Data3dPreprocessing

"""Points / Image pre-processor for point clouds /

vision-only / multi- modality 3D detection tasks.

```
It provides the data pre-processing as follows

- Collate and move image and point cloud data to

- 1) For image data:

  - Pad images in inputs to the maximum size of
    ``pad_value``. The padding size can be divis
    ``pad_size_divisor``.
  - Stack images in inputs to batch_imgs.
  - Convert images in inputs from bgr to rgb if
    (3, H, W).
  - Normalize images in inputs with defined std
  - Do batch augmentations during training.

- 2) For point cloud data:

  - If no voxelization, directly return list of
  - If voxelization is applied, voxelize point c
    ``voxel_type`` and obtain ``voxels``.
```

data(dict, two keys)

- data_samples
- inputs

inputs (dict, one key)

- points list of tensor (exactly the loaded raw pcd files)

voxelize (function) (pcd data need this process.)

args:

- points (List[Tensor]): Point cloud in one data batch.
- data_samples: the annotation data of every samples. Add voxel-wise annotation for segmentation.

returns:

Dict[str, Tensor]: Voxelization information.

- voxels (Tensor): Features of voxels, shape is

MxNxC for hard voxelization, NxC for dynamic voxelization.

- coors (Tensor): Coordinates of voxels, shape is Nx(1+NDim), where 1 represents the batch index.
- num_points (Tensor, optional): Number of points in each voxel.
- voxel_centers (Tensor, optional): Centers of voxels.

max_voxels 30000

voxel size: [0.25, 0.25, 8]

coor_range: [-50, -50, -5, 50, 50, 3]

voxels:

- shape: (max_voxels, max_points, points.size[1])

eg: (30000, 64, 4) --将点云中的点放入到这样一个tensor里.

coors:

*shape:(max_voxels，3)，dtype：int

eg：(30000，3)

每个voxel有一个空间坐标.

num_points_per_voxel

(30000)