




# 一种融合密度聚类与区域生长 算法的快速障碍物检测方法

汇报人:

2024-01-21



目

CONTENCT

录

- 引言
- 密度聚类算法原理及实现
- 区域生长算法原理及实现
- 融合密度聚类与区域生长算法障碍物检测方法
- 实验结果与分析
- 结论与展望



# 01

## 引言

# 研究背景与意义

自动驾驶技术快速发展，障碍物检测是其中的关键技术之一。

障碍物检测的准确性和实时性对自动驾驶系统的安全性和稳定性至关重要。

融合密度聚类与区域生长算法的快速障碍物检测方法，旨在提高检测的准确性和实时性，为自动驾驶技术的发展提供有力支持。





# 国内外研究现状及发展趋势



## 国内外研究现状

目前，障碍物检测方法主要包括基于图像处理、基于深度学习和基于传感器融合等方法。其中，基于图像处理的方法受限于光照、天气等环境因素；基于深度学习的方法需要大量的标注数据和计算资源；基于传感器融合的方法则受限于传感器类型和数量。

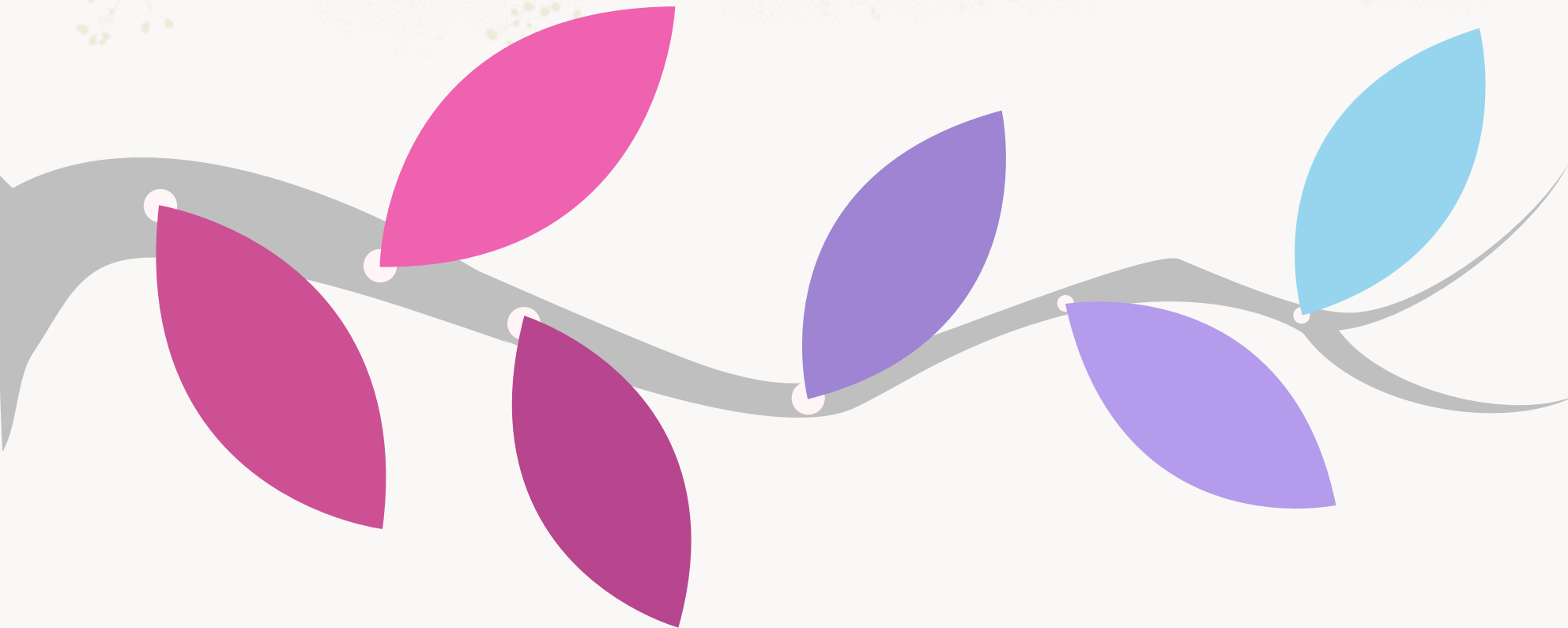
## 发展趋势

随着计算机视觉和人工智能技术的不断发展，未来障碍物检测方法将更加注重多模态数据的融合、实时性的提升以及鲁棒性的增强。





# 本文主要研究内容及创新点



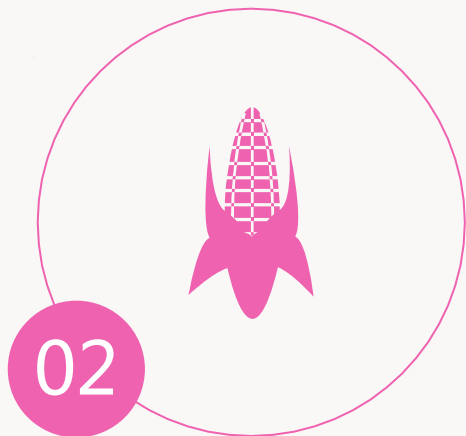
- 主要研究内容：本文提出了一种融合密度聚类与区域生长算法的快速障碍物检测方法。首先，利用密度聚类算法对点云数据进行初步聚类，得到障碍物的候选区域；然后，采用区域生长算法对候选区域进行精细分割，得到准确的障碍物边界；最后，通过实验验证所提方法的有效性和实时性。



# 本文主要研究内容及创新点



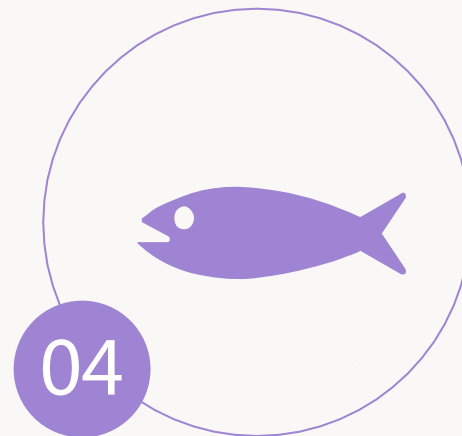
创新点：本文的创新点主要包括以下三个方面



创新点：本文的创新点主要包括以下三个方面



创新点：本文的创新点主要包括以下三个方面



创新点：本文的创新点主要包括以下三个方面



# 02

## 密度聚类算法原理及实现





# 密度聚类算法基本思想



基于密度的聚类假设聚类结构能够通过样本分布的紧密程度确定。

通常情况下，密度聚类算法从样本密度的角度来考察样本之间的可连接性，并基于可连接样本不断扩展聚类簇以获得最终的聚类结果。



# DBSCAN算法原理及实现

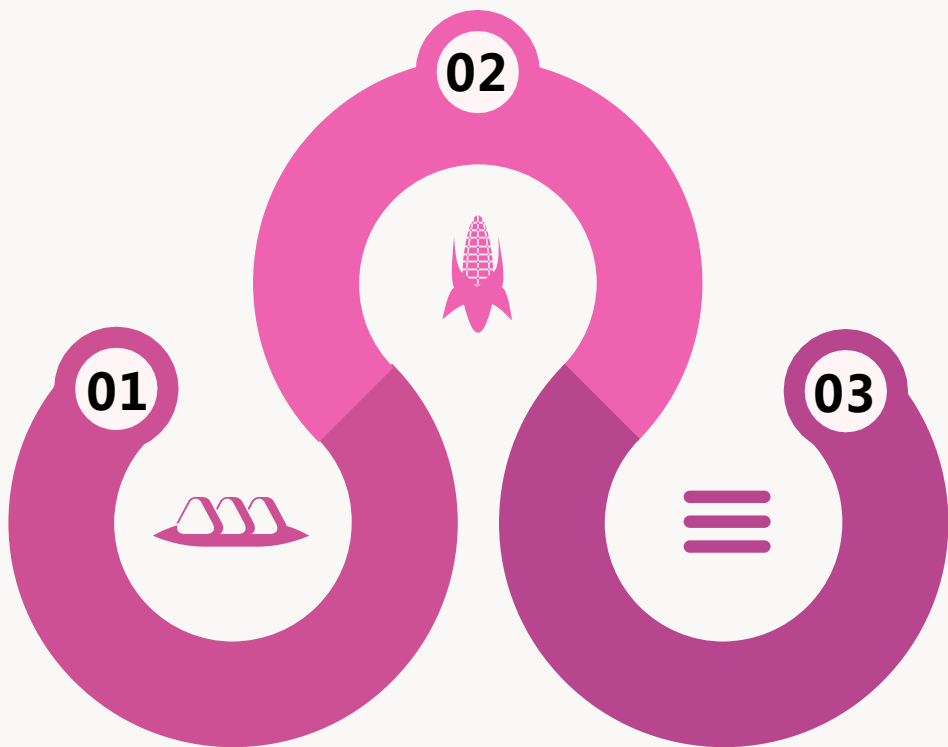
DBSCAN是一种典型的密度聚类算法，它基于一组邻域参数来刻画样本分布的紧密程度。

VS

算法原理：DBSCAN通过检查数据集中每点的邻域来寻找聚类的。如果某个点的邻域包含足够数量的点，则创建一个以该点为核心对象的簇；然后，DBSCAN迭代地聚集从这些核心对象直接密度可达的对象，这个过程可能涉及一些密度可达簇的合并。



# DBSCAN算法原理及实现



实现步骤



初始化核心对象集合为空集。



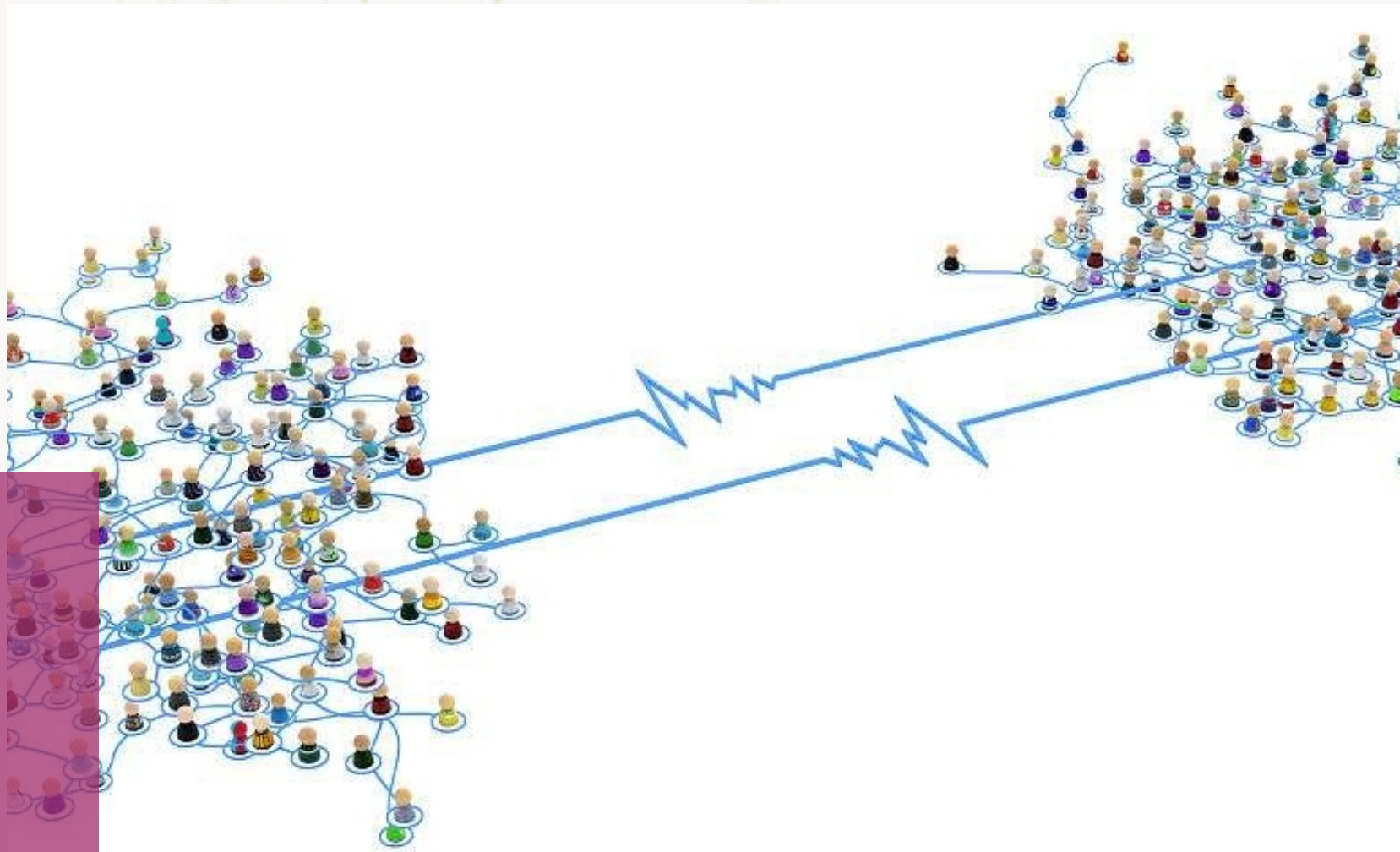
遍历数据集中的每个样本点，检查其邻域内的样本数量是否达到设定的阈值。



# DBSCAN算法原理及实现

如果达到阈值，则将该点加入核心对象集合，并对其进行了扩展，即找出所有从该点密度可达的样本点。

重复上述过程，直到所有核心对象都被处理完毕。







# 改进型DBSCAN算法设计及优化

151°20'50"W

151°11'22"W

01

针对传统DBSCAN算法对邻域参数敏感的问题，可以设计一种自适应确定邻域参数的方法。例如，可以利用K近邻的思想动态计算每个样本点的邻域半径和密度阈值。

02

为了提高算法效率，可以采用空间索引技术（如KD树、R树等）来加速邻域查询过程。此外，还可以使用并行计算技术来进一步提高算法的处理速度。

03

针对DBSCAN算法无法处理噪声点和异常点的问题，可以引入一种机制来识别并处理这些点。例如，可以设置一个噪声点判断阈值，将低于该阈值的点视为噪声点，并采取相应措施（如删除或单独处理）。

151°30'18"W

151°20'50"W

15





# 03

## 区域生长算法原理及实现

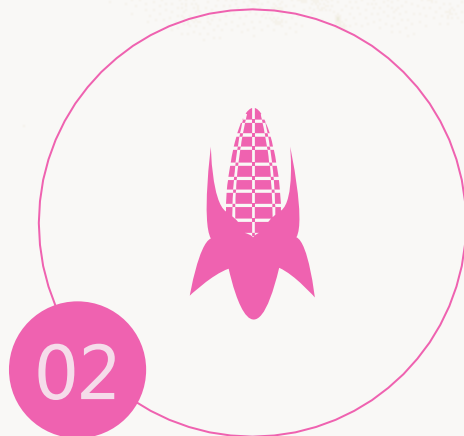


# 区域生长算法基本思想



## 种子点选取

选择图像中的一个像素作为种子点，通常选择图像中的某个特征点或随机点。



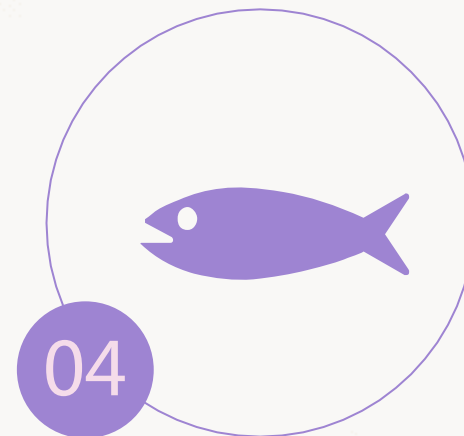
## 生长准则

定义一定的生长准则，用于判断邻近像素是否与种子点属于同一区域。



## 区域生长

从种子点开始，按照生长准则逐步将邻近像素加入到同一区域中，直到无法再加入新的像素。

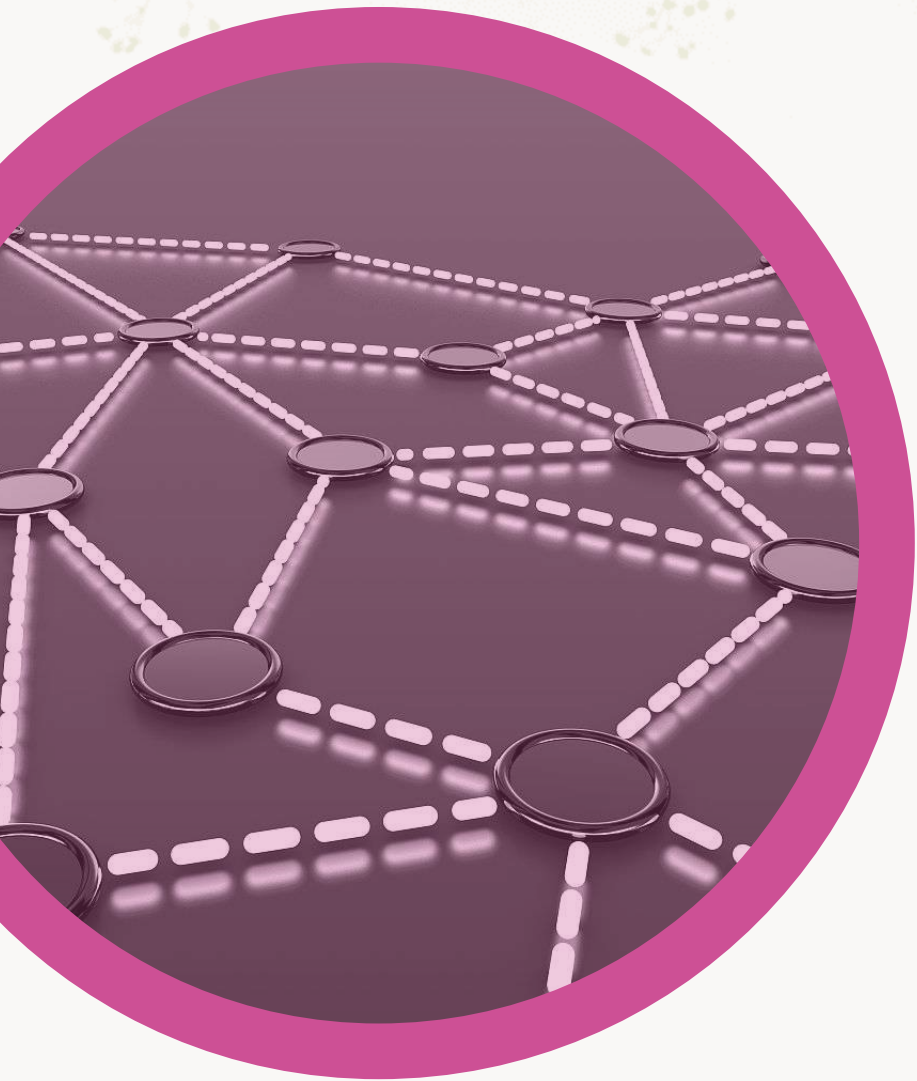


## 结果输出

输出最终的区域生长结果，即同一区域内的所有像素。



# 基于像素点区域生长算法设计及实现



01

初始化种子点列表，将种子点加入列表。

02

遍历种子点列表，对每个种子点进行如下操作

03

搜索种子点的邻域像素。



# 基于像素点区域生长算法设计及实现

根据生长准则判断邻域像素是否与种子点属于同一区域。

将符合条件的邻域像素加入到种子点所在区域，并将其标记为已访问。

重复上述步骤，直到所有种子点都被处理完毕。

输出最终的区域生长结果。





# 基于图像分割区域生长算法设计及实现



01

读取待处理的图像数据。

02

对图像进行预处理操作，如滤波、去噪等。

03

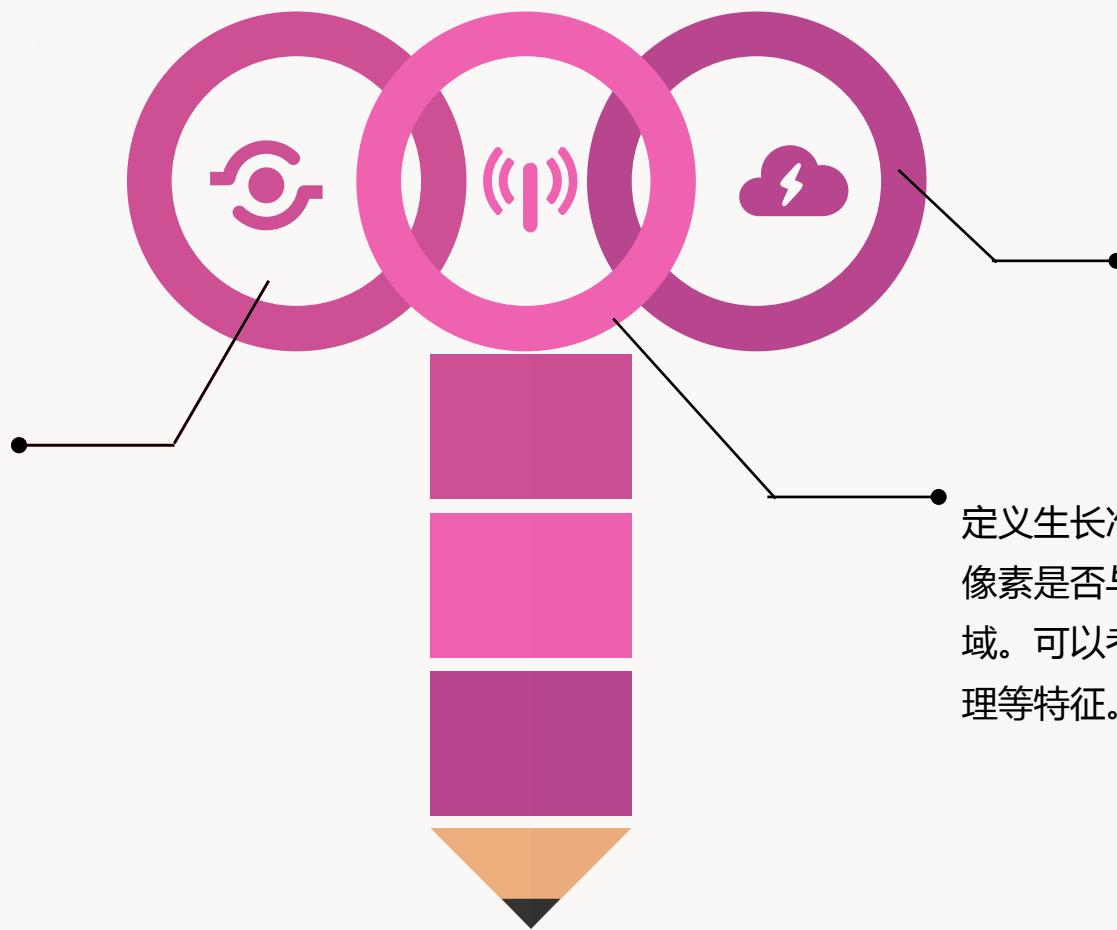
选择合适的图像分割算法，如阈值分割、边缘检测等，对图像进行初步分割。





# 基于图像分割区域生长算法设计及实现

在初步分割的基础上，应用区域生长算法对分割结果进行优化



选择合适的种子点，通常选择分割结果中的边缘像素或特征点。

定义生长准则，用于判断邻近像素是否与种子点属于同一区域。可以考虑像素的颜色、纹理等特征。



# 基于图像分割区域生长算法设计及实现



从种子点开始，按照生长准则逐步将邻近像素加入到同一区域中，直到无法再加入新的像素。

输出最终的区域生长结果，即优化后的图像分割结果。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：  
<https://d.book118.com/148137047063006103>