Nuscene-MMdetection数据转换分析

目录

[简介 1](#_Toc175154402)

[Nuscenes数据结构关联图 2](#_Toc175154403)

[Nuscenes数据结构具体定义 3](#_Toc175154404)

[Scene场景数据结构 3](#_Toc175154405)

[Sample定义 3](#_Toc175154406)

[激光传感器数据结构 4](#_Toc175154407)

[相机传感器数据结构 5](#_Toc175154408)

[车体位姿Ego pose数据结构 5](#_Toc175154409)

[Sensor calibration数据结构 6](#_Toc175154410)

[Annotation标注数据结构 6](#_Toc175154411)

[Instance数据结构 7](#_Toc175154412)

[MMdetection3D数据转换后数据结构 7](#_Toc175154413)

[Train\_info.pkl数据结构 8](#_Toc175154414)

[Camera数据结构 9](#_Toc175154415)

[MMDetection3d运行状态数据结构 11](#_Toc175154416)

[batch\_data 11](#_Toc175154417)

[Img\_metas 11](#_Toc175154418)

[LIDARInstance3DBoxes 12](#_Toc175154419)

[Detr3D算法结构与数据输入输出 13](#_Toc175154420)

[模型运行数据流 13](#_Toc175154421)

[模型训练数据流 14](#_Toc175154422)

# 简介

本文介绍了BEV算法中常用的Nuscenes数据集结构，并结合mmdetection3d框架下Detr3d算法中对Nuscenes数据的转换与利用，分析了三维数据集在相关算法中用于训练、推理时的适合结构

本文主要

https://mmdetection3d.readthedocs.io/zh-cn/latest/advanced\_guides/datasets/nuscenes.html

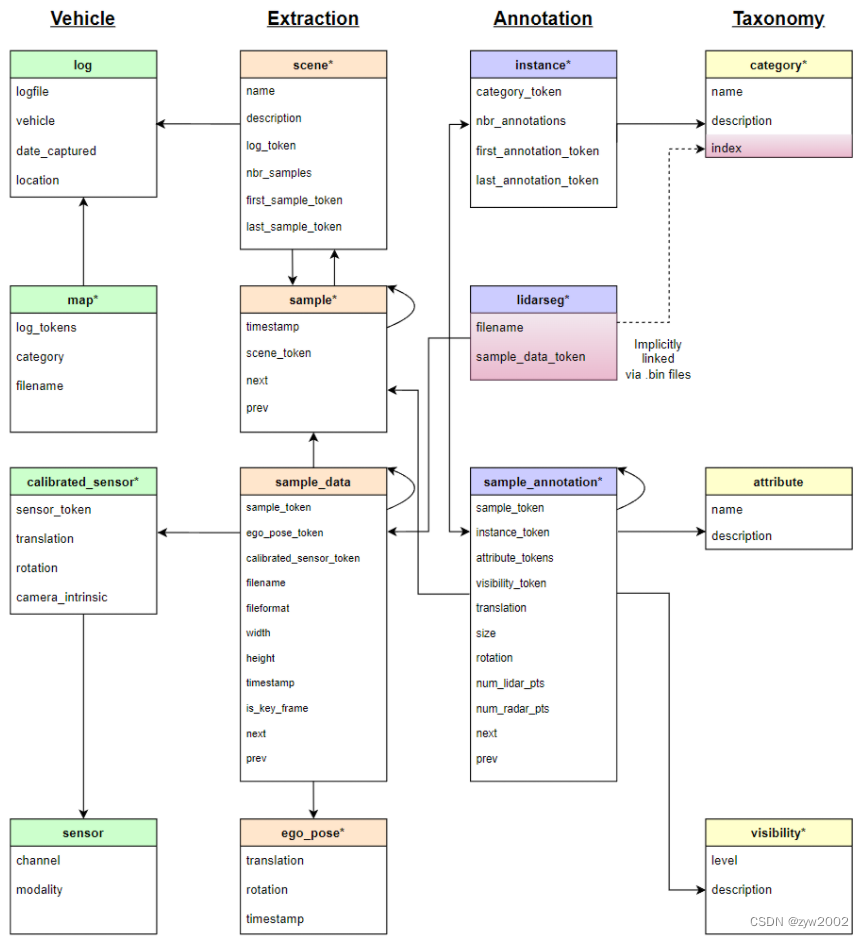
中，提到的利用

python tools/create\_data.py nuscenes --root-path ./data/nuscenes --out-dir ./data/nuscenes --extra-tag nuscenes

对nuscene数据的转换，实践分析基于mmdetection3d框架瞎，BEV 3D算法中所需要的数据

# Nuscenes数据结构关联图

下图给出了Nuscenes数据集中各个对象的关系图，具体对象定义在下文中给出



# Nuscenes数据结构具体定义

## Scene场景数据结构

Scene是Nuscenes的最上层数据结构，描述了一段时间（20s）内的数据

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| token | 标签 | str | '73030fb67d3cxx' |
| log\_token | Log信息token | str | '6b6513e6c8384cec |
| Nbr\_samples | Sample数量 | Int | 40 |
| First\_sample\_token | 首个sample的token | Str | 'e93e98b63d3b |
| Last\_sample\_token |  | Str | '40e413c922184255 |
| Name | 场景名称 | Str | 'scene-0001' |
| description | 描述场景 | str | 'Construction, maneuver between several trucks' |

## Sample定义

通过Scene中的first\_sample\_token，可以获得sample。Sample以链表形式存在

sample\_rec = nusc.get('sample', scene\_rec['first\_sample\_token'])

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| token | 标签 | str | 'e93e98b63d3b40209x' |
| Timestamp | 时间戳 | Int | 1531883530449377 |
| Prev | 上一个sample token | Str |  |
| Next | 下一个sample token | Str | '14d5adfe50bb444x' |
| Scene\_token | Sample属于的场景token | Str | '73030fb67d3c4 |
| Data | 数据 | Dict | ‘RADAR\_FRONT’=’bdddx’  ‘RADAR\_FRONT\_LEFT’=’1a  ‘RADAR\_FRONT\_RIGHT’=  ‘RADAR\_BACK\_LEFT’  ‘RADAR\_BACK\_RIGHT’  ‘LIDAR\_TOP’= '3388933b594’  ‘CAM\_FRONT’  ‘CAM\_FRONT\_RIGHT’  ‘CAM\_BACK\_RIGHT’  ‘CAM\_BACK’  ‘CAM\_BACK\_LEFT’  ‘CAM\_FRONT\_LEFT’  共十二个5 Radar 6 Cam 1 Lidar |
| anns | 标注 | List | ['173a50411564442ab195e132472fde71'  '5123ed5e450948ac8dc381772f2ae29a'  …] |

Sample\_data\_record

通过sample中data的token，可以获取传感器数据

## 激光传感器数据结构

sd\_rec = nusc.get('sample\_data', sample\_rec['data']['LIDAR\_TOP'])

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| token | 标签 | str | '3388933b594 |
| Sample\_token | 所属的sample token | str | 'e93e98b63d |
| Ego\_pos\_token | 自车位姿token | Str | 'e93e98b63d |
| Calibrated\_sensor\_token | 传感器标定参数token | Str | '7a0cd258d0 |
| Timestampe | 时间戳 | Str | 1531883530449377 |
| Fileformat | 传感器文件类型 | Str | 'pcd' |
| Is\_key\_frame | 是否是关键字 | bool | True |
| Height | 传感器高度 | Int | 0 |
| Width | 传感器宽度 | Int | 0 |
| Filename | 传感器文件名 |  | 'samples/LIDAR\_TOP/n015-2018-07-18-11-07-57+0800\_\_LIDAR\_TOP\_\_1531883530449377.pcd.bin' |
| prev | 上一个传感器token | Str |  |
| next | 下一个传感器token | Str | 'bc2cd87d110747 |
| Sensor\_modality | 传感器类型 | Str | 'lidar' |
| Channel | 通道名称 | Str | 'LIDAR\_TOP' |

毫米波数据

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| token | 标签 | str | 'bddd80ae33e |
| Sample\_token | 所属的sample token | str | 'e93e98b63d3 |
| Ego\_pos\_token | 自车位姿token | Str | 'bddd80ae33ec4 |
| Calibrated\_sensor\_token | 传感器标定参数token | Str | '7781065816974801af' |
| Timestampe | 时间戳 | Str | 1531883530440378 |
| Fileformat | 传感器文件类型 | Str | 'pcd' |
| Is\_key\_frame | 是否是关键字 | bool | True |
| Height | 传感器高度 | Int | 0 |
| Width | 传感器宽度 | Int | 0 |
| Filename | 传感器文件名 |  | 'samples/RADAR\_FRONT/n015-2018-07-18-11-07-57+0800\_\_RADAR\_FRONT\_\_1531883530440378.pcd' |
| prev | 上一个传感器token | Str |  |
| next | 下一个传感器token | Str | '90df03ad4710427aa' |
| Sensor\_modality | 传感器类型 | Str | 'radar' |
| Channel | 通道名称 | Str | 'RADAR\_FRONT' |

## 相机传感器数据结构

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| token | 标签 | str | '020d7b4f8581475 |
| Sample\_token | 所属的sample token | str | 'e93e98b63d3b402 |
| Ego\_pos\_token | 自车位姿token | Str | '020d7b4f858147558 |
| Calibrated\_sensor\_token | 传感器标定参数token | Str | '2e64b091b3b146a39 |
| Timestampe | 时间戳 | Str | 1531883530412470 |
| Fileformat | 传感器文件类型 | Str | 'jpg' |
| Is\_key\_frame | 是否是关键字 | bool | True |
| Height | 传感器高度 | Int | 900 |
| Width | 传感器宽度 | Int | 1600 |
| Filename | 传感器文件名 |  | 'samples/CAM\_FRONT/n015-2018-07-18-11-07-57+0800\_\_CAM\_FRONT\_\_1531883530412470.jpg' |
| prev | 上一个传感器token | Str |  |
| next | 下一个传感器token | Str | 'caa2bfad0b8a4a8090 |
| Sensor\_modality | 传感器类型 | Str | 'camera' |
| Channel | 通道名称 | Str | 'CAM\_FRONT' |

## 车体位姿Ego pose数据结构

每一个sensor都会包括一个对应的vehicle ego pose，

ego\_pose=nusc.get('ego\_pose', sd\_rec['ego\_pose\_token'])

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| token | 标签 | str | 3388933b59444c5d |
| Timestamp | 时间戳 | Int | 1531883530449377 |
| Rotation | 四元数旋转 | List[double] | [-0.7495886280607293, -0.0077695335695504636, 0.00829759813869316, -0.6618063711504101] |
| Translation | 平移 | List[double] | [1010.1328353833223, 610.8111652918716, 0.0] |

## Sensor calibration数据结构

cam\_calib=nusc.get('calibrated\_sensor', cam\_rec['calibrated\_sensor\_token'])

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| token | 标签 | str | '2e64b091b3b14 |
| Sensor\_token |  | str | '725903f5b62f5 |
| translation | 平移 | List[double] | [1.70079118954, 0.0159456324149, 1.51095763913] |
| Rotation | 四元数旋转 | List[double] | [0.4998015430569128, -0.5030316162024876, 0.4997798114386805, -0.49737083824542755] |
| Camera\_intrinsic | 相机内参 | List[List[double]] | [[1266.417203046554, 0.0, 816.2670197447984], [0.0, 1266.417203046554, 491.50706579294757], [0.0, 0.0, 1.0]] |

## Annotation标注数据结构

Annotation是针对一个sample里面标注的一个对象

my\_annotation\_token = sample\_rec['anns'][0]

my\_annotation\_metadata =  nusc.get('sample\_annotation', my\_annotation\_token)

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| 'token' | 标签 | str | 173a504115644 |
| sample\_token | 所属的sample token | str | e93e98b63d3b4020 |
| instance\_token |  | str | 5e2b6fd1fab |
| visibility\_token |  | str | 4 |
| attribute\_tokens |  | str | [] |
| translation | 位移 | List[double] | [994.031, 612.51, 0.728] |
| size | 尺寸 | List[double] | [0.3, 0.291, 0.734] |
| rotation | 旋转四元数 | List[double] | [-0.04208490861058176, 0.0, 0.0, 0.9991140377690821] |
| prev |  | str |  |
| next |  | str | 35034272eb1f4 |
| num\_lidar\_pts |  | Int | 2 |
| num\_radar\_pts |  | Int | 0 |
| category\_name |  | str | movable\_object.trafficcone |

## Instance数据结构

Instance是一个跨多sample，但是在一个scene里面的对象

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| 'token' | 标签 | str | 173a504115644 |
| sample\_token | 所属的sample token | str | e93e98b63d3b4020 |
| instance\_token |  | str | 5e2b6fd1fab |
| visibility\_token |  | str | 4 |
| attribute\_tokens |  | str | [] |
| translation | 位移 | List[double] | [994.031, 612.51, 0.728] |
| size | 尺寸 | List[double] | [0.3, 0.291, 0.734] |
| rotation | 旋转四元数 | List[double] | [-0.04208490861058176, 0.0, 0.0, 0.9991140377690821] |
| prev |  | str |  |
| next |  | str | 35034272eb1f4 |
| num\_lidar\_pts |  | Int | 2 |
| num\_radar\_pts |  | Int | 0 |
| category\_name |  | str | movable\_object.trafficcone |

# MMdetection3D数据转换后数据结构

从MMdetection3D的create\_data.py程序分析，该程序完成了对nuscene数据的转换

总共从1000个scene中，取了850个scene，标准是scene必须包含lidar数据，700个做train，150个做val

核心是通过

def \_fill\_trainval\_infos(nusc,

                         train\_scenes,

                         val\_scenes,

                         test=False,

                         max\_sweeps=10):

完成，针对每一个sample形成info数据包括。MMDetection3D主要是进行三维检测，故针对一个sample就可以了

总共28130个train sample，6019个val sample

Train\_info.pkl数据结构

里面的sample定义

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| lidar\_path | 雷达文件路径 | Str | ./data/nuscenes/samples/LIDAR\_TOP/n015-2018-07-18-11-07-57+0800\_\_LIDAR\_TOP\_\_1531883530449377.pcd.bin |
| token | Sample的token | Str | e93e98b63d3b40209056d129dc53ceee |
| sweeps | 之前最多10帧的雷达数据 | List | [] |
| cams | 相机信息 | dict | ‘CAM\_FRONT’: 后文  ‘CAM\_FRONT\_RIGHT‘:  'CAM\_FRONT\_LEFT'  'CAM\_BACK'  'CAM\_BACK\_LEFT'  'CAM\_BACK\_RIGHT' |
| lidar2ego\_translation | 激光相对于自车平移向量 | List[double] | [0.943713, 0.0, 1.84023] |
| lidar2ego\_rotation | 激光相对于自车旋转四元数 | List[double] | [0.7077955119163518, -0.006492242056004365, 0.010646214713995808, -0.7063073142877817] |
| ego2global\_translation | 自车全局平移向量 | List[double] | [1010.1328353833223, 610.8111652918716, 0.0] |
| ego2global\_rotation | 自车相对全局旋转四元数 | List[double] | [-0.7495886280607293, -0.0077695335695504636, 0.00829759813869316, -0.6618063711504101] |
| timestamp | 时间戳 | Int | 1531883530449377 |
| gt\_boxes | 真值三维框，每个框由一个7维向量表示，[0,2]代表位置，[3,5]代表尺度，[6]代表方向 | List[list[double]] | [[-1.61843454e+01 -1.17404151e+00 -1.24046699e+00 3.00000000e-01  2.91000000e-01 7.34000000e-01 1.36454530e+00]  [-1.54493912e+01 -4.28768163e+00 -1.30136452e+00 3.15000000e-01  3.38000000e-01 7.12000000e-01 1.25993027e+00] |
| gt\_names | 真值类型，来自cateroty转换 | List[str] | ['traffic\_cone' 'traffic\_cone' 'truck' 'car' 'pedestrian' 'car' 'truck'  'pedestrian' 'traffic\_cone' 'truck'] |
| gt\_velocity | 真值二维速度 | List[double] | [[-7.37269312e-03 2.72807041e-02]  [-2.31282614e-02 7.36653392e-02]  [ 5.79080632e-02 6.95126389e-02] |
| num\_lidar\_pts | 每个对象上的激光点数 | List[int] | [ 2 3 171 150 7 151 9 10 4 42] |
| num\_radar\_pts | 每个对象上的雷达点数 | List[int] | [0 0 7 2 0 3 3 0 0 6] |
| valid\_flag |  |  | [ True True True True True True True True True True] |

## Camera数据结构

Camera信息

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| data\_path | 图片路径 | Str | ./data/nuscenes/samples/CAM\_FRONT/n015-2018-07-18-11-07-57+0800\_\_CAM\_FRONT\_\_1531883530412470.jpg |
| type | 相机类型 | Str | CAM\_FRONT |
| sample data token |  | Str | 020d7b4f858147558106c504f7f31bef |
| sensor2ego\_translation | 相机相对车体位移 | List[double] | [0.0283260309358, 0.00345136761476, 1.57910346144] |
| sensor2ego\_rotation | 相机相对车体旋转四元数 | List[double] | [-0.7507230134456502, -0.007585124591152042, 0.008525945692580405, -0.6605185320776894] |
| ego2global\_translation | 自车全局位移 | List[double] | [1010.1256390280025, 610.760527131398, 0.0] |
| ego2global\_rotation | 自车全局旋转四元数 | List[double] | [-0.7507230134456502, -0.007585124591152042, 0.008525945692580405, -0.6605185320776894] |
| timestamp |  |  | 1531883530437525 |
| sensor2lidar\_rotation | 传感器相对于lidar的旋转矩阵 | List[list[double]] | [[-0.99991364, 0.01038427, -0.00805477],  [0.0081334 , 0.00755213, -0.9999384 ],  [-0.0103228 , -0.99991756, -0.00763594]] |
| sensor2lidar\_translation | 传感器相对于lidar的位移阵 | List[list[double]] | [0.00072265, 0.60818175, -0.31034774] |
| cam\_intrinsic | 相机内参 | List[list[double]] | [[1.26641720e+03, 0.00000000e+00, 8.16267020e+02],  [0.00000000e+00, 1.26641720e+03, 4.91507066e+02],  [0.00000000e+00, 0.00000000e+00, 1.00000000e+00]] |

2D annotation

通过nuscenes\_converter.export\_2d\_annotation，将保存的三维annotation信息转换成二维

Dete3d实际训练中所用到的数据

Mmcv/epoch\_based\_runner.py

def train(self, data\_loader, \*\*kwargs):

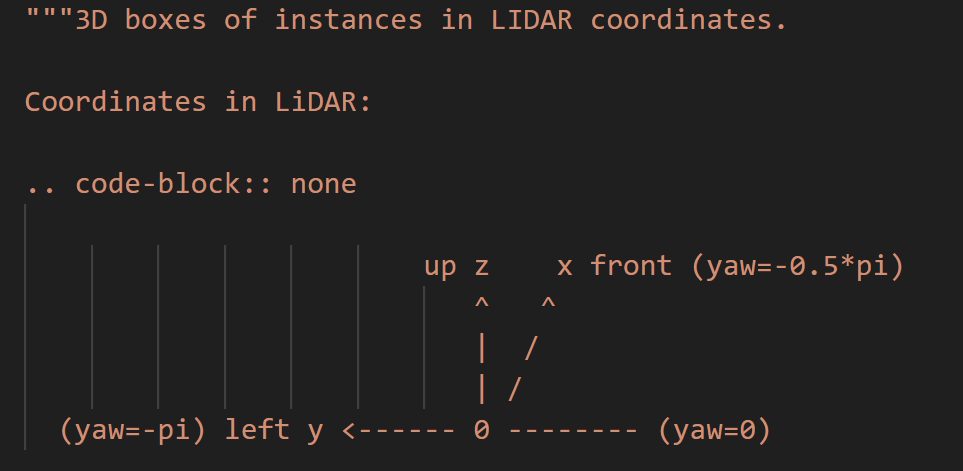
# MMDetection3d运行状态数据结构

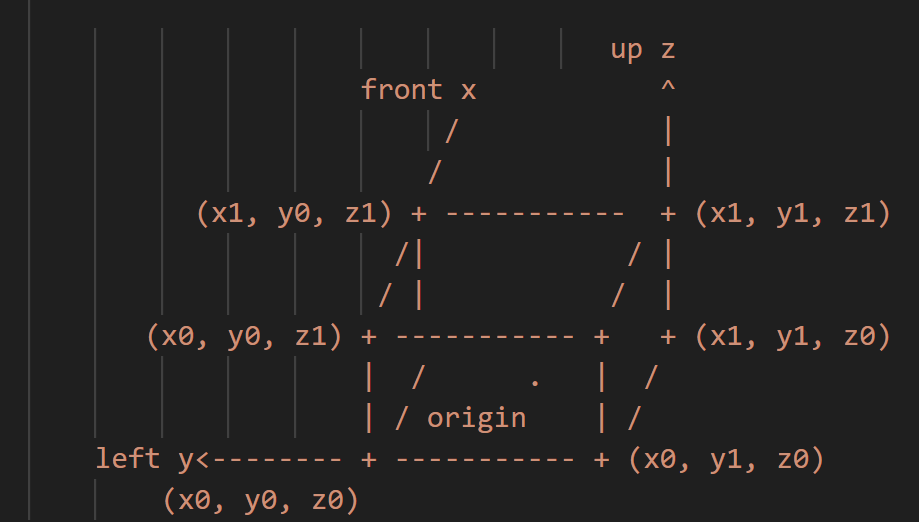
## batch\_data

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| data\_batch[‘gt\_bboxes\_3d’] | 3d对象外包框 | List[LIDARInstance3DBoxes] |  |
| data\_batch[‘gt\_lables\_3d’] | 3d对象分类 | List[LIDARInstance3DBoxes] |  |
| imgs | 图片数据 | List[tensor] |  |
| Img\_metas | 图片相关属性 | List[img\_meta] |  |
|  |  |  |  |

## Img\_metas

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 举例 |
| filename | 图片文件名 | List[str] | CAM\_FRONT.jpg  CAM\_BACK.jpg |
| Ori\_shape | 图片原始形状 | Tuple | (900,1600,3,6) |
| Img\_shape | 处理后图片形状 | List[tuple] | [(928,1600,3),(928,1600,3)] |
| Lidar2img | 激光到图片外参矩阵 | List[list] | 6x4x4 |
|  |  |  |  |
| Sample\_idx | 样本标志 |  | '26e4027ee468436e8 |
| Pts\_filename | 点云文件名 |  | LIDAR\_TOP.pcd.bin |





## LIDARInstance3DBoxes

gt\_bboxes\_3d属于LIDARInstance3DBoxes，是真值中的boudingbox对象，以下以一个包含31个对象的LIDARInstance3DBoxes为例解释

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 维度举例 |
| Bev | 2D BEV box of each box with rotation | Tensor | 31x5 |
| nearest\_bev | 2D BEV box of each box without rotation | Tensor | 31x5 |
| Bottom center | center of each box | Tensor | 31x3 |
| Bottom\_height |  | Tensor | 31 |
| center | center of each box | Tensor | 31x3 |
| Corners |  | Tensor | 31x8x3 |
| Height | height of each box | Tensor | 31 |
| Top\_height | top height of each box | Tensor | 31 |
| Volume | volume of each box | Tensor | 31 |
| Yaw |  | Tensor | 31 |

gt\_labels\_3d 是真值中的标签数据，以下以一个对象为例解释

|  |  |  |  |
| --- | --- | --- | --- |
| 成员 | 含义 | 类型 | 维度举例 |
| gt\_labels\_3d | Bbox对应的类 | Tensor | 31 |

tensor([8, 0, 2, 5, 0, 5, 0, 0, 1, 0, 0, 0, 0, 8, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 5, 0, 0, 0, 0, 1], device='cuda:0')

Detr3d输入文件

Detr3d.py

    def forward\_train(self,

                      points=None,

                      img\_metas=None, :包含了六张图片的annotation信息

                      gt\_bboxes\_3d=None,

                      gt\_labels\_3d=None,

                      gt\_labels=None,

                      gt\_bboxes=None,

                      img=None,

                      proposals=None,

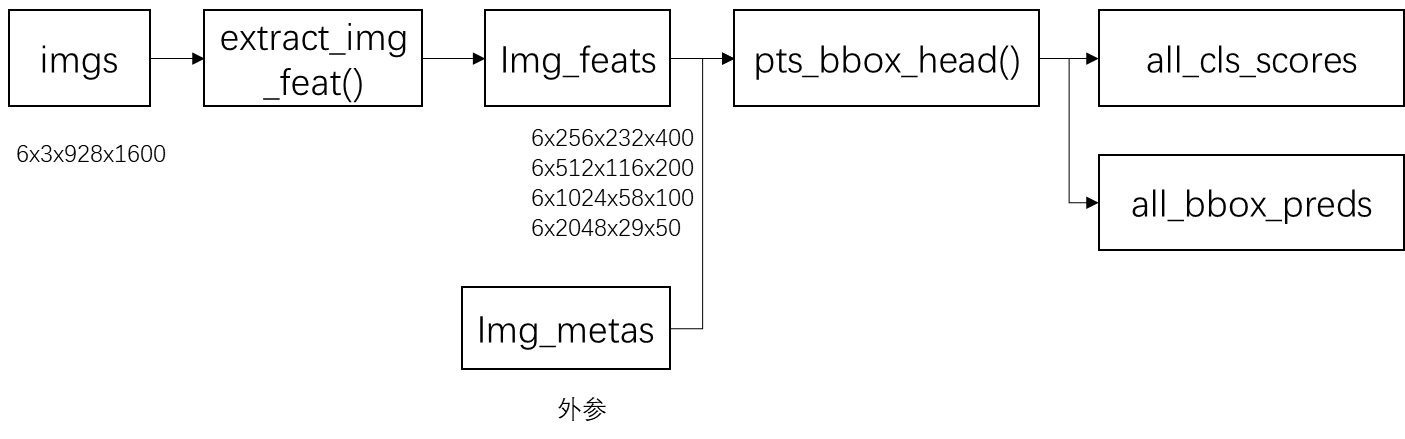
                      gt\_bboxes\_ignore=None,

                      img\_depth=None,

                      img\_mask=None):

# Detr3D算法结构与数据输入输出

## 模型运行数据流



Img\_metas内容参考上文数据结构定义

## 模型训练数据流

