附件1：任务区域地形数据及文件格式说明

**1、地形数据下载：**

下载地址： <http://120.24.70.139:9090/temp/map.7z>

下载地形数据MD5码：31bcaad2d3e25a2fd29e7adcfe858d1a

MD5码用于校验下载的文件是否完整。

**2、地形数据说明**

任务区域是在水平面上投影范围为62.5km×62.5km的山地区域，不考虑地球曲率的影响，地表无植被。结合测绘精度，将该区域划分为12500×12500个栅格，每个栅格是边长为5m的正方形小块。

**栅格**是本题用于描述空间位置和地形特征的基本对象，其基本属性包括坐标、高程、坡度、坡向等。

* **坐标**：以区域西南角栅格中心点为原点，正东为*x*轴正向，正北为*y*轴正向，单位1长度5m，建立平面直角坐标系。以每个栅格中心点坐标代表该栅格，如，区域西南角栅格的坐标为(0, 0)，该栅格可简称为栅格(0, 0)；东南角栅格的坐标为(12499, 0)，该栅格可简称为栅格(12499, 0)；栅格(0, 12499)为区域西北角栅格，栅格(12499, 12499)为东北角栅格。

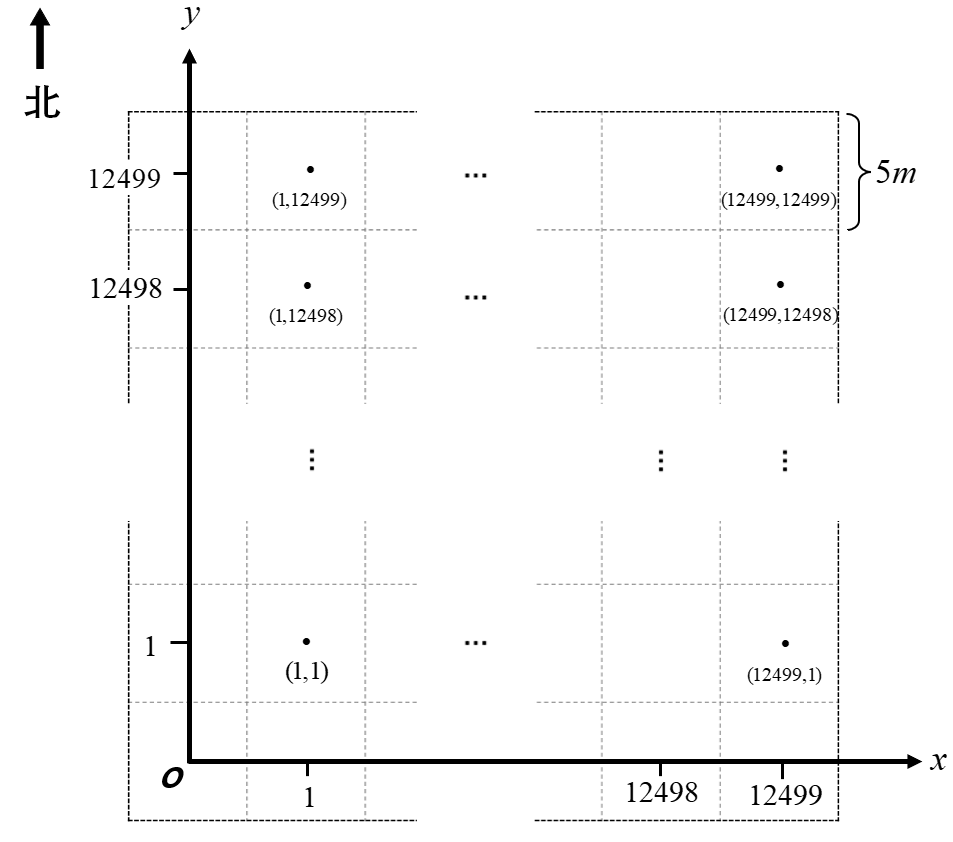


图1 栅格坐标示意图

* **高程**：经测绘得到每个栅格中心点的高程值（单位：m，保留2位小数），将栅格中心点的高程值简称为栅格的高程。如，栅格(12499, 0)的高程为5038.84，栅格(12499, 12499)的高程为4954.06。任务区域的地形数据保存在TIFF（Tagged Image File Format）格式文件map.tif中，即，该文件记录了整个任务区域所有栅格中心点的高程数据。TIFF格式是用于存储高分辨率图像数据的通用文件格式，常见高级编程语言均支持对该格式文件的读写。使用Python语言或Matlab语言读取TIFF文件的示例代码附后。为便于校验读取结果的正确性，任务区域中部分栅格的高程数据见表1。
* **坡度**与**坡向**：由于实际的地表曲面是未知的，仅能测得栅格中心点处的高程值，为简化问题，本题将每个栅格所对应的一小片地表曲面视为平面，这一平面小块称为该栅格的坡面，可简称为栅格坡面。栅格的坡面在水平面上的投影为该栅格，且近似后栅格中心点对应的高程保持不变。不考虑相邻栅格的坡面在栅格边界处的连续性。栅格坡面所在平面与水平面的二面角角度值为该栅格的坡度。栅格坡面的法向量在水平面*xoy*的投影与正北方向的夹角为坡向，以正北方向为0度，顺时针为正方向。栅格的坡度、坡向和栅格坡面的法向量可由该栅格及其相邻栅格的高程近似计算[1-2]，近似计算方法如下。

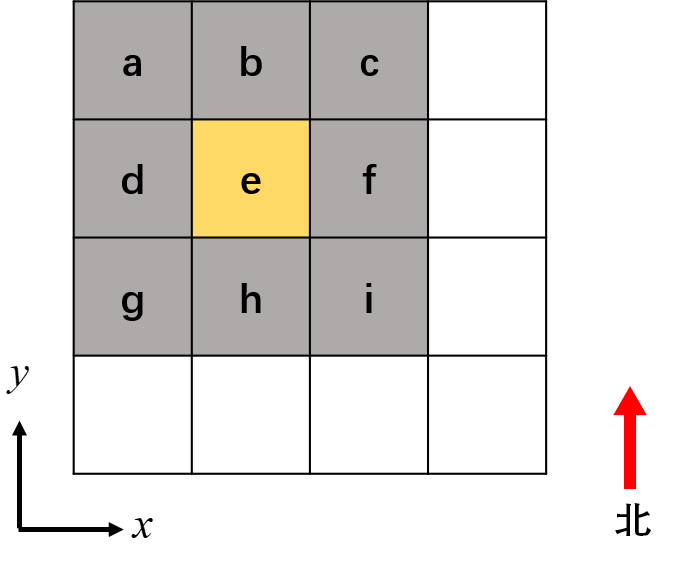


图2 栅格及其3×3邻域内相邻栅格示意图

如图2所示，灰色栅格为中心黄色栅格的8个相邻栅格，黄色栅格的坡度的近似计算公式如下：



其中，坡度用角度来表示，为栅格的高程，、是高程变化率。

具体来说，

 （1）

其中，为栅格边长（本题中为5m），分别为栅格的高程。

由于山地地形复杂多变，边长为5m的栅格容易忽略栅格内的路面起伏变化情况（1个栅格内路面可能经历多次上下起伏，高程变化无法在计算中体现），为了无人车能更好地应对山地运行潜在的高程变化风险，本题在计算高程变化率、时，增加1个高程变化因子*k*，修正后的计算公式（1）如下：

 （2）

高程变化因子与所选区域地貌特征相关，为了便于计算，本题约定*k*=5。

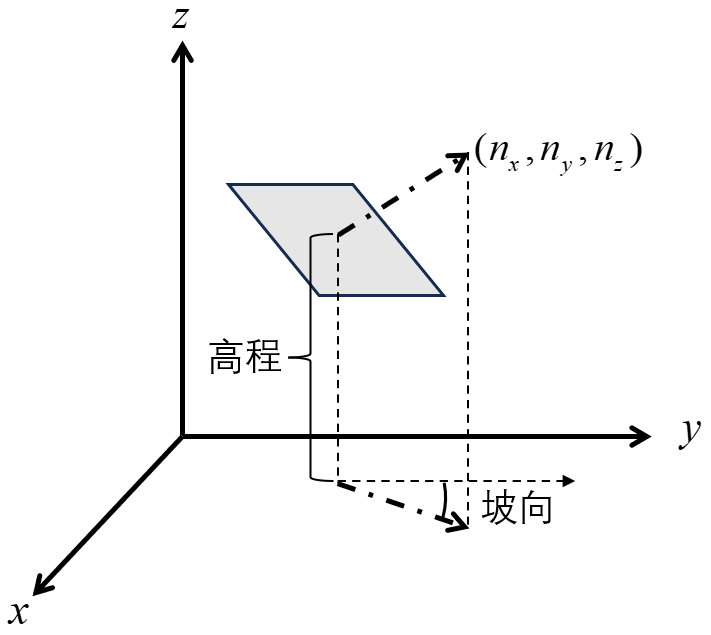


图3坡向、栅格坡面法向量关系示意图

栅格坡面法向量与坡度、坡向的关系如图3所示。黄色栅格的坡向的近似计算公式如下：



其中，坡向用角度来表示，、为公式(1)所示。

当，时，*A*=90°；当，时，*A*=180°；当，时，*A*=270°；当，时，*A*=0°或360°，特别是当，时，坡面为水平面。

综上，根据坡度与坡向可以得到该黄色栅格坡面的法向量：



**注**：由于任务区域边界上的栅格没有3×3邻域，因此不计算这些栅格的坡度和坡向；规定坡面是水平面时的法向量为：(0, 0, 1)。

* 部分栅格的属性

表1 部分栅格的属性

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 栅格*x*坐标 | 栅格*y*坐标 | 栅格高程(m) | 栅格坡度 | 栅格坡向 |
| 0 | 0 | 5119.02 | / | / |
| 1 | 0 | 5119.02 | / | / |
| 0 | 1 | 5119.02 | / | / |
| 1 | 1 | 5119.02 | 0° | 水平面 |
| 12498 | 0 | 5038.84 | / | / |
| 12499 | 0 | 5038.84 | / | / |
| 12498 | 1 | 5038.84 | 0.2798° | 270° |
| 12499 | 1 | 5038.84 | / | / |
| 0 | 12498 | 5658.34 | / | / |
| 1 | 12498 | 5658.34 | 0.2936° | 180° |
| 0 | 12499 | 5658.35 | / | / |
| 1 | 12499 | 5658.35 | / | / |
| 12498 | 12498 | 4954.06 | 0° | 水平面 |
| 12499 | 12498 | 4954.06 | / | / |
| 12498 | 12499 | 4954.06 | / | / |
| 12499 | 12499 | 4954.06 | / | / |

**3、读取地形数据（TIFF文件）的示例代码**

|  |
| --- |
| （1）使用Python语言编程从map.tif文件中读取数据时，会得到一个大小为12500×12500的二维数组，该数组的元素值与任务区域中的12500×12500个栅格的高程一一对应。通过数组的索引可以访问数组的元素，从而得到对应的高程。数组索引由行索引和列索引构成，其中行索引表示数组元素所在行的序号、列索引表示数组元素所在列的序号。设某一栅格的坐标为，与其对应的数组行索引、列索引为，若索引均以(0, 0)开始计数，对应关系:    Python（方法1）  ------------------------------------------------------------------------  import rasterio  file\_path = "map.tif" #指定文件路径  # 打开 TIF 文件  def read\_map(file\_path): with rasterio.open(file\_path) as src:  # 读取地图的数据  map= src.read(1)  return map  ```  请确保已经安装了rasterio库，可以使用pip指令进行安装：pip install rasterio  Python（方法2）  ------------------------------------------------------------------------  from PIL import Image  import matplotlib.pyplot as plt  file\_path ="map.tif" #指定文件路径 map = Image.open(file\_path) # 读取TIF文件  ------------------------------------------------------------------------  使用Python根据栅格坐标读取高程示例代码  ------------------------------------------------------------------------  # 根据栅格坐标读出对应的高程  import numpy as np  def get\_elevation(map, x, y):  [r, c] = (np.array([12499 - y, x])).astype(np.int16) print(f"栅格坐标({x},{y})的高程为：", map[r, c])  通过调用上述函数，可以利用栅格坐标获取对应的高程，例：  get\_elevation(map, 100, 100)  输出为：栅格坐标(100,100)的高程为： 5118.7 |

|  |
| --- |
| （2） 使用Matlab语言编程从map.tif文件中读取数据时，会得到一个大小为12500×12500的二维数组，该数组的元素值与任务区域中的12500×12500个栅格的高程一一对应。通过数组的索引可以访问数组的元素，从而得到对应的高程。数组索引由行索引和列索引构成，其中行索引表示数组元素所在行的序号、列索引表示数组元素所在列的序号。设某一栅格的坐标为，与其对应的数组行索引、列索引为，若索引均以(1, 1)开始计数，对应关系:    Matlab  ------------------------------------------------------------------------  % 指定文件路径  filePath = 'map.tif';  % 读取TIF文件  map = imread(filePath);  % 创建图形窗口并显示图像  figure;  imshow(imageData);  title('显示的TIF图像');  xlabel('X坐标');  ylabel('Y坐标');  colorbar; % 可选，根据需要添加  -------------------------------------------------------------------------  使用Matlab根据栅格坐标读取高程示例代码  ------------------------------------------------------------------------  function elevation = get\_elevation(map, x, y)  % MATLAB中的索引是从1开始的，所以需要调整坐标  % 同时，MATLAB的数组索引是行列顺序的，即先列后行  r = 12500 - y; % 假设map是12500x12500的矩阵，这里y是从上到下计数  c = x; % x是从左到右计数  % 检查索引是否在map的范围内内  if r > 0 && r <= size(map, 1) && c > 0 && c <= size(map, 2)  elevation = map(r, c);  fprintf('栅格坐标(%d,%d)的高程为：%f\n', x, y, elevation);  else  error('坐标(%d,%d)超出了map的范围。', x, y);  end  end |

**4、参考文献**

[1]刘学军.基于规则格网数字高程模型解译算法误差分析与评价[D].武汉大学, 博士学位论文, 2002.

[2] 周启鸣，刘学军，数字地形分析[M]，北京：科学出版社，2006.