Birds-eye-view Representation 변환을 통한 주변 환경 인식

우 수 한*

Abstract

먼 미래의 일인줄로만 알았던 자율주행기술이 인공지능 기술의 발전으로 우리의 일상에 점점 더 빠르게 가까워지고 있습니다. 자율주행을 위해서는 정확한 주변 환경 인식이 필요하며, 이를 위해 BEV representation을 통한 주변 환경 인식 기술이 활발히 연구되고 있습니다. BEV representation을 통한 주변 환경 인식은 차량 주변의 다양한 상황을 종합적으로 이해하는데 도움을 주고 차량의 경로설정에도 유리하기 때문에 Tesla를 비롯한 자동차 업계에서도 적극적으로 활용하고 있습니다.

본 글에서는 이렇게 자율주행에서 큰 역할을 하고 있는 BEV representation에 대한 소개와 함께 카메라에서 얻어진 perspective view(PV) image를 BEV representation 으로 변환하는 다양한 방법에 대해 소개하고자 합니다.

I. 서 론

최근 인공지능 기술의 발전으로 자율주행 관련 기술 개발이 빠르게 진행되면서 먼 미래의 일이라고 느꼈던 자율주행이 우리 주변으로 가까이 다가오고 있습니다. 자율주행의 첫 번째 단계는 현재 주행하고 있는 자동차(ego-vehicle)의 주변 상황에 대한 정확한 인식입니다. ego-vehicle 주변에 통행하고 있는 다른 자동차, 오토바이, 보행자와 같은 동적 객체들과 주차된 자동차, 도로의 장애물, 도로의 영역정보(차도, 인도 등)등 정적인 객체들을 모두 포함한 주변환경 전체에 대한 포괄적인 이해가 필요합니다.

* 연세대학교 전기전자공학부 Computational Intelligence Lab, 박사과정, wsh112@yonsei.ac.kr

이런 관점에서 위에서 바라본 새의 눈 시점인 bird-eye-view(BEV) 형식으로 환경을 인식하고 처리하는 것이 최근 많은 관심을 받고 있습니다. BEV representation은 운전자의 시점과 같도록 설치된 하나 혹은 여러 개의 카메라에서 BEV 수직한(일반적으로 차량 전방에 정면으로 부착된 카메라의 시점) perspective view(PV)로 얻어진 이미지를 평면으로 변환하는 것으로 구현됩니다. 카메라가 촬영한 이미지를 거리에 따라 투영하여 모든 물체를 단일 평면 상에 표시합니다.

이러한 BEV representation을 사용하면 자율주행 차량이 주변 환경을 더욱 자세하게 이해할 수 있습니다. 또한, BEV representation은 사람이 지도를 보는 방식과 완전히 같은 방식이므로 차량 주행 경로를 계획하는 데 더욱 직관적으로 사용될 수 있습니다. 이는 자율주행 차량이 주행 중에 다른 차량, 보행자 또는 도로 표지판과 충돌하지 않도록 안전한 경로를 계획하는 데 도움이 됩니다.

그림 1 Tesla autopilot 기술의 BEV representation을 통한 주변 환경 인식

BEV representation은 다양한 자율주행 시스템에서 사용되고 있습니다. 예를 들어, 가장 높은 수준의 자율주행 기술을 보유한 것으로 평가받는 Tesla에서도 BEV representation을 사용하여 차량의 주변 환경을 분석하고, 안전하고 효율적인 주행 경로를 계획하고 있습니다(그림 1). 이와 같이, BEV representation은 차량 주변의 다양한 상황을 종합적으로 이해하는데 도움을 주고 차량의 경로설정에도 유리하기 때문에 미래의 교통 시스템에서 활발

히 사용될 것으로 예상됩니다. 본 글에서는 PV image를 이용해 주변 상황에 대한 여러 정보가 표현된 BEV representation을 생성하기 위한 다양한 방법에 대해 소개합니다.

Ⅱ. 기하학을 기반으로 한 BEV 변환

PV image를 BEV representation으로 바꾸는 전통적이고 직관적인 방법은 두가지 view 간의 자연스러운 기하학적 투영 관계를 사용하는 것입니다. 이런 기하학 기반 방법들은 두 view간의 간극을 어떻게 극복하느냐에 따라 크게 Homography 기반 방법과 depth 기반 방법 두 그룹으로 나눠집니다. Homography는 두 view상의 좌표들 사이의 선형변환관계를 의미합니다. PV와 BEV 사이에는 단순한 선형변환 이상의 정보가 필요하므로 homography 기반 방법들은 단순화된 기하학적 관계만을 사용하는 초기 연구들이나, 수평면에 대한 인식에만 초점을 맞춘 연구들이 대부분입니다. 반면 depth 기반 방법들은 실제 시나리오에 대해 조금 더 일반적인 방법이라고 할 수 있습니다. 이 두 가지 방법 유형에 대해 자세히 다뤄보도록 하겠습니다.

1. Homography-based methods

3D 공간의 점들은 perspective mapping에 의해 완벽하게 image 공간으로 변환될 수 있지만, 2D 공간상의 image 픽셀을 3D 공간으로 투영하는 문제는 명확한 솔루션이 존재하지 않는 ill-posed problem입니다. Inverse Perspective Mapping(IPM) [1]은 3D공간으로 매핑되는 점들이 하나의 수평면 위에 존재한다는 추가 제약 조건하에서 수학적으로 불가능한 2D to 3D 매핑 문제를 해결하기 위해 제안되었습니다. 이는 front view image를 BEV image로 변환하는 데 대한 선구자적인 작업입니다. 변환은 카메라 rotation homography에 이어 anisotropic scailing을 적용합니다[2]. Homography matrix는 카메라의 intrinsic, extrinsic parameter를 통해 물리적으로 유도할 수 있습니다. 일부 방법 [3]은 컨볼루션 신경망(CNN)을 사용하여 PV image에서 수직 소실점과 지면 소실선을 추정하여 homography matrix를 결정합니다. IPM 작업을 통해 PV image를 BEV image로 변환한 후, optical flow estimation, detection, segmentation, path planning 다양한 task를 BEV image를 기반으로 수행할 수 있습니다. VPOE [4]는 Yolov [5]을 사용하여 BEV image에서 차량 위치와 방향을 추정합니다.

그림 2 Inverse Perspective Matching 예시.([40])

하지만 이러한 IPM 기반 접근법은 변환된 point들이 한 평면(지면)에 존재한다는 가정에 크게 의존하므로, 건물, 차량 및 보행자와 같은 지면에 붙어있지 않은 물체까지 정확하게 변환하는 것은 어렵습니다(그림 2). 따라서 일부 방법은 이런 지면 위에 위치하지 않는 물체에 대한 왜곡을 줄이기 위해 각 픽셀의 semantic 정보를 활용합니다. OGMs[6]은 차량의 footprint segmentation 결과를 PV에서 BEV로 변환할 때 이용하여, 한 평면에 존재한다는 가정을 따르면서도 차체가 지면보다 위에 위치하면서 발생하는 왜곡을 방지합니다.

일부 접근 방식은 전처리나 후처리 단계에서 IPM을 적용하는 대신, 딥러닝 네트워크 training 단계에서 feature map을 변환하는 데 사용합니다. Cam2BEV[7]는 각 뷰의 feature map을 변환하여 다중 차량 카메라 image를 사용하여 차량 주변의 전체 BEV 시맨틱 맵을 얻습니다. MVNet[8]은 IPM을 기반으로 2D feature를 BEV 공간에서 공유하여 multi-view feature를 aggregation 합니다. [9]에서는 2D image에서의 detection prediction을 통해 3D detection box를 최적화하는 방법을 제안하고, 2D와 3D 공간 사이의 homography loss를 통해 네트워크가 2D image와 BEV image간의 기하학적 제약 조건을 학습하도록 했습니다.

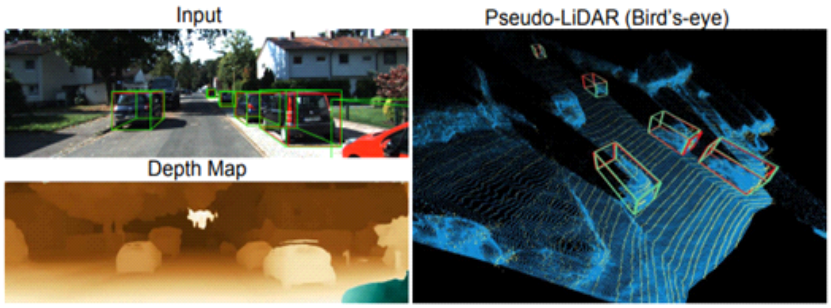
이렇게 homography 기반 방법들은 평평한 지면에 대한 가정을 기반으로 PV image와 BEV image를 물리적으로 매핑하는 방법입니다. IPM은 다양한 perception task에서 image 프로젝션이나 feature 프로젝션에 적용됩니다. Mapping 절차는 행렬 곱셈만을 통하므로 직관적이며, 학습이 필요하지 않아 효율적인 선택입니다. 하지만 실제 PV에서 BEV로의 변환은 ill-posed problem이기 때문에 IPM은 강력한 가정인 평면 가정을 통해 BEV 변환의 아주 일부분만을 해결했다고 볼 수 있습니다.

2. Depth-based methods

IPM 기반 방법은 모든 점이 지면에 있다는 가정을 기반으로 하고 있습니다. 이것은 2D image 상의 공간과 3D BEV image 상의 공간을 연결하는 feasible한 방법을 제공하지만 중요한 정보 중 하나인 객체의 높이정보에 대한 희생이 필요합니다. 이러한 높이정보의 희생을 피하면서 2D상의 pixel 또는 feature를 3D 공간으로 변환하기 위해서는 2D image의 각 픽셀에 대응되는 depth 정보가 필요합니다. 이런 관점에서 depth 정보를 이용하여 PV image를 BEV representation으로 변환하는 방법이 depth정보를 기반으로 한 방법입니다.

Depth를 기반으로 BEV 변환을 하는 방식들은 monocular image에서 depth 정보를 예측해야하기 때문에 보통 depth 예측과 객체 검출이 구별되어 이루어집니다. 예측된 depth를 이용해 검출된 객체를 정확한 3D 공간에 mapping하는 방법을 사용합니다. Image를 통해 depth를 예측하기 위해 Pseudo-LiDAR[10]와 같은 point-based 방식을 사용하거나, OFT[11]와 같은 voxel-based 방식을 사용합니다. MonoDepth[12]와 같은 방법을 사용하여 image pixel 각각의 depth를 나타내는 depth map을 생성해서 사용하기도 합니다.

그림 3 Depth based method 중 하나인 Pseudo-LiDAR의 BEV 예시.([10]의 Fig.2.2)



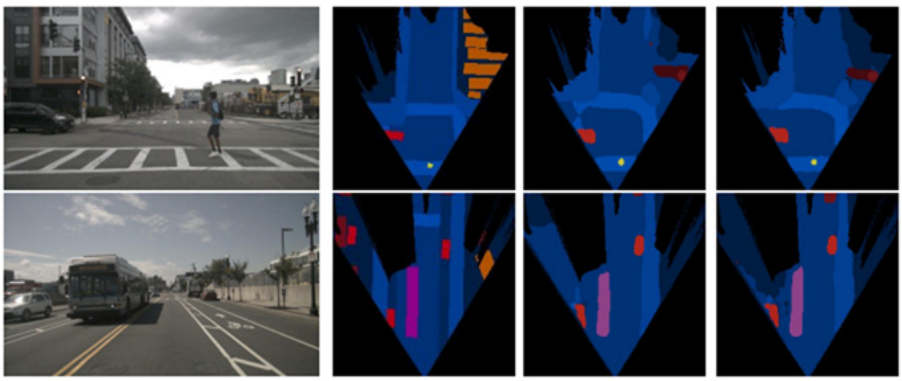
이렇게 예측된 depth와 함께 lane segmentation network [13], [14], [15], [16], object detection network [17], [18], [19], [20] 등등 여러가지 객체 검출 네트워크를 사용하여 목적에 맞는 형식의 BEV representation을 구현합니다.

Ⅲ. Neural Network를 기반으로 한 BEV 변환

1절에서는 카메라 투사 과정의 물리적 원리를 명시적으로 활용하여 PV에서 BEV representation으로의 변환을 진행합니다. 이렇게 물리적 원리를 사용한 방법은 결과물의 해석이 용이합니다. 이런 viewpoint 변환의 또 다른 대안은 데이터 기반 방식으로, 입력 image와 출력 image 간의 대응 관계를 학습하여 PV에서 BEV로 변환하는 모델입니다. 이 방식에서는 별도의 기하학적 계산이나 변환 없이, 딥러닝 네트워크가 PV와 BEV 간의 대응 관계를 학습하여 image를 변환합니다.

Network를 기반으로 한 방법은 [그림 1]과 같이 BEV representation을 생성하기 위해 object detection, lane detection, road segmentation 등등 여러 task를 따로 수행한 후 그 결과를 합쳐서 결과를 뽑아내는 기존 방법들과 달리 PV image와 여러 task의 결과물이 합쳐진 grount truth BEV map을 1:1로 대응시켜서 하나의 네트워크만을 가지고도 그림 4와 같이 여러 task를 한번에 해결할 수 있다는 장점이 있습니다.

그림 4 End-to-end network인 Image2Map[38] 결과 예시([38]의 Fig.2)



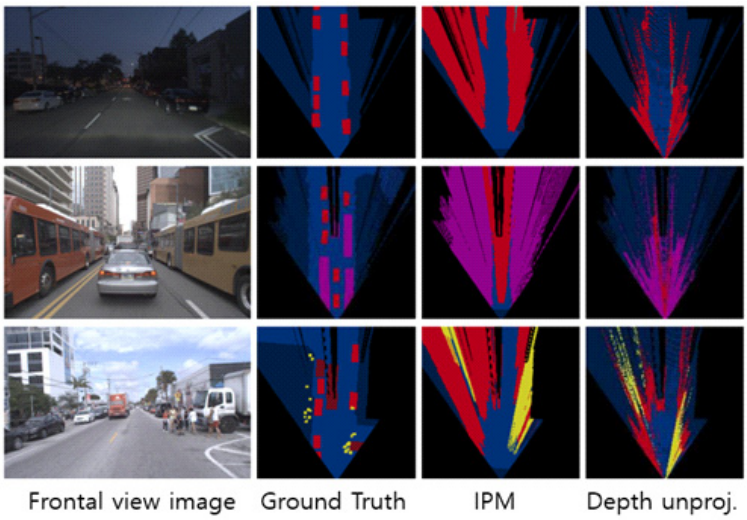
이런 방법의 단점은 대부분의 딥러닝 네트워크의 기본적인 단점을 공유합니다. 딥러닝 네트워크를 이용하기 때문에 결과에 대한 해석이 어렵다는 점과, 딥러닝 네트워크의 학습을 위해서 많은 양의 입력 image와 출력 image 쌍 데이터가 필요하다는 것입니다.

하지만 BEV representation 변환 문제에서는 다른 문제들에 비해 학습에 필요한 데이터를 생성하는 것이 상당히 어렵습니다. image 각각의 객체에 대응하는 box정보만이 필요한

object detection 데이터, 각각 pixel에 하나하나의 class를 mapping해야하는 semantic segmentation 데이터는 하나의 image로 네트워크 학습을 위한 데이터 생성 작업을 진행할 수 있습니다. 하지만 BEV representation을 위한 데이터는 자동차, 사람에 의해 가려진 부분의 뒷부분 정보, 그리고 정확한 거리정보까지 상당히 많은 정보가 필요하므로 단일 image 상에서는 데이터를 생성할 수 없고 연속된 여러 image와 정확한 거리정보까지 모두 이용하여 데이터를 생성해야 합니다. 예를 들어 단순히 PV image에 대한 2D semantic map을 앞서 소개한 homography 기반 방식들을 이용해 변환하면 그림 5와 같은 결과가 나와서 네트워크의 학습에는 이용할 수 없습니다. 하지만 최근 BEV representation에 대한 관심 증가와 함께 관련된 public dataset[21], [22]의 등장으로 네트워크를 이용한 BEV 변환 연구가 활발히 진행되고 있습니다.

Network based 방식에서 사용되는 딥러닝 네트워크는 크게 Multilayer perceptron (MLP) 기반 방식과 Transformer 기반 방식으로 나눌 수 있습니다. MLP는 입력과 출력을 연결하는 fully connected layer를 가진 뉴럴 네트워크의 기본적인 구조입니다. 반면에 Transformer는 attention mechanism을 사용하여 입력과 출력의 관계를 모델링하는 구조로, 자연어 처리를 위해 제안된 구조이지만[Attention is all you need], 최근 computer vision 영역에서도 뛰어난 성능을 보이고 있습니다. 각각의 방식을 사용한 방법을 몇 가지 소개하도록 하겠습니다.

그림 5 **BEV 생성 네트워크를 학습하기 위한 GT제작의 어려움([26]의 Fig.4)**



1. Multilayer Perceptron(MLP) 기반 방법

Multilayer perceptron, 즉 MLP는 어느정도 복잡한 mapping 함수로써 기능할 수 있으며, 서로 다른 modality(ex. RGB image + Thermal Image), 서로 다른 차원, 또는 서로 다른 표현의 입력들을 출력으로 매핑하는 데에서 이미 인상적인 성과를 거두고 있습니다. 일부 방법들은 서로 다른 뷰(PV & BEV)의 매핑에 대한 camera calibration 자체를 MLP를 이용해 학습합니다.

VED [23]은 MLP bottleneck 구조를 사용한 variational encoder-decoder 구조를 사용하여 주행 장면의 PV image를 2차원 semantic 정보가 포함된 BEV Cartesian 좌표계상의 map으로 변환합니다. VED는 2D image에서 semantic 정보가 포함된 BEV representation으로 변환하는 문제를 end-to-end로 학습가능한 네트워크를 처음으로 사용해서 해결한 연구입니다. VPN [24]은 이러한 end-to-end로 학습가능한 네트워크를 사용한 방법에서 여러 카메라를 사용할 때 모든 카메라의 정보가 포함된 global feature를 만들기 위해, two-layer MLP를 통해 flattening-mapping- reshaping 하여 feature의 view를 변환하고 이를 통해 각 PV image의 feature map을 하나의 BEV feature map으로 변환합니다. VPN의 뷰 변환 모듈을 기반으로 FishingNet [25]은 PV image의 feature를 BEV 공간 feature로 변환하고 레이더 및 LiDAR 데이터와 결합하여 multimodal 인식 및 예측을 수행합니다.

PON [26] 및 STA-ST [27]은 feature pyramid [28]를 활용하여 여러 해상도에서 image 피처를 추출하여 receptive field를 넓힘과 동시에 보행자와 같은 작은 객체에 대해서도 정확한 feature를 얻을 수 있도록 합니다(그림 a). 그렇게 얻은 feature에 MLP를 적용해 feature에서 높이(H) 축을 없애고 깊이축(Z)이 생기도록 확장하여 뷰 변환을 수행합니다(그림a). 이러한 설계는 PV image에서 BEV representation으로 변환하는데 있어서 feature의 넓이축 보다는 높이축 방향에서 많은 정보가 필요하다는 것을 기반으로 합니다. 수평 방향에서는 간단한 카메라 기하학을 사용하여 BEV representation에서의 위치와 PV representation에서 위치 간의 관계를 mapping할 수 있지만, 수직 방향에서는 다른 물체에 의한 가려짐, 해당 pixel에 대한 깊이 정보 부족, 알려지지 않은 지형등으로 인해 그 pixel이 위,아래의 context정보가 필요합니다. 이러한 수직 방향 뷰 변환 아이디어는 추후 소개드릴 Transformer 기반 방법에서도 사용됩니다.

그림 6 PON[26]의 feature pyramid 방식을 사용한 네트워크 구조([26]의 Fig2)

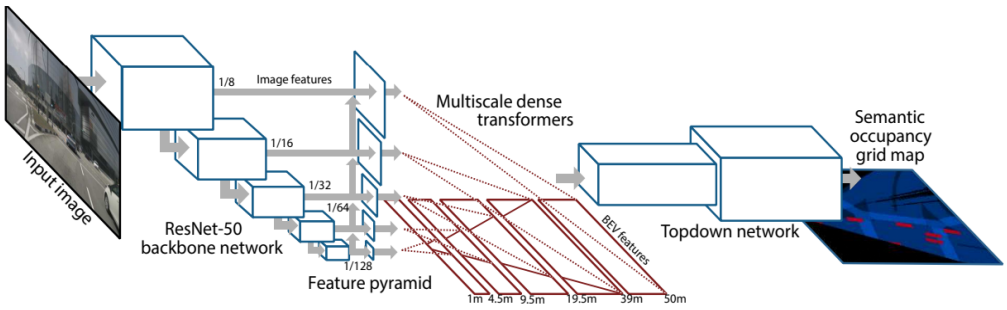
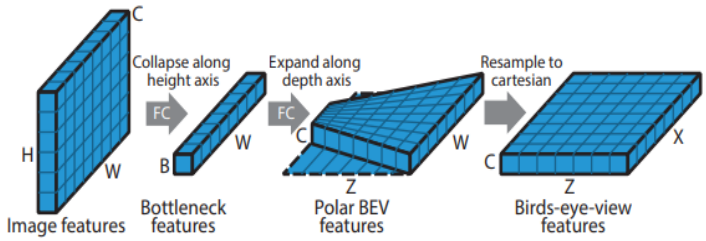


그림 7 PON의 MLP를 이용한 PVtoBEV feature transform([26]의 Fig3).

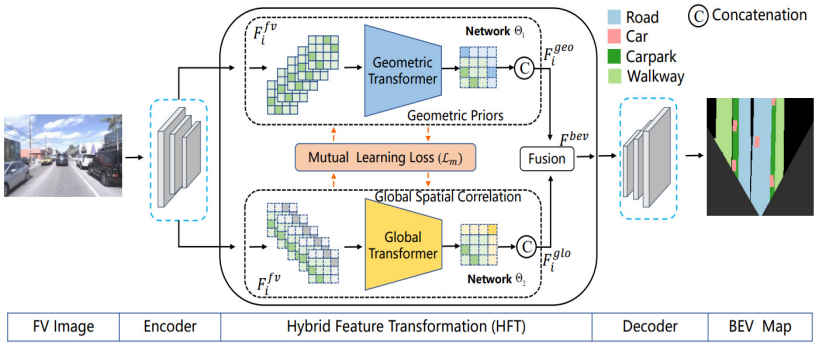


HDMapNet [29]도 위와 같은 MLP 기반의 feature projection 전략을 채택하여, instance embedding과 여러 방향의 카메라의 방향을 BEV 형식의 vectorized map elements로 생성하는것을 목표로 합니다. HDMapNet에서는 PV to BEV의 단방향 projection 만으로는 PV image의 정보가 효과적으로 전달됐는지 확인하기 어렵기 때문에, 추가적인 MLP를 사용하여 반대로 BEV feature를 PV feature로 다시 reprojection하여 매핑이 올바른지 다시 확인하는 bidirectional projection 방법도 사용합니다. 이 bidirectional projection을 기반으로, PYVA [30]는 view projection을 강화하기 위한 cycled self-supervision 체계를 제안합니다. 또한, attention-based feature selection 과정을 도입하여 두 view의 feature를 상호 연관시켜 더 정확한 BEV representation을 생성해 낼 수 있는 더 강력한 BEV feature를 얻습니다.

HFT [31]는 camera model 기반 feature transformation과 camera model-free feature transformation의 장단점을 분석합니다. 전자는 IPM 기반 방법으로 지역 도로 및 주차장과 같은 영역에서 PV-to-BEV transformation을 쉽게 처리할 수 있지만, 평면 지면 가정

에 의존하기 때문에 지면 위쪽 영역에서 왜곡이 발생합니다. 후자인 MLP-based 방식은 이러한 왜곡을 피할 수 있지만 geometric prior가 없어 수렴이 느립니다. 두 가지 접근법의 이점을 활용하고 각각의 단점을 피하기 위해, HFT는 geometric 정보를 활용하는것과 global context를 학습하는 두 가지 branch로 구성된 하이브리드 feature transformation을 사용했습니다.

그림 8 Hybrid feature transformation을 사용하는 HFT.([31]의 Fig2.)



MLP 기반 방법은 카메라의 calibration matrix등이 없어도 MLP 자체를 일반적인 매핑 함수로 사용하여 PV to BEV 변환을 수행합니다. MLP는 이론적으로는 모든 함수를 근사할 수 있지만 실제 사용시에는 image의 depth 정보 부족, 다른 객체에 의해 가려지는 등의 원인으로 BEV 형식을 추론하는 것은 어려운 일입니다. 또한 MLP 기반 방법은 일반적으로 Transformer 기반 방법보다 성능이 좋지 않은 편입니다. 다음 절에서는 최근에 제안된 Transformer 기반 방법들을 소개하겠습니다.

2. Transformer 기반 방법

Transformer의 cross attention을 사용하는 방법은 MLP와 같이 카메라 모델을 직접 활용하지 않고도 관점 변환을 수행할 수 있습니다. MLP 기반 방식과 transformer 기반 방식의 주요한 차이점은 세 가지입니다. 첫 번째로, MLP는 layer의 weight가 inference 중에는 고정되어 다양한 연산을 수행하는데 어려움이 있지만, transformer에서는 attention의 weight가 입력데이터에 따라 변하기 때문에 데이터에 따라 더 적절한 연산을 수행할 수 있게 됩니다. 두 번째로, 자연스럽게 픽셀의 위치정보까지 같이 입력되는 MLP와는 달리 tra-

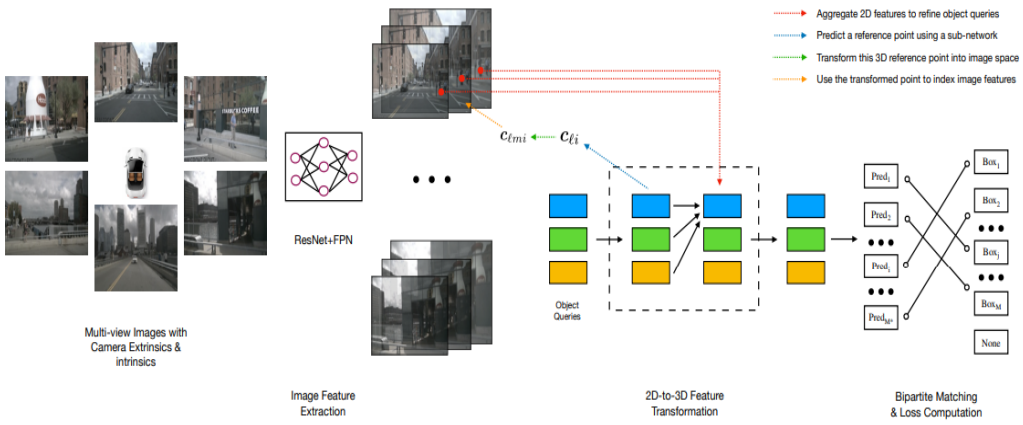
nsformer는 positional embedding 없이는 permutation-invariant한 특성을 갖습니다. 마지막으로 MLP와 같이 PV에서 BEV를 생성하는 정방향 뷰 변환을 처리하는 대신, transformer 기반 방식들은 BEV 각 위치에 대한 쿼리를 먼저 구성하고 attention mechanism을 통해 해당 쿼리 위치에 대한 image feature를 통해 쿼리에 대한 값을 구하는 방식을 사용합니다.

Transformer를 이용해 PV image를 BEV로 변환하는 연구는 Tesla에서 처음으로 연구되었고 21년 8월 Tesla AI day에서 발표했습니다. 이 방법을 통해 현재 최고수준의 자율주행을 시제품에서 선보이고 있으므로(그림 1) 관련 발표 이후 transformer를, 특히 transformer의 cross attention을 이용한 PV to BEV 변환 연구가 활발히 이루어지고 있습니다.

Transformer를 이용한 BEV 변환 기술은 앞서 언급했던 MLP와의 차이점인 query 생성 방식에 따라 크게 sparse query 방식과 dense query 방식, 또 그 두가지 방식을 같이 사용하는 hybrid query 방식으로 구분할 수 있습니다. 다음으로는 각 방식의 대표적인 연구들을 소개하고 분석해보도록 하겠습니다.

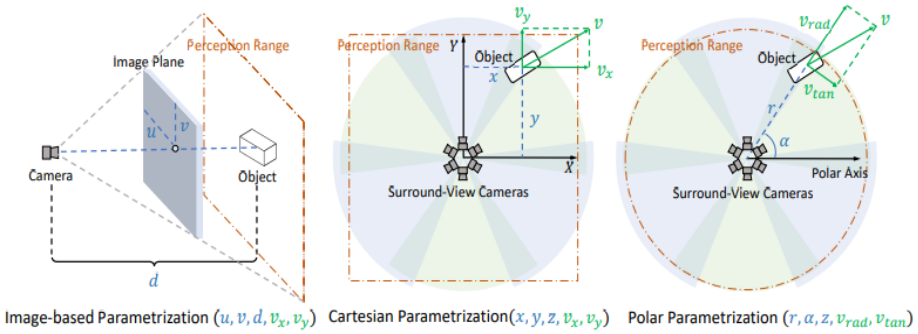
Sparse query 기반 방식들은 dense한 image feature를 전부 transform 하는 것이 아니라 미리 정해진 수의 query embedding을 통해 sparse한 인식 결과를 내놓습니다. 따라서 이런 방식은 sparse한 query의 한계로 dense한 scene 전체를 segmentation하는 방법들에는 적용되기 어렵지만 3D object detection과 같은 객체의 인식을 위한 방법들에 적용됩니다.

그림 9 DETR의 query 기반 detection framework를 3D로 확장한 DETR3D의 구조(출처 : [33]의 Fig1.)



Transformer를 사용한 3D object detection 연구에서는 자연스럽게 2D object detection task에서 인상적인 결과를 보인 DETR[32]에 영감을 받아 진행되었습니다. DETR3D[33]는 DETR의 query 기반 detection framework를 그대로 사용하면서 query에서 3D reference point를 추가로 예측하고, 그 point를 camera calibration matrix로 2D image상에 투영하고, 그렇게 투영된 위치에서 image feature를 sampling하여 3D object detection을 수행했습니다. PolarDETR[34]은 surround-view 카메라의 view symmetry를 inductive bias로 이용하여 더 빠르게 optimize하고 퍼포먼스를 향상시키기 위하여 bounding box 표현형식, nerwork prediction과 loss 계산을 포함한 polar parameterization 방법을 제안합니다.

그림 10 PolarDETR[34]의 polar parameterization([34]의 Fig1.)



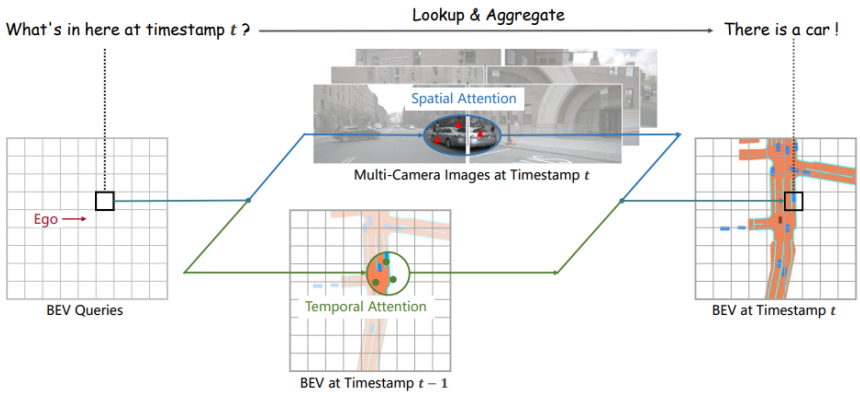
Dense-query 기반 방식은 sparse query 기반 방식과 다르게 3D 공간상 위치에 query들이 dense하게 미리 할당되어 있습니다. 따라서 Query의 수는 예측하고 싶은 BEV 공간상의 spatial resolution에 따라 달라지게 되고 보통 sparse-query 기반 방식보다 훨씬 많은수의 query 개수를 갖게 됩니다. Dense-query와 image feature간의 interaction을 통해 3D detection, segmentation등 여러 task를 해결해서 dense한 BEV representation 결과를 얻을 수 있습니다.

Dense-query 기반 방식은 Tesla에서 가장 먼저 사용했습니다. Tesla에서는 positional encoding과 context summary를 사용하여 BEV 공간에서 dense한 query를 생성한 다음, 생성된 query와 Tesla 자동차의 multi-view 카메라를 통해 얻은 multi-view feature 사이에 cross-attention을 적용하여 [그림 1]과 같은 BEV representation 결과를 얻습니다. 하지만 tesla에서 적용한 기본적인 transformer의 attention 연산은 query의 수에 따라 memory complexity가 기하급수적으로 상승하기 때문에, 해당 방식은 모델에 사용되는

image 해상도와 BEV representation의 해상도가 memory 사용을 줄이기 위해 제한되고 이는 모델의 성능을 방해하는 요소입니다.

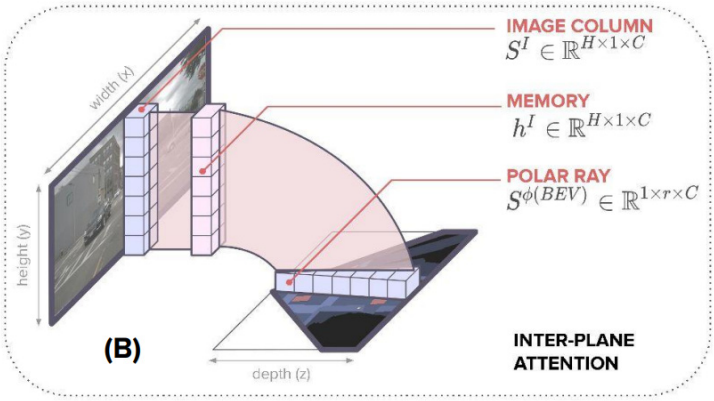
최근엔 이런 dense-query 기반 방식의 memory 이슈를 해결하기 위한 많은 연구들이 진행됐습니다. 가장 많이 사용되는 방법은 image 전체에 attention 연산을 진행하는 기본적인 attention 연산 대신 sparse한 위치에만 attention 연산을 수행하는 deformable attention[35]으로 대체하는 것입니다. BEV segmentation을 수행하는 모델인BEVSegFormer [36]과 3D lane detection을 수행하는 PersFormer[15]에서는 view transformation 모듈에 deformable detection을 적용했습니다. BEVFormer[37]에서는 view transform 부분뿐만 아니라 deformable attention을 서로다른 multiview camera의 image feature와 dense query사이의 interaction에도 사용했습니다.

그림 11 Dense-query 기반 방식인 BEVFormer[37]([37]의 Fig1.)



Attention 연산의 memory 문제를 해결하기 위해 geometry constraint를 이용하는 연구도 진행됐습니다. Image2Map[38]에서는 그림a와 같이 PV image상의 vertical scanline (image column)과 BEV representation상의 camera의 중심에서 뻗어나가는 하나의 polar ray가 1대1로 대응된다고 가정하고 BEV segmentation을 진행했습니다. 이 가정을 통해 view transform 문제를 1D sequence-to-squence 문제로 변환하여 2D image feature 전체의 attention을 1D line에 대한 attention으로 변환하여 연산량을 획기적으로 줄였습니다.

그림 12 Image column과 polar ray의 1:1 대응을 가정한 Image2Map[38](출처 : [38]의 Fig1.b)



앞서 살펴본것과 같이 sparse query를 기반으로 한 방식은 object 중심의 task를 해결에 적합하고, dense한 BEV representation을 생성하기는 어렵기 때문에 BEV segmentation에는 적합하지 않습니다. 이를 해결하기 위해 PETRv2[39] 에서는 dense한 segmentation query와 sparse한 object query를 같이 사용하는 방법을 택하여 BEV representation 변환을 수행했습니다.

이러한 transformer 기반 BEV 변환 방식들은 현재 강력한 관계 모델링 능력 및 데이터 종속적 특성으로 인해 기존 방식에 비해 확실한 성능을 보이고 있습니다. 또한 transformer 기반 방법은 원래 소개되었던 자연어 처리 분야 뿐만 아니라 computer vision 분야에서도 그 강력한 성능으로 인해 활발히 연구되고 있어 transformer 기반 네트워크의 발전 가능성은 무궁무진 하다고 할 수 있습니다.

Ⅳ. 결 론

지금까지 PV image를 BEV representation으로 변환하여 주변 환경을 인식하는 연구들에 대해 간략히 소개했습니다. 단순히 두 view간의 기하학적인 관계를 이용하여 이미지를 수직평면으로 확장하는 IPM부터, end-to-end 학습을 통해 변환을 수행하는 딥러닝 네트워크 기반 방식도 소개했습니다. 관련된 인공지능기술의 발전으로 많은 정보를 더욱 정확하게 BEV상에 표현할 수 있게 되고, 자동차에 탑재될 하드웨어의 발전이 함께 이루어져

더 빠른 속도로 복잡한 연산을 처리할 수 있게 되면 머지않아 상용차에서도 자율주행 자동차를 만나볼 수 있을것이라고 기대합니다.

참고문헌

[1] H. A. Mallot, H. H. Bülthoff, J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical flow computation and obstacle detection," Biological cybernetics, vol. 64, no. 3, pp. 177- 185, 1991.

[2] R. Hartley and A. Zisserman, Multiple view geometry in computer vision. Cambridge university press, 2003.

[3] S. Ammar Abbas and A. Zisserman, "A geometric approach to obtain a bird's eye view from an image," in Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, 2019, pp. 0–0.

[4] Y. Kim and D. Kum, "Deep learning based vehicle position and orientation estimation via inverse perspective mapping image," in 2019 IEEE Intelligent Vehicles Symposium(IV). IEEE, 2019, pp. 317-323.

[5] J. Redmon and A. Farhadi, "Yolov3: an incremental improvement. arxiv [preprint] arxiv," arXiv preprint arXiv:1804.02767, vol. 2, 2018.

[6] A. Loukkal, Y. Grandvalet, T. Drummond, and Y. Li, "Driving among flatmobiles: Bird-eye-view occupancy grids from a monocular camera for holistic trajectory planning," in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2021, pp. 51-60.

[7] L. Reiher, B. Lampe, and L. Eckstein, "A sim2real deep learning approach for the transformation of images from multiple vehicle- mounted cameras to a semantically segmented image in bird's eye view," in 2020 IEEE 23rd International Conference on Intelligent Transportation Systems(ITSC). IEEE, 2020, pp. 1?7.

[8] Y. Hou, L. Zheng, and S. Gould, "Multiview detection with feature perspective transformation," in Computer Vision – ECCV 2020 – 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII, ser. Lecture Notes in Computer Science, A. Vedaldi, H. Bischof, T. Brox, and J. Frahm, Eds., vol. 12352, 2020, pp. 1?18.

[9] J. Gu, B. Wu, L. Fan, J. Huang, S. Cao, Z. Xiang, and X. Hua, "Homography loss for monocular 3d object detection," CoRR, vol. abs/2204.00754, 2022.

[10] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Weinberger, "Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving," in CVPR, 2019.

[11] T. Roddick, A. Kendall, and R. Cipolla, "Orthographic feature transform for monocular 3d object detection," arXiv preprint arXiv:1811.08188, 2018.

[12] Godard, Cl ment, Oisin Mac Aodha, and Gabriel J. Brostow. "Unsupervised monocular depth estimation with left-right consistency." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

[13] N. Garnett, R. Cohen, T. Pe'er, R. Lahav, and D. Levi, "3d-lanenet: end-to-end 3d multiple lane detection," in IEEE International Conference on Computer Vision, 2019, pp. 2921?2930.

[14] Y. Guo, G. Chen, P. Zhao, W. Zhang, J. Miao, J. Wang, and T. E. Choe, "Gen-lanenet: A generalized and scalable approach for 3d lane detection," in European Conference on Computer Vision. Springer, 2020, pp. 666?681.

[15] L. Chen, C. Sima, Y. Li, Z. Zheng, J. Xu, X. Geng, H. Li, C. He, J. Shi, Y. Qiao, and J. Yan, "Persformer: 3d lane detection via perspective transformer and the openlane benchmark," in European Conference on Computer Vision(ECCV), 2022.

[16] R. Liu, D. Chen, T. Liu, Z. Xiong, and Z. Yuan, "Learning to predict 3d lane shape and camera pose from a single image via geometry constraints," in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, no. 2, 2022, pp. 1765?1772.

[17] S. Shi, X. Wang, and H. Li, "Pointrcnn: 3d object proposal generation and detection from point cloud," in IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[18] Y. Yan, Y. Mao, and B. Li, "Second: Sparsely embedded convolu- tional detection," Sensors, vol. 18, no. 10, 2018.

[19] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[20] T. Yin, X. Zhou, and P. Krähenbühl, "Center-based 3d object detection and tracking," CVPR, 2021.

[21] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multimodal dataset for autonomous driving," in CVPR, 2020.

[22] Behley, Jens, et al. "Semantickitti: A dataset for semantic scene understanding of lidar sequences." Proceedings of the IEEE/CVF international conference on computer vision. 2019.

[23] C. Lu, M. J. G. van de Molengraft, and G. Dubbelman, "Monocular semantic occupancy grid mapping with convolutional variational encoder-decoder networks," IEEE Robotics and Automation Letters, vol. 4, no. 2, pp. 445-452, 2019.

[24] B. Pan, J. Sun, H. Y. T. Leung, A. Andonian, and B. Zhou, "Cross- view semantic segmentation for sensing surroundings," IEEE Robotics and Automation Letters, vol. 5, no. 3, pp. 4867-4873, 2020.

[25] N. Hendy, C. Sloan, F. Tian, P. Duan, N. Charchut, Y. Xie, C. Wang, and J. Philbin, "Fishing net: Future inference of semantic heatmaps in grids," ArXiv, vol. abs/2006.09917, 2020.

[26] T. Roddick and R. Cipolla, "Predicting semantic map represen- tations from images using pyramid occupancy networks," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 11 138-11 147.

[27] A. Saha, O. Mendez, C. Russell, and R. Bowden, "Enabling spatio- temporal aggregation in birds-eye-view vehicle estimation," 2021 IEEE International Conference on Robotics and Automation(ICRA), pp. 5133-5139, 2021.

[28] T.-Y. Lin, P. Doll´ar, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2117-2125.

[29] Q. Li, Y. Wang, Y. Wang, and H. Zhao, "Hdmapnet: An online hd map construction and evaluation framework," arXiv e-prints, pp. arXiv-2107, 2021.

[30] W. Yang, Q. Li, W. Liu, Y. Yu, Y. Ma, S. He, and J. Pan, "Projecting your view attentively: Monocular road scene layout estimation via cross-view transformation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 15 536-15 545.

[31] J. Zou, J. Xiao, Z. Zhu, J. Huang, G. Huang, D. Du, and X. Wang, "Hft: Lifting perspective representations via hybrid feature trans- formation," arXiv preprint arXiv:2204.05068, 2022.

[32] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in European conference on computer vision. Springer, 2020, pp. 213-229.

[33] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in Conference on Robot Learning. PMLR, 2022, pp. 180-191.

[34] S. Chen, X. Wang, T. Cheng, Q. Zhang, C. Huang, and W. Liu, "Polar parametrization for vision-based surround-view 3d detection," arXiv:2206.10965, 2022.

[35] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable detr: Deformable transformers for end-to-end object detection," arXiv preprint arXiv:2010.04159, 2020.

[36] L. Peng, Z. Chen, Z. Fu, P. Liang, and E. Cheng, "Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs," arXiv preprint arXiv:2203.04050, 2022.

[37] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi- camera images via spatiotemporal transformers," arXiv preprint arXiv:2203.17270, 2022.

[38] A. Saha, O. Mendez, C. Russell, and R. Bowden, "Translating images into maps," in 2022 IEEE International Conference on Robotics and Automation(ICRA). IEEE, 2022.

[39] Y. Liu, J. Yan, F. Jia, S. Li, Q. Gao, T. Wang, X. Zhang, and J. Sun, "Petrv2: A unified framework for 3d perception from multi-camera images," arXiv preprint arXiv:2206.01256, 2022.

[40] https://towardsdatascience.com/a-hands-on-application-of-homography-ipm-18d9e47c152f