

代码内容审核建议 8 (2.2.3&2.3.1&2.3.2&2.4.1)

2.2.3 基于 NDVI 指数解译影像与建立采样工具

1. 流程表示方法需要优化

出现箭头这样的方式还是不妥的，建议此处采用一张图来简要表示流程，然后在图下面进行详细叙述，不要出现箭头

2.2.3 基于NDVI指数解译影像与建立采样工具

如果只是分析城市的绿地、裸地和水体，不涉及到更精细的土地覆盖分类，例如灌丛、草地、裸地、居民地、园地、耕地、河流、湖泊等等，可以利用Landsat系列遥感影像通过NDVI、NDWI和NDBI等手段提取绿地（可进一步细分耕地和林地）、裸地和水体等。基于NDVI的遥感影像解译，首先读取能够反映不同季节绿地情况的Landsat不同季节影像，根据研究的目的范围裁切，裁切的边界可以在QGIS中完成->然后计算不同季节的NDVI;->再通过使用交互的plotly图表，分析NDVI取值范围，判断不同土地覆盖阈值范围，解译影像;->如果要判断解译的精度，需要给出采样，随机提取点的真实土地覆盖类型，这个过程是一个需要手工操作的过程。python内嵌库tkinter的图形用户界面(Graphical User Interface,GUI)能够方便帮助我们快速的建立交互操作平台，完成采样工作->最后计算混淆矩阵和百分比精度，判断解译的精度。

2. 错别字 (thintet->thinter)

现。线性变换用`np.matmul()`计算，即两个矩阵之积。最好将计算的采样点坐标根据给定的文件位置保存。用于后续精度分析。

注意，tkinter编写的GUI不能在Jupyter (Lab) 下运行。需要在spyder等解释器下打开运行。可以新建.py文件，将下述代码复制于该文件，再运行。如果自行编写于tkinter开发的GUI工具，也需要在spyder等解释器下编写调试。完成的代码文件名为'image_pixel_sampling_zoom.py'。

2.3.1 编程微积分

1. 求导公式有误 【2. 导数值为 p, 7. 分子应该是 g(x)】

- 求导的基本公式

1. $p' = 0$ (p为常数);
2. $(px)' = 0$ (p为常数);
3. $(af(x))' = af'(x)$ # 常系数微分
4. $\{f(x) + g(x)\}' = f'(x) + g'(x)$; # 和的微分
5. $(x^n)' = nx^{n-1}$; # 幂函数的导数
6. $\{f(x)g(x)\}' = f(x)'g(x) + f(x)g(x)'$; # 积的微分
7. $\left\{\frac{f(x)}{g(x)}\right\}' = \frac{g'(x)f(x) - g(x)f'(x)}{f(x)^2}$ # 商的微分
8. $\{g(f(x))\}' = g'(f(x))f'(x)$ # 复合函数的微分
9. $\{f^{-1}(x)\}' = \frac{1}{f'(x)}$ # 反函数的微分方程

2. 错别字 【合成为一→合称为, summation→summation】

2.3.1.2 积分(Integrate)

积分是导数的逆运算 (针对计算方式而言), 利用积分可以求出变化的规律和不规则图形的面积。积分和导数通常配套使用 合成为微积分。积分通常分为定积分和不定积分两种。对于定积分, 给定一个正实值函数 $f(x)$, $f(x)$ 在一个实数区间 $[a, b]$ 上的定积分为 $\int_a^b f(x)dx$, 可以在数值上理解为在 oxy 坐标平面上, 由曲线 $(x, f(x))$ ($x \in [a, b]$), 直线 $x = a, x = b$ 以及 x 轴围成的曲边梯形的面积值 (一种确定的实数值)。 $f(x)$ 的不定积分 (或原函数) 是指任何满足导数是函数 $f(x)$ 的函数 $F(x)$ 。一个函数 $f(x)$ 的不定积分不是唯一的: 只要 $F(x)$ 是 $f(x)$ 的不定积分, 那么与之相差一个常数的函数 $F(x) + C$ 也是 f 的不定积分。 $\int_a^b f(x)dx$ 读作求函数 $f(x)$ 关于 x 的积分, $\int_a^b f(x)dy$ 读作求函数 $f(x)$ 关于 y 的积分, 因为积分是导数的逆运算, 因此可以理解为 '关于 x 求导得到 $f(x)$ 的原函数即为积分'。因此, 对于表述 '计算求导后得 $f(x)$ 的函数', '求 $f(x)$ 的不定积分', '求 $f(x)$ 的原函数', 这三种表达方式意思相同。

- 不定积分、定积分和面积

$\int f(x)dx$ 实际上表示将 $f(x) \times dx$ 进行 \int (积分), 而 \int 是 'summation (合计)' 的开头字母的变形, 表示对 $f(x) \times dx$ 的合计之意。 $f(x)$ 是 '与 x 对应的 y 轴坐标', dx 表示延 x 轴的最小增量。因此 $f(x) \times dx$ 就是变化横轴增量 dx 下矩形的面积。当对所有位于区间 $[a, b]$ 下变化增量为 dx 的矩形面积求积分 (合计) 后 (宽度极限小的长方形的集合), 即为区间为 $[a, b]$ 的横轴与曲线围合的面积。

3. 公式书写, 缺少括号

- 区分求积法

对于函数 $f(x)$, 给定区间 $[a, b]$, 假设进行 n 次分割, 长方形从左向右依次为 $x_1, x_2, x_3, \dots, x_k, \dots, x_n$, 单个矩形的面积为 $\frac{b-a}{n} \times f(x_k)$, 将全部的长方形面积加起来为 $S_{a-b} = \frac{b-a}{n} \times [f(x_1) + \frac{b-a}{n} \times f(x_2) + \dots + \frac{b-a}{n} \times f(x_n)] = \frac{b-a}{n} \{f(x_1) + f(x_2) + \dots + f(x_n)\} = \lim_{n \rightarrow \infty} \frac{b-a}{n} \{f(x_1) + f(x_2) + \dots + f(x_n)\} = \lim_{n \rightarrow \infty} \frac{b-a}{n} \sum_{k=1}^n f(x_k)$ (或 $\lim_{n \rightarrow \infty} \frac{b-a}{n} \sum_{k=0}^{n-1} f(x_k)$)。

下述代码在使用区分求积法时, 给定的函数为 $f'(x) = x^2$ (是函数 $f(x) = \frac{x^3}{3}$ 的导函数), 已知区间为 $[a, b]$, 依据上述公式则有 $S_{a-b} = \lim_{n \rightarrow \infty} \frac{b-a}{n} \sum_{k=0}^{n-1} f(x_k) = \lim_{n \rightarrow \infty} \frac{b-a}{n} \sum_{k=0}^{n-1} (a + k \times \frac{b-a}{n})^2$ 。在使用极限计算求和公式时, 需要使用 `doit()` 方法计算不被默认计算的对象 (极限、积分、求和及乘积等), 否则不能计算极限。

定积分求给定区间的面积, 直接使用 Sympy 提供的 `integrate` 方法, 给定区间计算结果约为 0.29, 与区分求积法计算结果相同。

4. 公式编写有误 （公式放在"\$...\$"中间）

上述公式中,

$f(0)$ <-----0次的常数项, 即 $a_0 = f(0)$ ----- (A)

$f'(0)x$ <-----1次项, 即 $a_1 = f'(0)$ ----- (B)

$\frac{1}{2!}f''(0)x^2$ <-----2次项, 即 $a_2 = \frac{1}{2!}f''(0)$ ----- (C)

$\frac{1}{3!}f'''(0)x^3$ <-----3次项, 即 $a_3 = \frac{1}{3!}f'''(0)$ ----- (D)

泰勒展开, 不一定非要从 $x = 0$ 的地方开始, 也可以从 $(x_0, f(x_0))$ 处开始, 此时只需要将0替换为 x_0 , 展开方法同上, 得:

$$f(x) = f(x_0) + \frac{1}{1!}f'(x_0)(x - x_0) + \frac{1}{2!}f''(x_0)(x - x_0)^2 + \frac{1}{3!}f'''(x_0)(x - x_0)^3 + \dots + \frac{1}{n!}f^{(n)}(x_0)(x - x_0)^n + \dots$$

• 误差项

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} (x - x_0)^{n+1} \quad (\text{推导过程略})$$

5. 公式符号没有解释含义意义

• 误差项

$$R_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} (x - x_0)^{n+1} \quad (\text{推导过程略})$$

2.3.2 卷积与 SIR 空间传播模型

1. k 的值的意义作用没有介绍，公式缺少不完整（绿色框内容的值-1）

We also assume that a fixed fraction k of the infected group will recover during any given day.

k 用来表示被感染的人群在给定任意一段时间内恢复的一个固定的比例

恢复人群微分方程：根据上述假设，可以得到， $\frac{dr}{dt} = ki(t)$ ，其中 k 是“受感人群”到“恢复人群的转换比例”，同样应用上述假设，如果 $k = 0.2$ ，则 $\Delta R = 0.2 * 3 = 0.6$ ，最终， $S = 1.6$ ， $I = 4.8$ ， $R = 3.6$ 。

受感人群微分方程：因为 $s(t) + i(t) + r(t) = 1$ ，可以得到 $\frac{ds}{dt} + \frac{di}{dt} + \frac{dr}{dt} = 0$ ，因此 $\frac{di}{dt} = bs(t)i(t) - ki(t)$ 。

最终的SIR传播模型为：

$$\begin{cases} \frac{ds}{dt} = -bs(t)i(t) \\ \frac{di}{dt} = bs(t)i(t) - ki(t) \\ \frac{dr}{dt} = ki(t) \end{cases}$$

对于SIR模型的解释参考来自于MAA(Mathematical Association of America), The SIR Model for Spread of Disease - The Differential Equation Model<作者: David Smith and Lang Moore>; 代码参考The SIR epidemic model

2. 字母要前后文对应 (β, γ 应该替换为 k 和 b)

同时代码中的 `gama`，`beta` 也要修改

2) SIR空间传播模型

SIR传播模型是给出了总体人口数 N ，以及 S, I, R 的初始值，和 β, γ 转换系数，计算 S, I, R 人口数量的变化。SIR传播模型不具有空间属性，因此引入了卷积扩散，可以实现一个源向四周扩散的变化过程，并且始终以源的空间位置强度最大，并向四周逐步减弱。这样通过结合SIR模型和卷积扩散，可以实现SIR的空间传播模型。

```
import util_misc
s_t=util_misc.start_time()
#参数配置
start_pt=[418, 640] #[3724, 3415]
beta=0.3
gamma=0.1
dispersion_rates=[0, 0.07, 0.03] #S层卷积扩散为0, I层卷积扩散为0.07, R层卷积扩散为0.03
dt=1.0
hours_per_second=30*24 #7*24
duration=12 #12
fps=15 #15
SIR_gif_savePath="./imgs/SIR_sp.gif"

SIR_sp=SIR_spatialPropagating(classi_array=classi_array, cost_mapping=cost_mapping, start_pt=start_pt, beta=beta, gamma=gamma, dispersion_rates=dispersion_rates)
SIR_sp.execute()
util_misc.duration(s_t)
```

start time: 2022-01-06 18:55:30.966194

```

class SIR_spatialPropagating:
    """
    funciton - SIR的空间传播模型

    Paras:
        classi_array - 分类数据 (.tif, 或者其它图像类型), 或者其它可用于成本计算的数据类型
        cost_mapping - 分类数据对应的成本值映射字典
        beta - beta值, 确定S-->I的转换率
        gamma - gamma值, 确定I-->R的转换率
        dispersion_rates - SIR三层栅格各自对应的卷积扩散率
        dt - 时间更新速度
        hours_per_second - 扩散时间长度/终止值(条件)
        duration - moviepy参数配置, 持续时长
        fps - moviepy参数配置, 每秒帧数
        SIR_gif_savePath - SIR空间传播计算结果.gif文件保存路径
    """
    def __init__(self, classi_array, cost_mapping, start_pt=[10, 10], beta=0.3, gamma=0.1, dispersion_rates=[0, 0.07, 0.07]):
        from sklearn.preprocessing import MinMaxScaler

        # 将分类栅格, 按照成本映射字典, 转换为成本栅格(配置空间阻力)
        for idx, (identity, cost_value) in enumerate(cost_mapping.items()):
            classi_array[classi_array==identity]=cost_value

        self.mms=MinMaxScaler()
        normalize_costArray=self.mms.fit_transform(classi_array) # 标准化成本栅格

        # 配置SIR模型初始值, 将S设置为空间阻力值
        SIR=np.zeros((3, normalize_costArray.shape[0], normalize_costArray.shape[1]), dtype=float)
        SIR[0]=normalize_costArray

        # 配置SIR模型中I的初始值。1, 可以从设置的1个或多个点开始; 2, 可以将森林部分直接设置为I有值, 而其它部分保持0。
        # start_pt=int(0.7*normalize_costArray.shape[0]), int(0.2*normalize_costArray.shape[1]) # 根据行列拾取点位置
        # print("起始点:", start_pt)
        start_pt=start_pt
        SIR[1, start_pt[0], start_pt[1]]=0.8 # 配置起始点位置值

        # 配置转换系数, 以及卷积核
        self.beta=beta #  $\beta$ 值
        self.gamma=gamma #  $\gamma$ 值
        self.dispersion_rates=dispersion_rates # 扩散系数
        dispersion_kernel=np.ones((3, 5, 5), dtype=float)

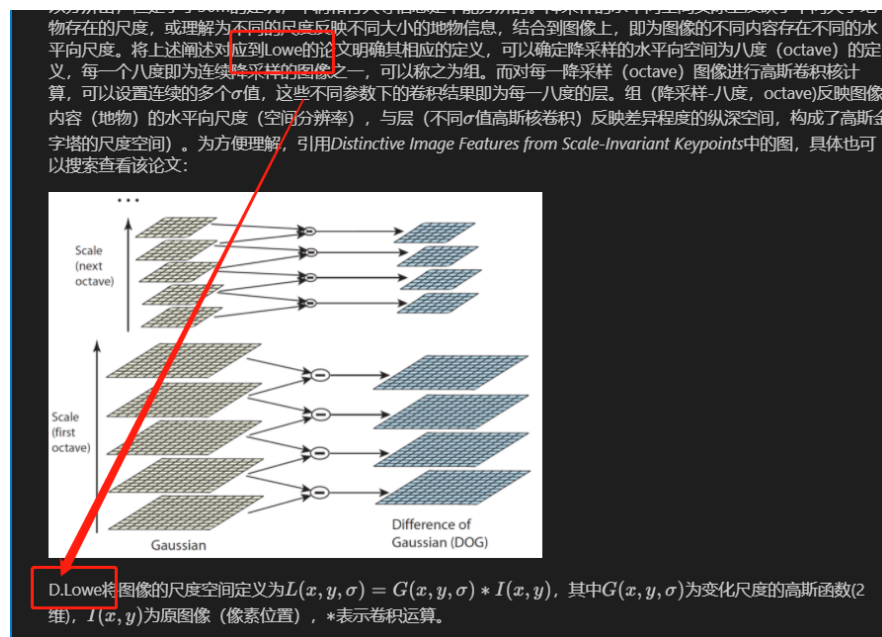
```

2.4.1 图像特征提取与动态街景视觉感知

1. 描述不清，略费解

• B 尺度空间 (scale space) 与高斯差分金字塔

从高斯分布 (函数) 到高斯模糊可知, 在固定卷积核大小时 (可参看卷积部分), 不同的 σ 值, 因为产生权重值的分布陡峭程度不同, 图像的模糊程度会对应变化。更好理解为基于变化权重值的图像拉伸, 从图像整体上来讲就是各个像素值与各周边像素值的差异程度。而 σ 值的变化带来差异程度分布即加权均值分布的差异, 这个差异变化是固定卷积核大小下尺度空间的纵深向空间。当 σ 设置值较大时, 差异程度分布趋向于均值, 反映图像的概貌特征; 设置较小时, 差异程度分布明显, 强化了目标像素与周边像素的差异性, 对应于图像的细节特征。如果对图像连续降采样, 每一个降采样都是前一图像的 $1/4$, 即长宽分别缩小一倍, 这一连续降采样获得的多个图像就是尺度空间的水平向空间。降采样类似于遥感影像的空间分辨率, 如果分辨率为 1m , 即遥感影像的每一单元 (cell) 大小为 1m , 可以分辨出建筑、道路、车辆, 甚至行人的信息, 但是如果降采样到空间分辨率为 30m 时, 则较大的建筑, 农田, 林地可以分辨出, 但是小于 30m 的建筑, 车辆和行人等信息是不能分辨的。降采样的水平向空间实际上反映了不同大小地物存在的尺度, 或理解为不同的尺度反映不同大小的地物信息, 结合到图像上, 即为图像的不同内容存在不同的水平向尺度。将上述阐述对应到Lowe的论文明确其相应的定义, 可以确定降采样的水平向空间为八度 (octave) 的定义, 每一个八度即为连续降采样的图像之一, 可以称之为组。而对每一降采样 (octave) 图像进行高斯卷积核计算, 可以设置连续的多个 σ 值, 这些不同参数下的卷积结果即为每一八度的层。组 (降采样-八度, octave) 反映图像内容 (地物) 的水平向尺度 (空间分辨率), 与层 (不同 σ 值高斯核卷积) 反映差异程度的纵深空间, 构成了高斯金字塔的尺度空间)。为方便理解, 引用 *Distinctive Image Features from Scale-Invariant Keypoints* 中的图, 具体也可以搜索查看该论文:



可添加完整论文引用, 描述作者名时需要统一形式, 以免造成误解, 并对 octave 进行明确描述

Lowe, David G. "Distinctive image features from scale-invariant keypoints." *International journal of computer vision* 60.2 (2004): 91-110.

为了让尺度体现其连续性, 高斯金字塔在简单降采样的基础上加上了高斯滤波。将图像金字塔每层的一张图像使用不同参数做高斯模糊, 使得金字塔的每层含有多张高斯模糊图像, 将金字塔每层多张图像合称为一组 (Octave), 金字塔每层只有一组图像, 组数和金字塔层数相等, 每组含有多张 (也叫层 Interval) 图像。另外, 降采样时, 高斯金字塔上一组图像的初始图像 (底层图像) 是由前一组图像的倒数第三张图像隔点采样得到的。

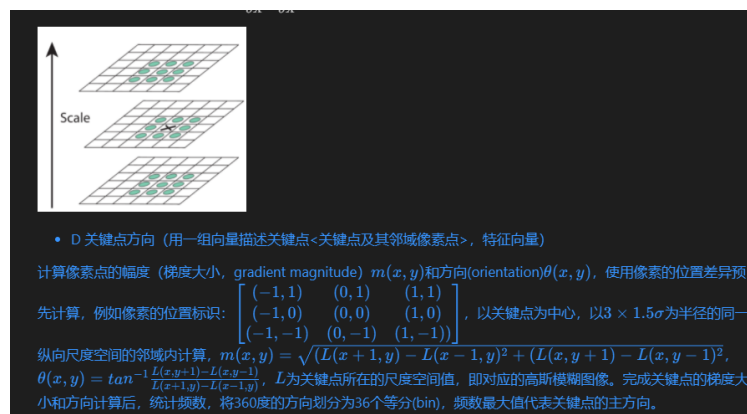
原文链接: https://blog.csdn.net/qq_29598161/article/details/106166843

2. 图片超链接不全，此问题还有多处，此后不再提示【可能和 markdown 解释器有关，不加也不会造成影响，但是严格来说，是应该加的】

```
<a href=""></a>
```

D. Lowe将图像的尺度空间定义为 $L(x, y, \sigma) = G(x, y, \sigma) * I(x, y)$ ，其中 $G(x, y, \sigma)$ 为变化尺度的高斯函数(2维)， $I(x, y)$ 为原图像(像素位置)， $*$ 表示卷积运算。

下图是不加的结果

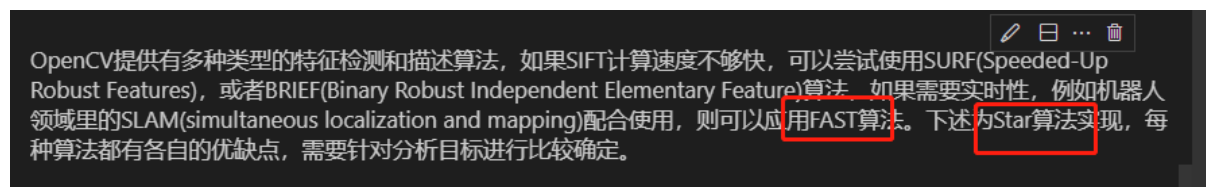


蓝色字自动添加的超链接效果，

3. Fast, star 算法没有书写全名

Fast : Features from accelerated segment test

Star 没有找到完整名字



STAR(CenSurE) in OpenCV

STAR is a feature detector derived from CenSurE. Unlike CenSurE however, which uses polygons like squares, hexagons and octagons to approach a circle, Star emulates a circle with 2 overlapping squares: 1 upright and 1 45-degree rotated. These polygons are bi-level. They can be seen as polygons with thick borders. The borders and the enclosed area have weights of opposing signs. This has better computational characteristics than other scale-space detectors and it is capable of real-time implementation. In contrast to SIFT and SURF, which find extrema at sub-sampled pixels that compromises accuracy at larger scales, CenSurE creates a feature vector using full spatial resolution at all scales in the pyramid.

https://docs.opencv.org/3.4/dc/d7d/tutorial_py_brief.html

4. 此处讨论一下注释的语言风格问题

不少代码中，中英文注释混用，打印信息中英文也有混用，是否应该全部统一为中文或者英文，或者对于打印的信息，对关键中英文，进行同时备注

2.4.2 sentinel-2 及超像素级分割下高空分辨率特征尺度界定

1. 此处，调整下描述展示方式，可以第一个英文单词，可以采取这样的格式，或者直接采取用表格下面以第一个度为例叙述

度 (degrees)，用于 Geodetic coordinates WGS84 (EPSG:4326)坐标系统，是 1984 年定义的世界大地测量系统(World Geodetic System)，用于 GPS 设备来定义地球位置的经纬度坐标。

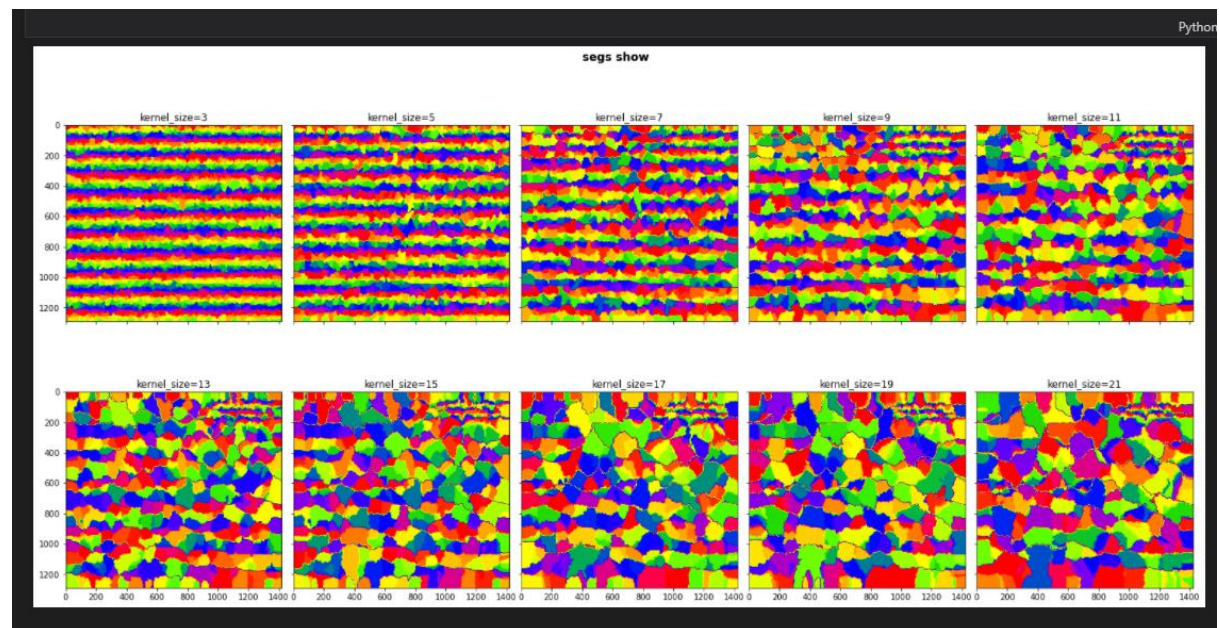
几个世纪以来，人们一直在使用坐标系统和地图投影将地球的形状转换成可用的平面地图。而世界地图很大，不能直接在电脑上显示，所以引出快速浏览和缩放地图的机制，地图瓦片(map tiles)。将世界划分为很多小方块，每个小方块都有固定的地理面积和规模。这样可以在不加载整个地图的情况下浏览其中的一小部分。这涉及到几种表示方法，大地坐标，投影系统，像素坐标和金字塔瓦片，及它们之间的相互转换。

1. 度——**Degrees** Geodetic coordinates WGS84 (EPSG:4326): 使用1984年定义的世界大地测量系统(World Geodetic System)，GPS设备用于定义地球位置的经纬度坐标。
2. 米——**Meters** Projected coordinates Spherical Mercator (EPSG:3857): 全球投影坐标(Global projected coordinates)，用于GIS，A Web Map Tile Service (WM(TS)服务的栅格瓦片(raster tile)生成。
3. 像素——**Pixels** Screen coordinates XY pixels at zoom: 影像金字塔每一层(each level of the pyramid)的特定缩放像素坐标。顶级(zoom=0)通常有 256×256 像素，下一级为 512×512 等。带有屏幕的设备 (电脑，手机) 等在定义的缩放级别计算像素坐标，并确定应该从服务器加载的区域用于可视屏幕。
4. 瓦片——**Tiles** Tile coordinates Tile Map Service (ZXY): 影像金字塔中指定缩放级别下(zoom level)瓦片的索引，即x轴和y轴的位置/索引。每一级别下所有瓦片都有相同的尺寸，通常为 256×256 像素。就是由粗到细不同分辨率的影像集合。其底部为图像的高分辨率表示，为原始图像，瓦片数应与原始图像的大小同；顶部为低分辨率的近似影像，最顶层只有1个瓦片，而后为4，16等。

2. 逗号(“，”)和及不要同时使用

• 05 - 显示分割图像，分割边界。

3. 图像分割，最好展示下原始图像，再展示分割后的结果



4. 该部分整体评价及修改意见

2.4.2.2 超像素级分割下高空分辨率特征尺度界定

在景观生态学中，斑块 - 廊道 - 基质模型是构成景观空间结构，描述景观空间异质性的一个基本模式。其中，斑块是指景观中相对均质的非线性区域。自然界各种等级系统都普遍存在时间和空间的斑块化。反应系统内部小、形状、边界性质及斑块的距离等空间分布特征构成了不同的生态带，形成了生态系统的差异，调节生态景观要素，例如河流廊道，生态廊道等。其中生态廊道又称野生动物生态廊道或绿色廊道，是指用于连接因生境的区域。生态廊道有利于野生动物的迁移扩散，提高生境间的连接，促进濒危物种不同群间的基因交流，大，连接性最好的景观要素类型。斑廊基景观空间结构的提出为城市格局规划提供了依据，在宏观尺度上研究区域，如何自然界定斑块、廊道和基质的区域？或者即使是一个可以肉眼辨识的斑块，这个斑块自身也是由不同的林地，不同物种的农田等，或者不可见的地表温度变化，物质流动等，那么又如何细分斑块的空间区

一方面需要能够反映地物变化的信息数据，例如遥感影像的各个波段对不同地物的探测，sentinel-2影像中植被健康信息；或衍生数据，如反演的地表温度，以及NDVI等反应植被分布的指数，NDWI反应水体分布的指数。在分析这些数据时，可以介入超像素级分割的概念，探索由像素（或空间点数据）局部分组形成的区域，的区域优先聚集即分割，分割区域的变化根据所提供反应不同内容的数据确定，例如探索植被分布的NDVI波段，例如red,green和blue波段组合更倾向于优先聚集同一地物，例如建筑区域，林地区域等。

超像素级分割是一种语义分割，是计算机视觉的基本方法，可以更加精准的执行地物分割、探测和分类等任务。在探索地物变化和地物之间的关系提供一新的策略。scikit-image提供了四种分割的算法。在下述实验中计算Felzenszwalb方法在分割图像时，虽然逐级增加scale参数大小，但是分割的图像并不是上级区域覆盖下级区域。因此选用Quickshift方法，指定逐级增加的kernel_size参数，获取不同深度分割的结果。通过计算逐级覆盖的研究区域不同深度分割下区域间的关联，以及区域的差异性程度。

从 04 - Felzenszwalb 超像素级分割方法开始，

对于实验使用的数据，所做的处理，以及最终达到的效果，得出的结果，描述太过于简略，此部分需要添加内容

本书其他部分也有此问题，

文字介绍应与代码相对应，

绘制显示的图像应该有解释，

代码中的注释不能作为正文中内容的代替，正文中仍需对方法，图表及结果进行解释叙述