

OCT大图像处理试验计划

1. 实验目标

- 目标1:** 验证将大图像切分成小图像后进行增强对比度和边缘膨胀处理的效果，确保细节信息得到充分保留。
- 目标2:** 测试小图像拼接成大图用于数据标注的可行性，确保标注过程中图像无明显断裂和拼接痕迹。
- 目标3:** 检查标注完成后切分成小图片用于训练的数据一致性和标注信息准确性。
- 目标4:** 对比训练后的小图像分割模型预测结果再拼接成大图，与原大图进行对比，评估分割质量和整体一致性。

2. 实验环境

- 硬件:**
 - 高分辨率OCT大图像采集设备或已有图像数据集
 - 高性能计算机或服务器（GPU 加速推荐）
- 软件:**
 - 操作系统：Linux / Windows（根据实验环境配置）
 - Python 3.x
 - 图像处理库：Pillow、OpenCV
 - 数据增强与图像拼接工具包
 - 深度学习学习框架：PyTorch
 - 数据标注工具：LabelMe 或类似工具

3. 实验流程

3.1 阶段一：大图像切分与预处理

- 步骤1.1:** 导入OCT大图像数据。
- 步骤1.2:** 将大图像按照预定尺寸（如256×256或512×512）切分成多个小图片。
- 步骤1.3:** 对每个小图片进行数据增强，包括：
 - 对比度增强处理
 - 边缘膨胀处理（如利用边缘检测、膨胀操作）
- 步骤1.4:** 存储处理后的小图像数据，并记录原图与切分图像的对应关系。

3.2 阶段二：小图片拼接与数据标注

- 步骤2.1:** 将预处理过的小图片按照原大图的布局进行拼接。
- 步骤2.2:** 检查拼接后的大图是否存在明显断裂或拼接痕迹。
- 步骤2.3:** 使用数据标注工具对拼接大图进行标注（如区域标注、边缘标注等）。
- 步骤2.4:** 保存标注信息及拼接大图数据。

3.3 阶段三：标注后数据切分与训练数据准备

- **步骤3.1：** 根据标注信息，将标注后的大图重新切分成小图片（保证每个切分的小图包含完整的标注信息）。
- **步骤3.2：** 检查切分后的小图标注信息是否完整且与原大图对齐。
- **步骤3.3：** 构建训练数据集（包括输入图片与对应的标注mask）。
- **步骤3.4：** 数据预处理（如归一化、resize等）准备进入模型训练阶段。

3.4 阶段四：训练与预测后大图拼接

- **步骤4.1：** 使用选定的分割模型（如U-Net或FCN）在切分的小图数据上进行训练。
- **步骤4.2：** 训练完成后，对测试数据（切分的小图）进行预测，得到分割mask。
- **步骤4.3：** 将预测得到的小图分割结果按照原图布局拼接成大图。
- **步骤4.4：** 对拼接后的大图进行整体效果评估（如像素级准确率、IOU、视觉效果评估等）。