숭실대학교

2024년 자율주행 인공지능 챌린지: 차량용 객체 복합 상태 인식

발표자 : 이정윤

2024.11.19
팀명: VIP
팀원: 이정윤, 길다영, 염정훈

주최 과학기술정보통신부 Ministry of Science and ICT
주관 kakaomobility ETRI 한국전자통신연구원 IITP 정보통신기획평가원
KADIF 자율주행기술개발혁신사업단 자율주행DNA기술포럼

Contents

01. 문제 정의 및 목표

대회 개요

주행환경의 차량/버스의 분류(Agent), 의미론적 위치(Location), 후미등 상태(State)를 인식하는 동시에 Instance Segmentation을 진행

문제 정의

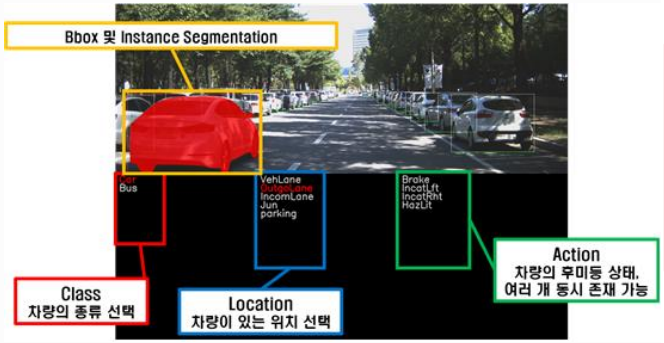
기존 Segmentation 모델로는 차량/버스의 분류(Agent)만 가능

목표

Post-Processing 모듈을 추가하여 의미론적 위치(Location)와 후미등 상태(State)를 인식

다양한 데이터 증강 기법 도입

추론 시 TTA (Test Time Augmentation) 사용



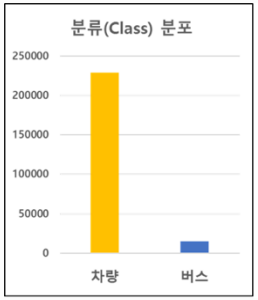
분류 (Agent)	위치 (Location)	상태 (State)
• 차 • 버스	주행차로 진행방향 차로 맞은편 차로 교차로 주차장 (차량/정지 전용)	• 브레이크 등 • 좌/우 깜빡이 • 비상등

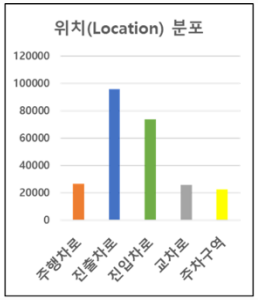
각 객체는 1개의 유일한 분류와 위치를 가지고,
행동의 경우 복수의 행동을 가질 수 있음 (Ex: 정지+Brake)

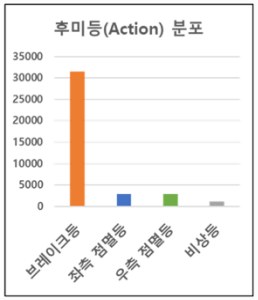
02. 데이터셋 분석

데이터 분석

차량/버스의 분류(Agent), 의미론적 위치(Location)는 Multi-class로, 후미등 상태(State)는 Multi-label로 분류







2개 5개 16(2^4) 개

1. 브레이크등
2. 좌측 점멸등
3. 우측 점멸등
4. 비상등
5. 브레이크 + 좌측 점멸등
6. 브레이크 + 우측 점멸등

...

16. 브레이크 + 좌/우측 점멸등 + 비상등

분류	학습 데이터	평가 데이터
총 이미지 수	33,187	8,909
총 객체 수	243,919	미공개
이미지당 평균 객체 수	7.35	-

총 클래스 개수 : $2 \times 5 \times 16 =$ 160개

차량 / 주행차로 / 브레이크등 + 좌측 점멸등
차량 / 주행차로 / 브레이크등 + 우측 점멸등
...

최종 총 클래스 개수 : 160개 → 70개

제외 클래스

차량 / 주행차로 / 좌측 점멸등 + 우측 점멸등
차량 / 주행차로 / 브레이크등 + 우측 점멸등
...

* multi-class 분류 : 각 샘플이 여러 클래스 중 오직 하나의 클래스에만 속할 수 있는 경우. ex) 개, 고양이, 새 중 하나로 이미지를 분류하는 경우. 각 이미지에는 딱 하나의 레이블만 할당.
* multi-label 분류 : 각 샘플이 여러 클래스에 동시에 속할 수 있는 경우. ex) 하나의 이미지에 개와 고양이가 모두 등장하면, 두 레이블(개, 고양이)을 동시에 할당.

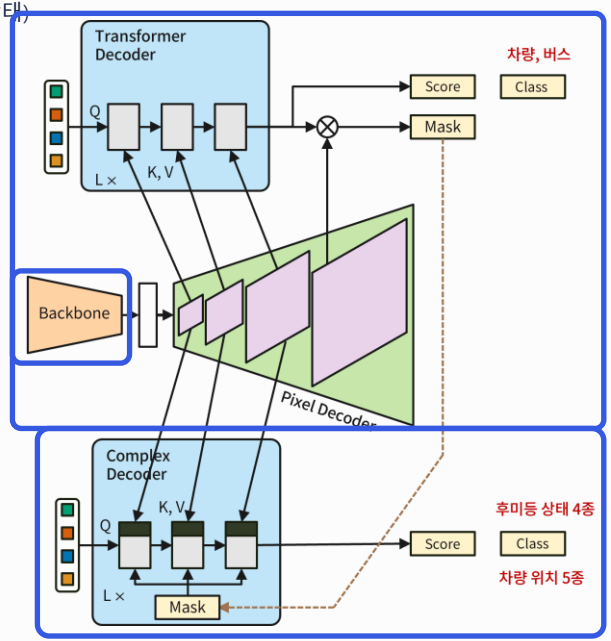
03. 전략 : 모델 설계

모델

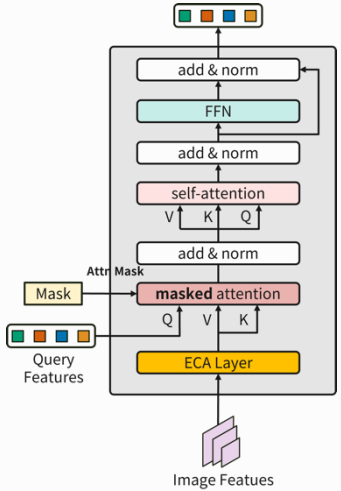
Transformer Decoder (Mask2Former) : 차량/버스의 분류

Complex Decoder : Location Decoder(의미론적 위치) + State Decoder(후미등 상태)

Post-processing



▲ Model Overview



▲ Decoder

Mask2Former(CVPR 2022)

Youtube-VIS 2021과 COCO Dataset로 사전 학습된 CTVIS(ICCV 2023) 가중치를 사용

Backbone

Hiera(ICML 2023) : SAM2에서 사용된 백본

비디오 데이터셋 특성을 고려하여 더 강력한 표현력 제공

ViT-Adapter(ICLR 2023)

Backbone(Hiera)와 Pixel Decoder 사이의 연결 담당

기존 pretrained weights 유지

Complex Decoder (Post-processing)

Location Decoder(의미론적 위치), State Decoder(후미등 상태)

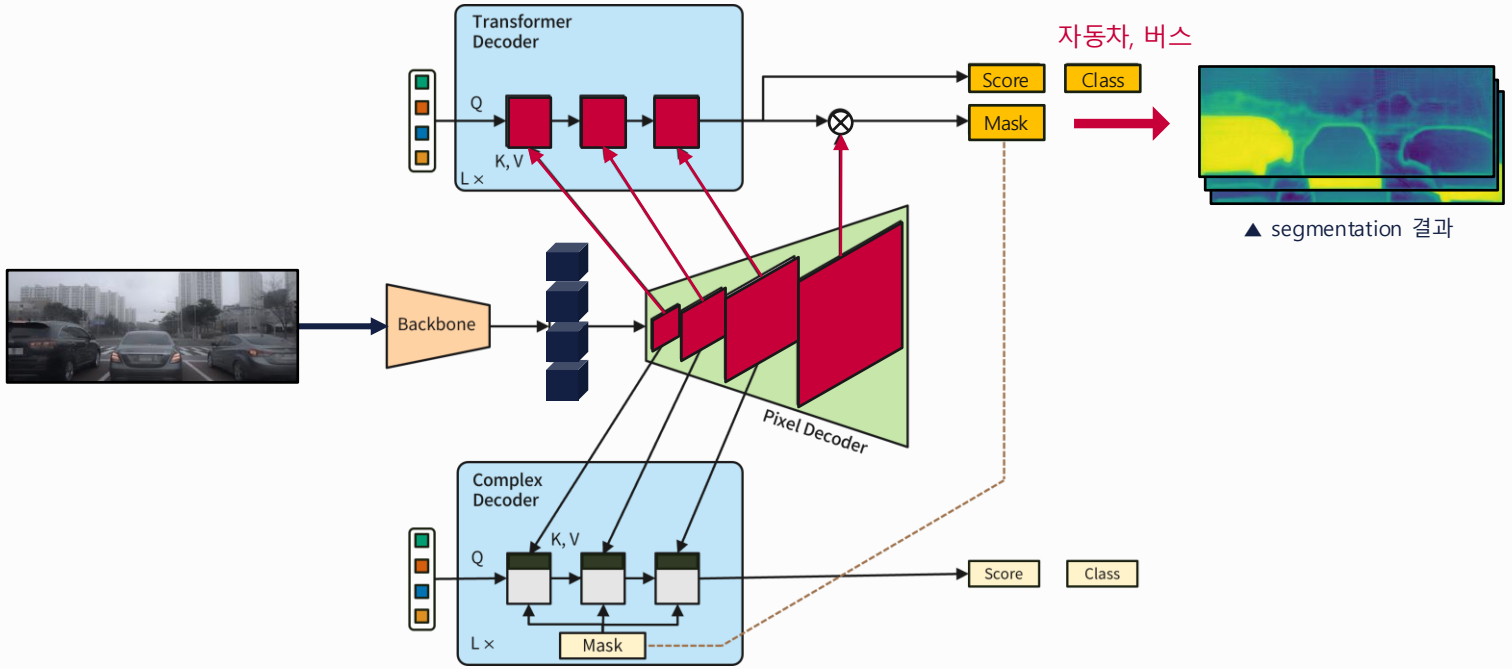
ECA-Net(CVPR 2020) 후미등의 색상 정보에 집중할 수 있도록 채널별 주의 집중

*Ryali, Chaitanya, et al. "Hiera: A hierarchical vision transformer without the bells-and-whistles." International Conference on Machine Learning. PMLR, 2023.
*Ravi, Nikhila, et al. "Sam 2: Segment anything in images and videos." arXiv preprint arXiv:2408.00714 (2024).
*Chen, Zhe, et al. "Vision transformer adapter for dense predictions." arXiv preprint arXiv:2205.08534 (2022).
*Ying, Kaining, et al. "CTVIS: Consistent Training for Online Video Instance Segmentation." arXiv preprint arXiv:2307.12616 (2023).
*Cheng, Bowen, et al. "Masked-attention mask transformer for universal image segmentation." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022.

03. 전략 : 모델 설계

모델 학습 과정

Agent(자동차, 버스) → State(후미등) → Location(위치)

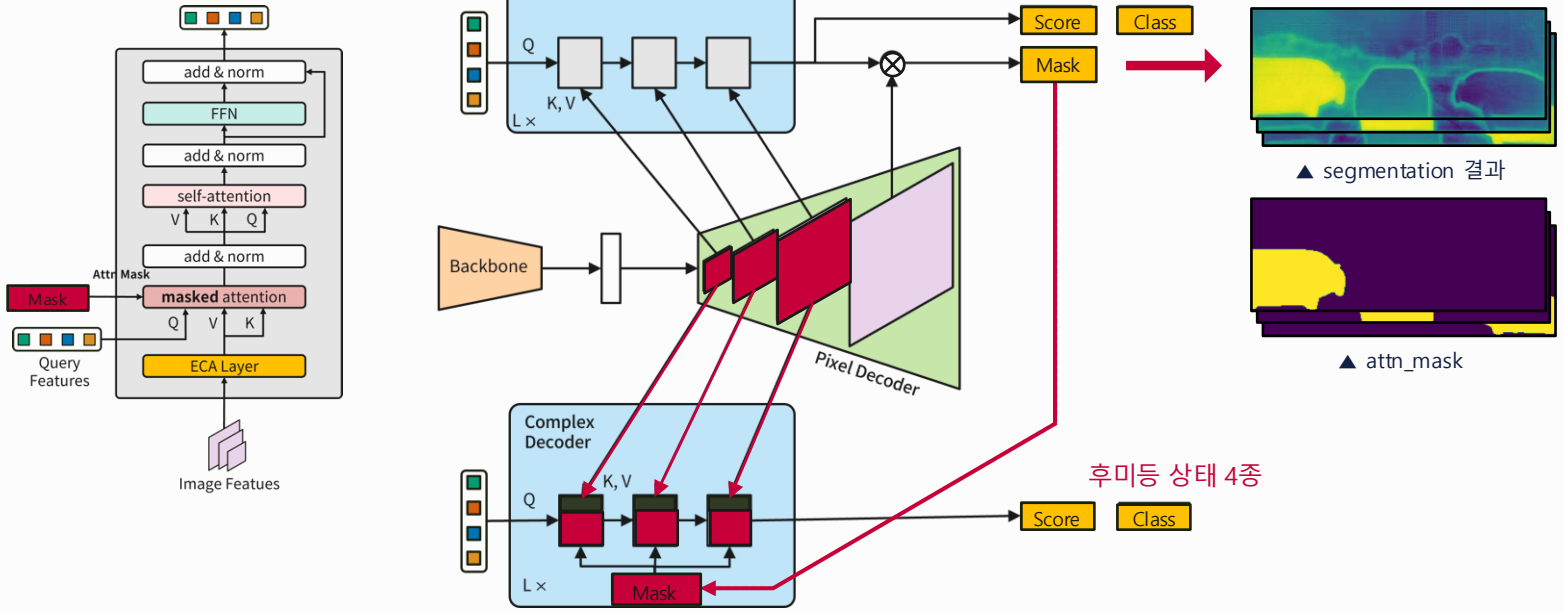


▲ segmentation 결과

03. 전략 : 모델 설계

모델 학습 과정

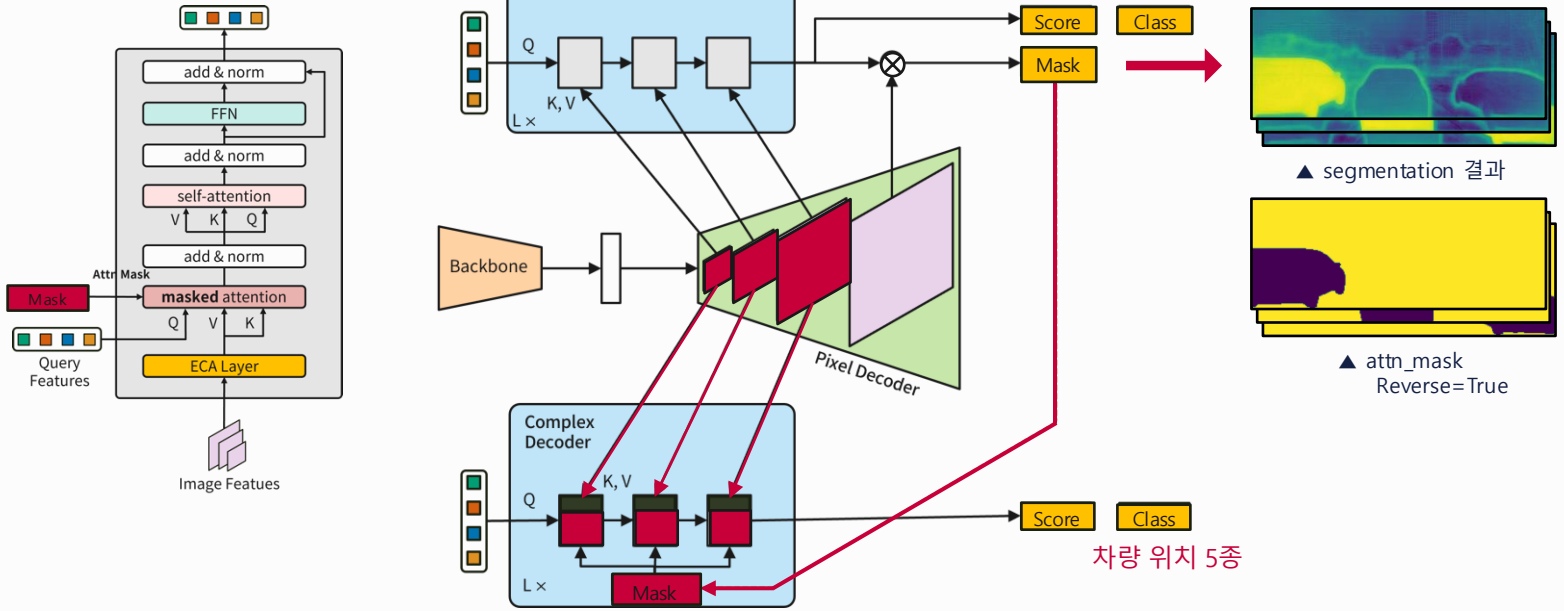
Agent(자동차, 버스) → State(후미등) → Location(위치)



03. 전략 : 모델 설계

모델 학습 과정

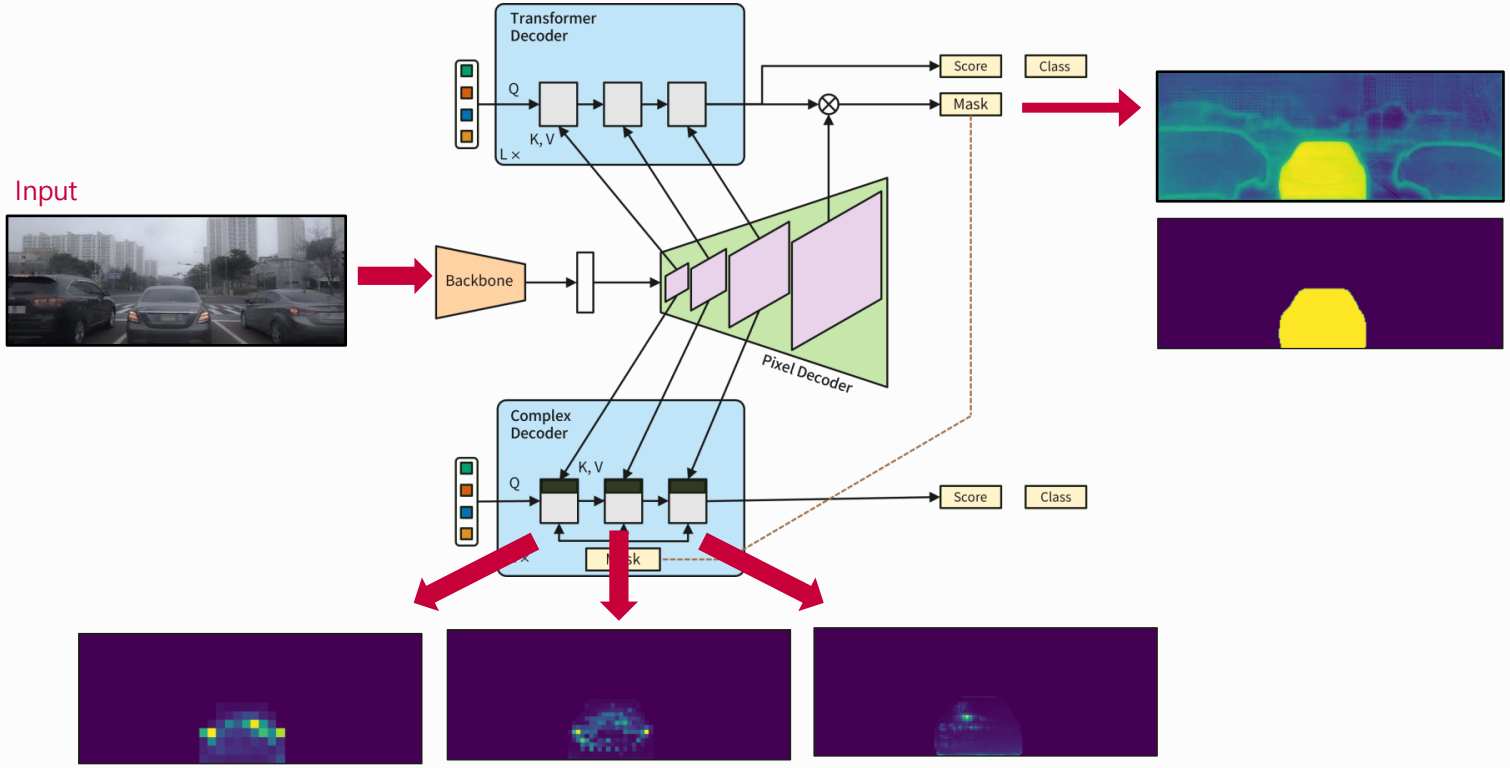
Agent(자동차, 버스) → State(후미등) → Location(위치)



03. 전략 : 모델 설계

모델 학습

Agent(자동차, 버스) → State(후미등) → Location(위치)



03. 전략 : 데이터 증강

데이터 증강 방법

Flip, CoarseDropout, Blur, RandomContrast, RandomBrightness, RandomSaturation

Origin image

Flip

CoarseDropout

Blur

RandomContrast

RandomBrightness

RandomSaturation

Intensity = 0.5

Intensity = 1.5

03. 전략 : TTA (Test Time Augmentation)

TTA (Test Time Augmentation)

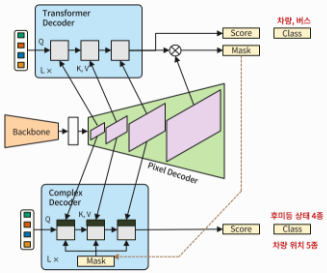
모델의 추론 성능을 높이기 위해 테스트 시에 여러 가지 데이터 증강 기법(Flip)을 적용하는 기법

원본에 대한 예측값과 데이터 증강에 대한 예측값의 평균을 최종 예측값으로 지정

Origin image

Flip image

Model



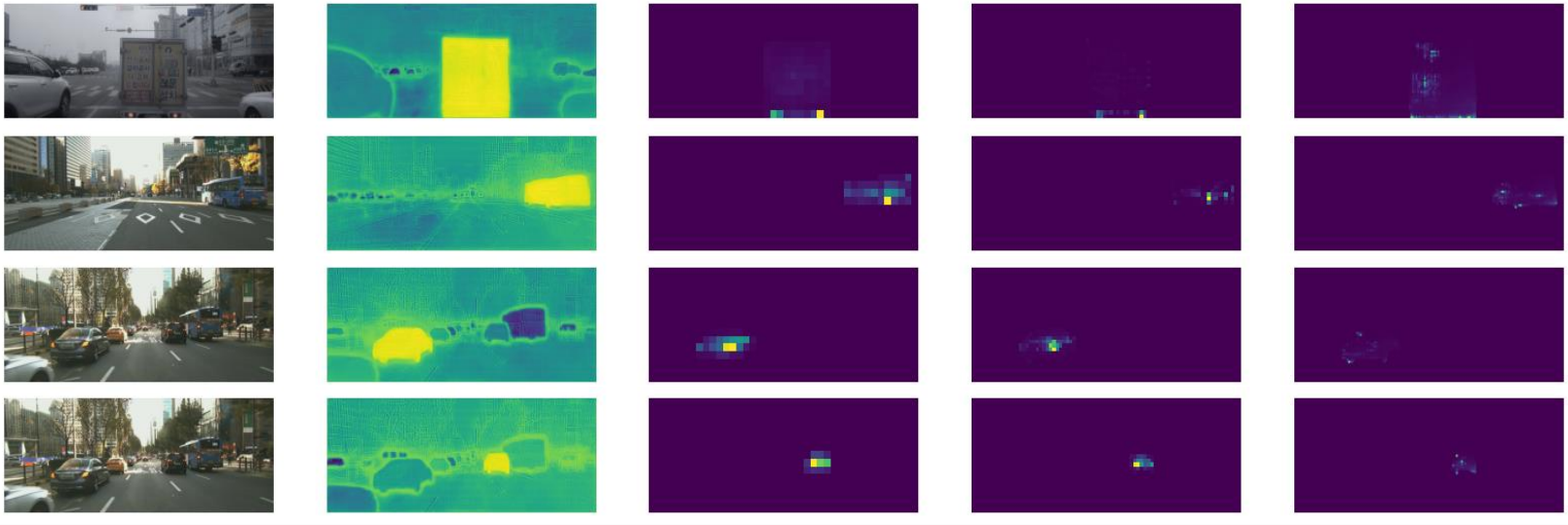
Avg

Final Prediction

04. 실험 결과

정성 평가 및 정량 평가

State Decoder의 두번째 Layer와 세번째 layer의 예측값의 평균을 validation에 사용 했을 때 가장 좋은 성능 (0.7484 mAP)



▲ input

▲ Transformer Decoder Result

▲ State Decoder의 첫번째 layer

▲ State Decoder의 두번째 layer

▲ State Decoder의 세번째 layer

05. 결론 및 향후 계획

결론

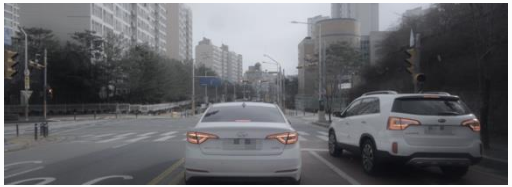
기존 Mask2Former 모델에 Post-processing 추가하여 차/버스 종류 뿐 아니라 후미등 상태 및 차량 위치 파악에서 baseline인 Yolo모델보다 더 우수한 정확도 보임

훈련 과정에서 다양한 데이터 증강 방법 및 TTA 사용하여 성능 향상

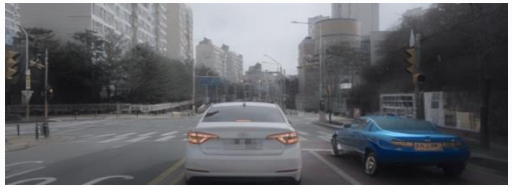
향후 개선 사항

Diffusion을 사용한 데이터 증강 기법 사용

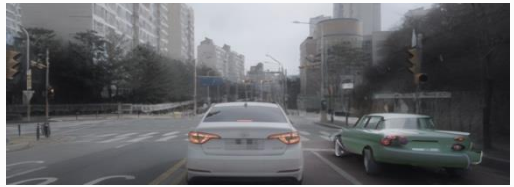
훈련에 앙상블 기법 도입



▲ origin image



▲ Diffusion을 사용한 합성 데이터 1



▲ Diffusion을 사용한 합성 데이터 2