Temporal information benefits translation, orientation, and velocity accuracy. (c) and (f): The scale and attribute error gaps among different methods are minimal. Temporal information does not work to benefit an object's scale prediction.

Key takeaways

Key takeaways

- **BEVFormer**
  - *BEV map의 각 위치에서의 Feature 정보를* ***Spatiotemporal Attention Mechanism****을 활용하여 Multi-camera image로부터 효율적으로 추출*
  - *Depth Estimation의 오차로부터 자유로운* ***Top-Down (3D → 2D) 방식의 방법론****.*
  - *Camera만 사용하더라도* ***Lidar 사용 모델과 대등한 수준의 퍼포먼스****.*



Thanks !