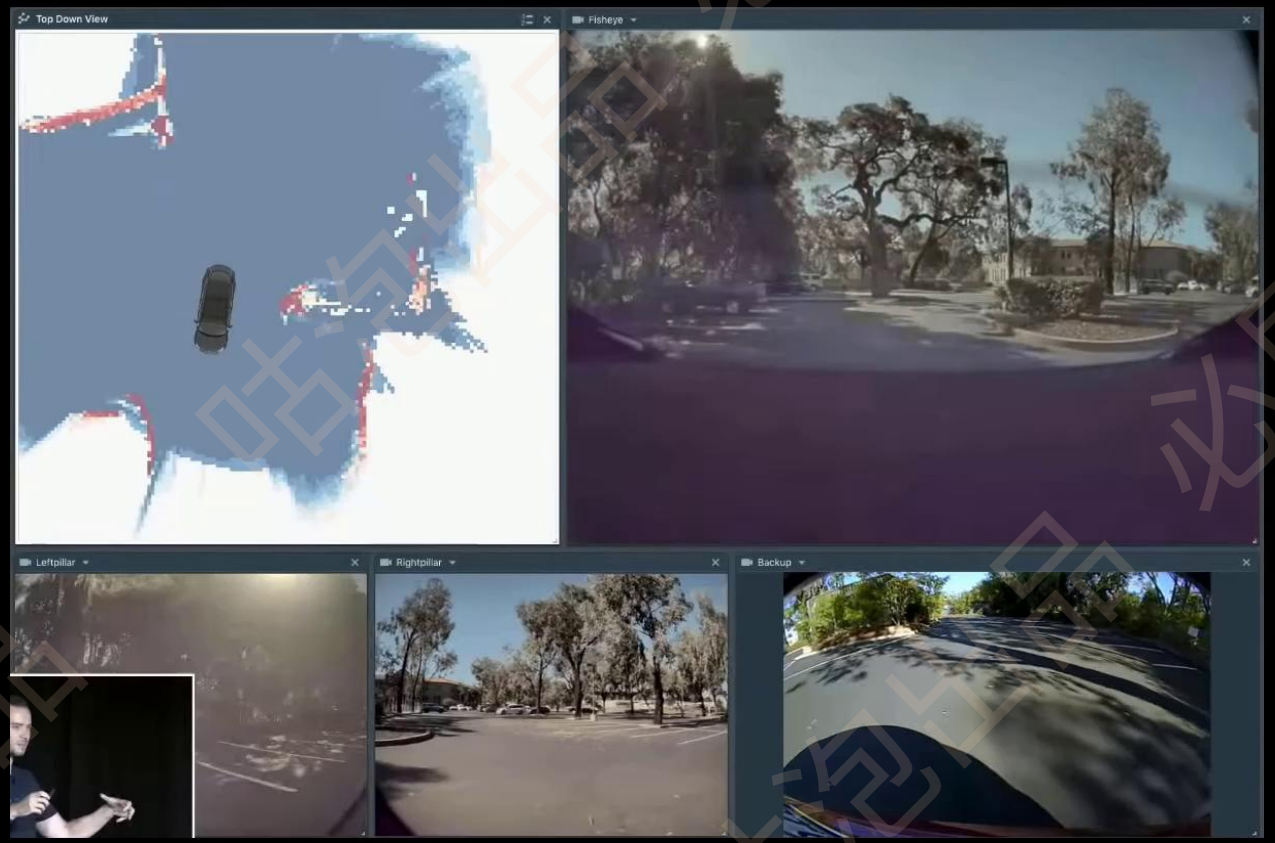
BEV

多个视觉摄像头，分别提取特征后如何聚合呢？



Problem 1
The across-camera fusion and the tracker are very difficult to write explicitly.

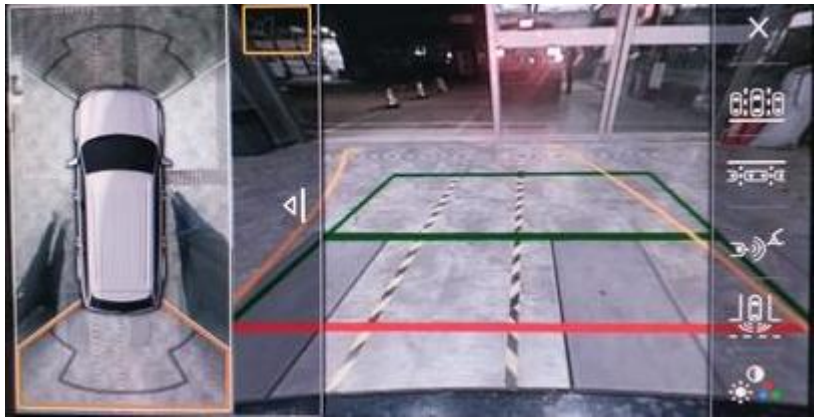
Problem 2
Image space is not the right output space.

BEV

基本上就是投影映射得到高维特征空间表示

BEV

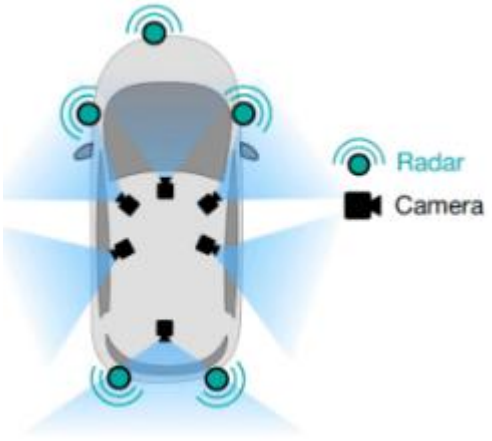
- BEV要解决一件什么事(上帝视角光环)
  - 跟驾驶相关的视觉任务基本都是多传感器融合，如何融合呢?
  - 后融合(结果)；前融合(数据)；特征级融合(这个是咱们的主题)
  - 将多传感器特征汇总在3D空间上，在这上面做预测，分析，决策
  - 而且BEV空间特征可以更容易拓展到下游任务

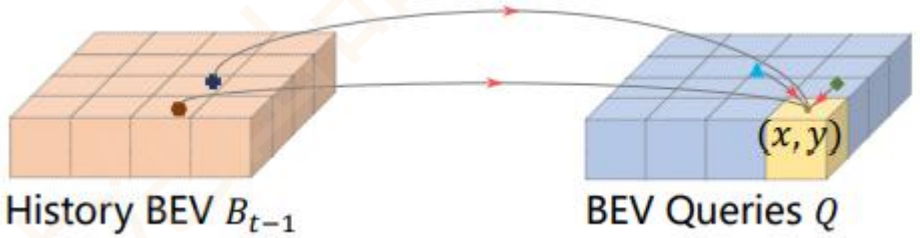


BEV

3D VS 4D

- BEV空间的特征我们可以当作是3D的，但是能不能再拓展一步呢?
- 如果再考虑到时间维度，那就是一个4D特征空间了，包括是时序信息
- 时序特征更适合预测速度，轨迹，检测等任务而且还可以进行 ‘猜想’





BEV

- 特征融合过程中可能遇到的问题
  - 自身运动补偿：车在运动，不同时刻之间的特征要对齐
  - 时间差异：不同传感器可能具有时间差，要对齐这部分信息
  - 空间差异：最后肯定都要映射到同一坐标系，空间位置特征也要对齐
  - 谁去对齐呢？肯定不是手动完成的，这些都交给模型去学习就好了

BEV

- BEV最终得到了什么
  - 相当于我们在上帝视角下重构了一个特征空间，空间的大小我们自己定义
  - 特征空间相当于一个网格，网格的间隔也可以自己定义，对应精度也会有差异
  - 在特征空间中，我们可以以全局的视角来进行预测，特征都给你了，咋用你来定
  - 难点：既想做的细致，还想节约计算成本，怎么办？BevFormer它来了

BEVFormer

- 比较出名，提供了一个基本框架
  - 一个核心：纯视觉解决方案（多个视角摄像头进行特征融合）
  - 两个策略：将Attention应用于时间与空间维度（其实就是对齐特征）
  - 三个节约：Attention计算简化，特征映射简化，粗粒度特征空间
  - 基本奠定了框架结构：时间+空间+DeformableAttention

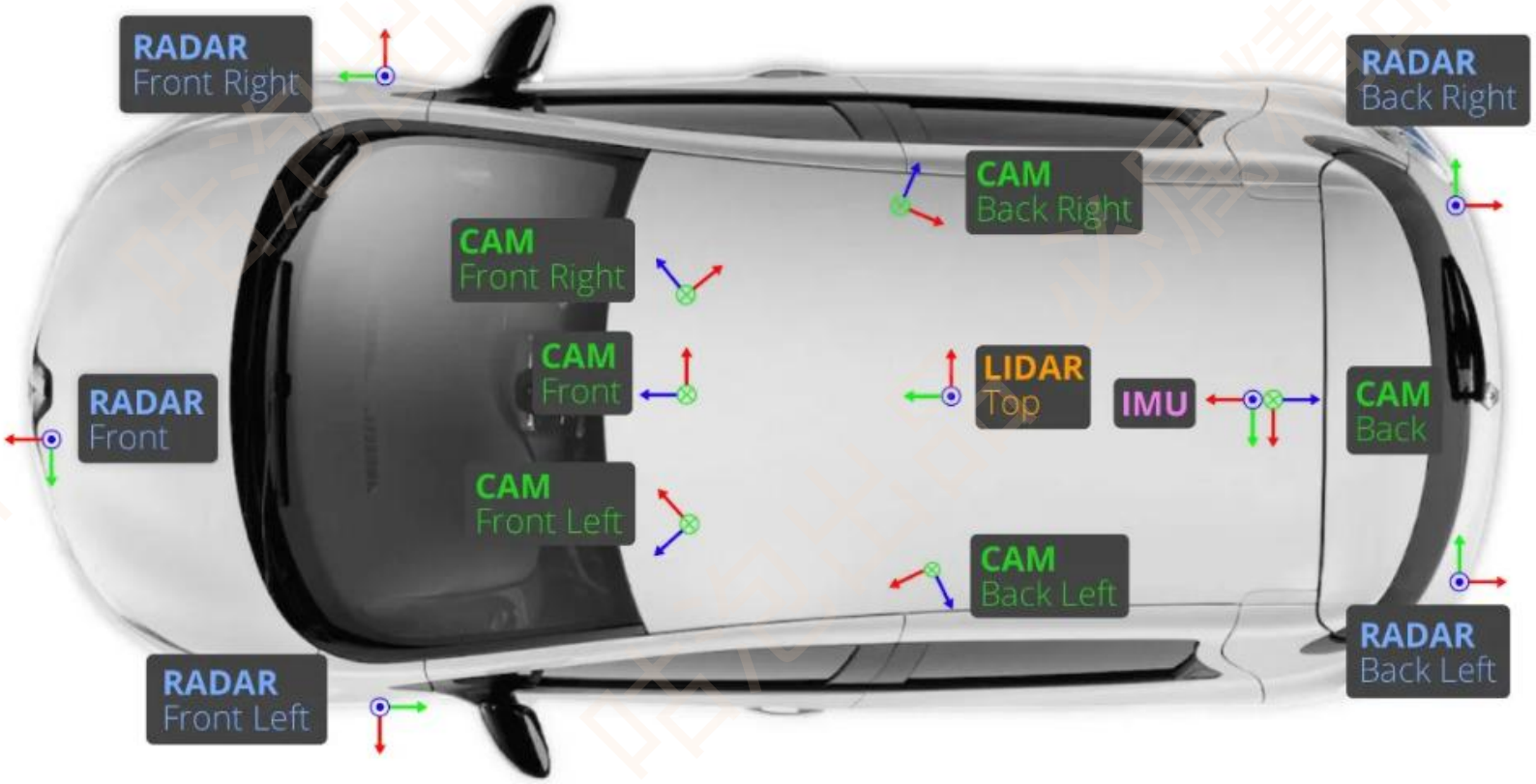
BEVFormer

- 输入数据格式
  - 输入张量（bs，queue，cam，C，H，W）
  - queue表示连续帧的个数，主要解决遮挡问题
  - cam表示每帧中包含的图像数量，nuScenes数据集中有6个
  - C，H，W分别表示图片的通道数，图片的高度，图片的宽度

BEVFormer

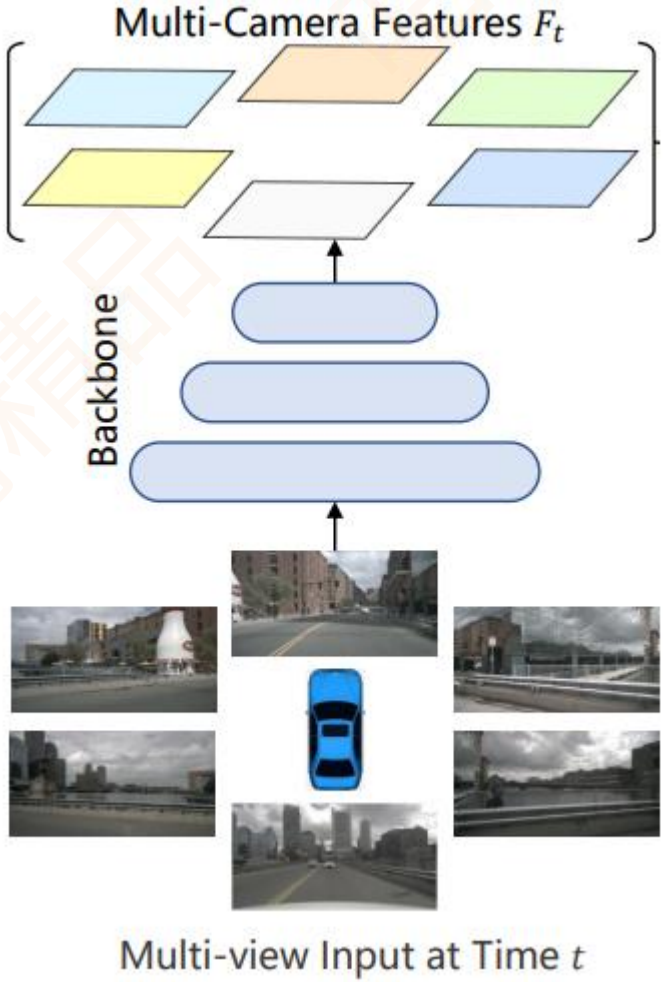
- 输入数据格式

- 咱们只用这6个视觉的CAM数据



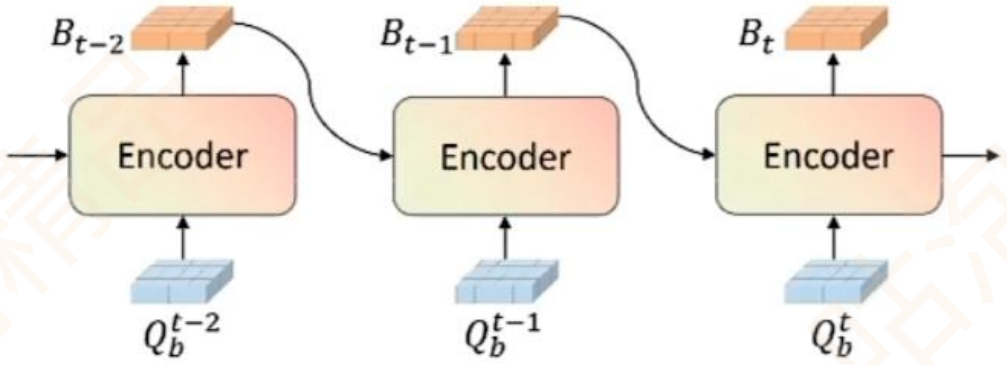
BEVFormer

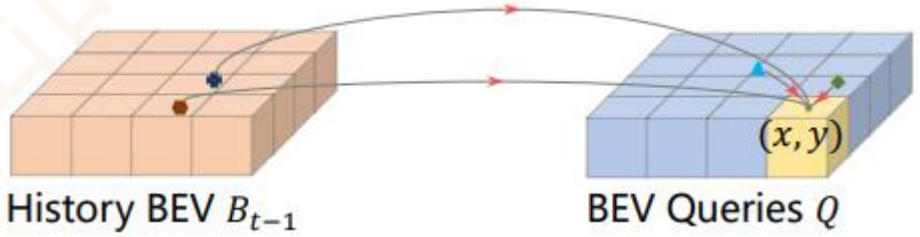
- Backbone
  - 这步没啥特别的，就是分别提特征就好了
  - 基本上啥Backbone都可以，但是最好速度快一些
  - 分别得到6个视角下的特征图，给后续空间注意力用
  - BEV特征空间中个每一个点都要在这些特征中采样



BEVFormer

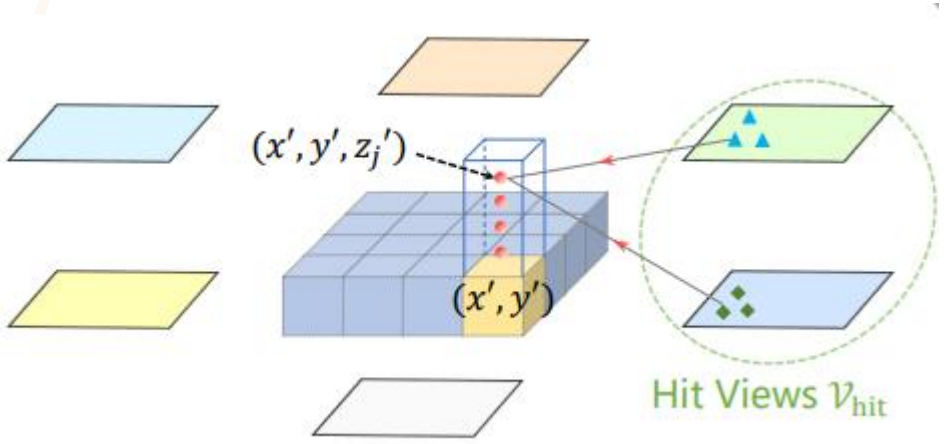
- 时间注意力模块
  - 类似RNN的方式来利用前面一些时刻的特征
  - 在时间上，不同帧中的车和周围物体都会有偏移，如何从历史中选需要的呢？
  - 其实依旧是DeformableAttention，这东西现在已经是硬流通货币了





BEVFormer

- 空间注意力模块
  - 要融合多个视角的特征相当于query会遍历所有视角找有用的信息
  - 3D空间的点要投影到2D空间，但是由于遮挡或者相机内外参不准
  - 那么投影点就是一个参考而已，在这个基础上附近再进行特征采样
  - 其实依旧是DeformableAttention的思想
    （Local > point >global）



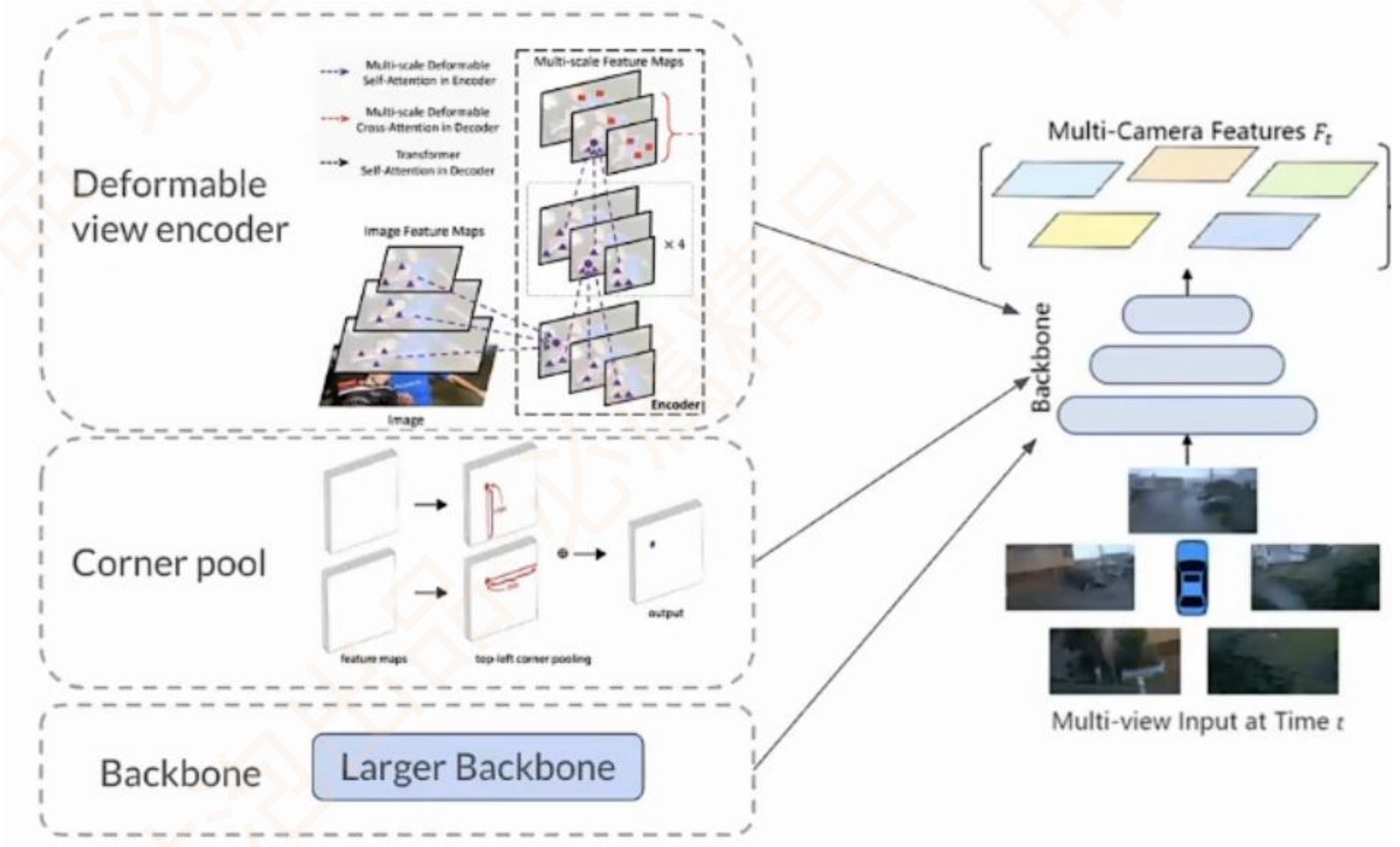
BEVFormer

- 空间注意力模块

- 如果特征空间是200*200那么就有4W的query
- 4W个query和6个视角特征算Attention，这个有点慢了
- 通过映射已经能得到投影关系，针对每一个视角只选择其中一部分query
- 这样就相当于能把4W个的计算量大约降低成类似6000个（原作者说的）

BEV

- 后续拓展升级
  - 多尺度特征图
  - Corner Pool(针对检测)
  - 突出角点特征
  - Backbone更强

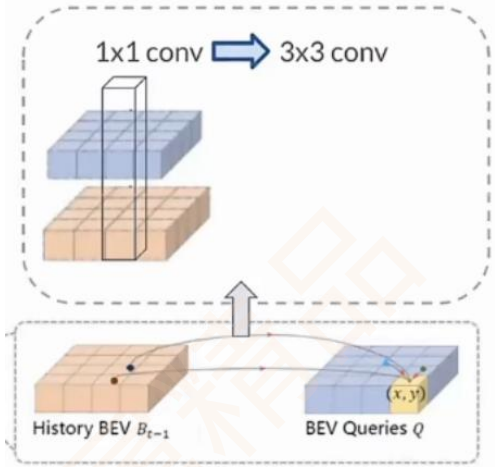


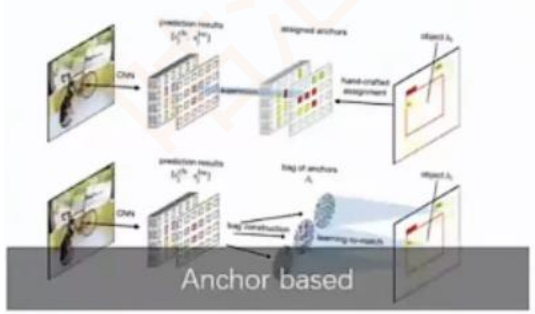
BEV

- 后续拓展升级

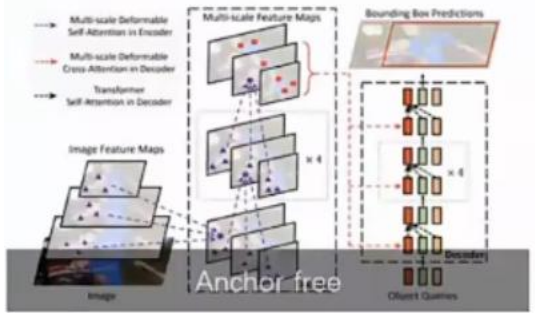
  - 偏移量的预测可以用更大的卷积核1->3

  - 多种检测器都可以用，然后做集成





- anchor based head收敛快，BEV下缓解遮挡
- *anchor based在车类等大物体上指标更好*



- DETR head基于Hungarian算法assign，对BEV下小物体更友好
- *anchor free在行人等小物体上指标更好*

BEV

- 时间与空间顺序
  - 先时间再空间，因为前面时刻BEV有很多信息
  - 相当于要充分利用先验知识，在此基础上再
  - 继续融合当前帧的特征，从而构建当前BEV
  - 重复多次（6次）得到最后的BEV空间特征