1. **crop\_img.py**

将卫星图片裁切，生成每个基站-接收机链路对应的样本图片，选择指定的城市（宁波、杭州、温州、马来）和子网格（小区），以其中每个链路的基站和接收机作为图片的左右端点，裁切正方形区域的卫星图片作为输入。

具体步骤包括：

1. 首先选择从基站至某测试点位置的中心点作为旋转和裁切中心；
2. 然后旋转图中BS——>UE的方向为水平方向；
3. 最后以基站和测试点之间距离的1.2倍为窗口进行裁切。
4. **coordinate\_conv.py**

提取各城市的二进制电子地图并转化为npy文件。其中包括**坐标转化**和**转换二进制坐标地图**的函数，不能独立运行；坐标转换包括到gps经纬度到火星坐标系以及到瓦片坐标系，再由瓦片转换到像素值进行裁剪（每个瓦片是固定的256\*256像素）；将clutter文件中的二进制地图信息转化为一个矩阵存到npy文件中。

1. **Trajectory\_DigitalMap.py**

可视化不同路测路线的接收机轨迹以及基站等信息在电子地图上的分布，效果如图1所示（红色点是基站位置）。

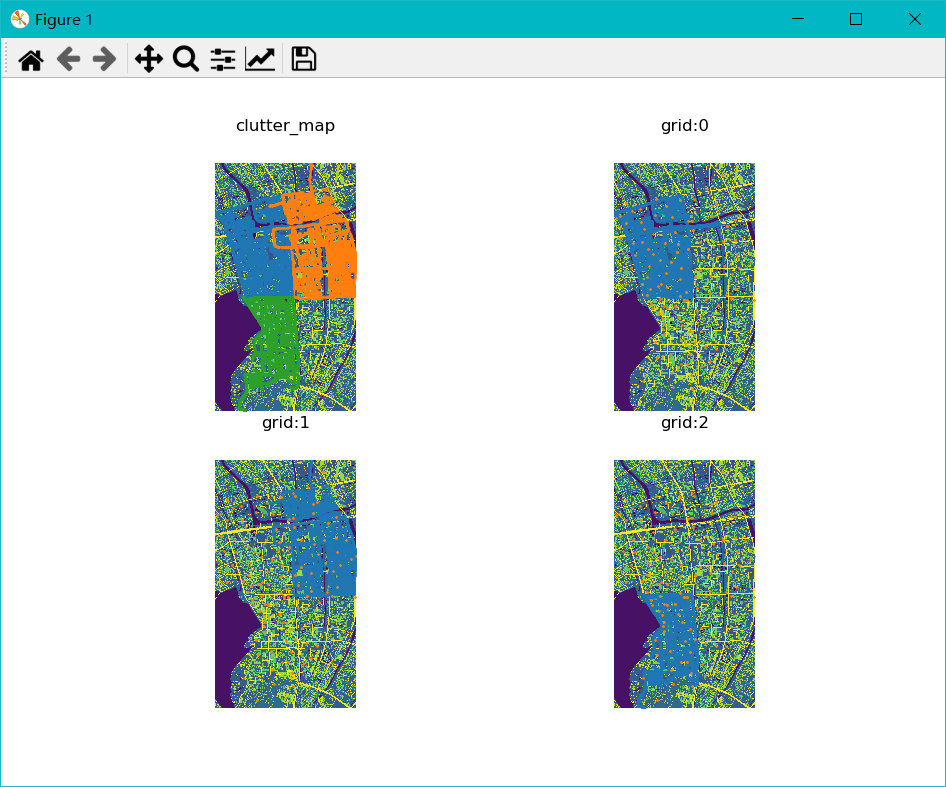


图 1接收机轨迹（电子地图）

1. **Trajectory\_SatelliteMap.py**

可视化某个城市网格中的接收机位置UE在卫星图上的分布，如图2所示。

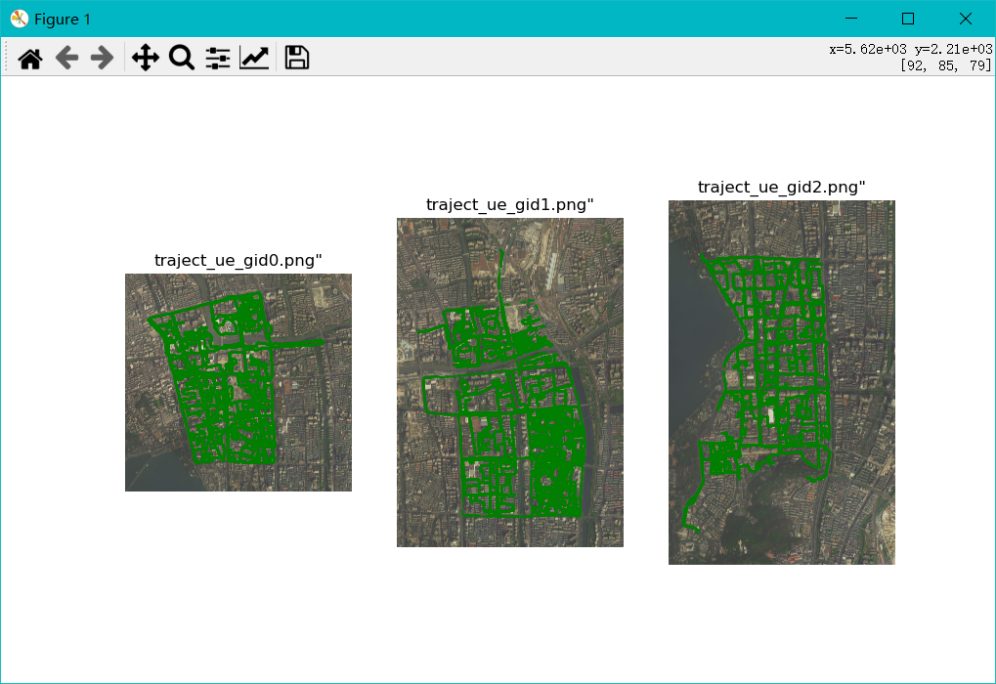


图 2 接收机轨迹（卫星地图）

1. **model.py**

模型文件

1. **Training\_nopic.py**

使用小区编号、距离和方位角进行训练和测试的流程。

1. **Training\_wpic.py**

使用卫星图片、小区编号、距离和方位角进行训练和测试的流程。

1. **map\_loader.py**

根据经纬度、分辨精度下载指定地区的卫星地图。

1. **tools.py**

辅助函数包