

# 图像增强算法优化工作汇报

日期：2025年3月12日

## 工作概述

今天主要对图像增强算法进行了多次优化，目标是提高图像的亮度和对比度，特别是增强暗区细节，同时避免过度曝光。经过多次迭代，我们成功开发了一种能够在保持细节的同时大幅提升图像可见性的算法。

## 算法优化过程

### 1. 初始增强算法

首先实现了基础的增强算法，包括：

- CLAHE（对比度受限的自适应直方图均衡化）
- Gamma校正
- 边缘增强

### 2. 极限增强尝试

应要求，我们将算法参数推向极限：

- CLAHE参数提高到极限 (`clipLimit=60.0`)
- Gamma值降低到0.15以最大程度提亮暗区
- 增加了多级亮度提升 (+80和+120)
- 对亮区进行了二值化处理，使其达到最大亮度

### 3. 二值化处理尝试

尝试了完全二值化的处理方法：

- 使用自适应阈值进行二值化
- 结合Otsu阈值法获得最佳全局对比度
- 应用形态学操作清理二值图像

### 4. 平衡增强算法

发现过度曝光问题后，调整为更平衡的算法：

- 降低CLAHE参数 (`clipLimit=10.0`)
- 提高Gamma值至0.5减少过度曝光
- 使用百分位数 (5-95) 控制对比度拉伸范围
- 添加双边滤波保持边缘同时减少噪点

### 5. 多级区域增强算法（最终版）

最终实现了针对不同亮度区域的多级增强算法：

- 将图像分为暗区 (<80)、中间调 (80-180) 和亮区 (>180)

- 对暗区应用强力亮度提升 (+60)
- 对中间调区域应用适度亮度提升 (+30)
- 对亮区不进行额外增强以避免过度曝光
- 使用更保守的对比度拉伸 (2-98百分位, 最大值限制在230)
- 适度的边缘增强 (权重1.7) 提高细节可见性

## 处理流程

完整的图像处理流程包括：

1. 从源目录提取原始PNG图像
2. 将大图像分割为 $1024 \times 1024$ 的小块
3. 对每个小块应用增强算法
4. 将增强后的小块重新组合成完整图像

## 图像增强效果展示

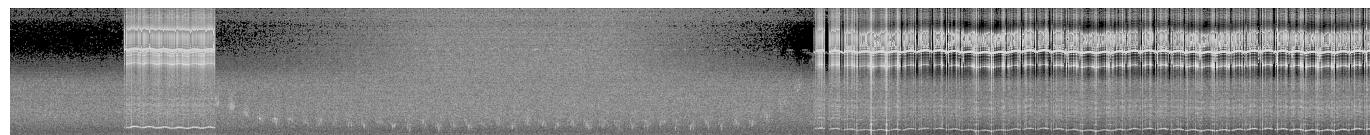
以下是几组原始图像与增强后图像的对比，展示了算法的效果：

### 示例1：普通光照条件下的图像增强

原始图像：



增强后图像：

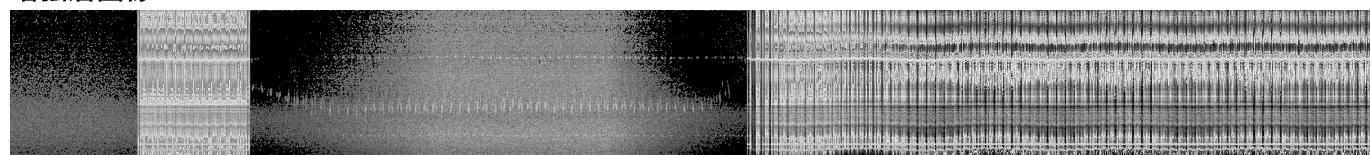


### 示例2：暗区细节增强

原始图像：



增强后图像：

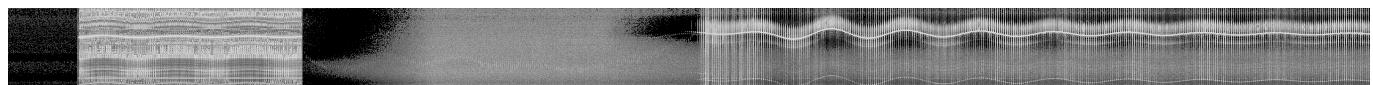


### 示例3：有蓝光条件下的图像增强

原始图像：



增强后图像：



## 后续工作

可能的后续优化方向：

1. 进一步调整暗区、中间调和亮区的阈值
2. 尝试不同的边缘增强参数
3. 考虑添加自适应参数，根据图像特性自动调整增强程度
4. 探索深度学习方法进一步提高图像质量

## 技术细节

最终算法的关键参数：

- 初始CLAHE: clipLimit=8.0, tileGridSize=(8,8)
- Gamma校正: gamma=0.6
- 暗区阈值: 80 (亮度值)
- 亮区阈值: 180 (亮度值)
- 暗区亮度提升: +60
- 中间调亮度提升: +30
- 对比度拉伸: 2-98百分位
- 边缘增强: 权重1.7, 高斯模糊sigma=3
- 双边滤波: d=5, sigmaColor=50, sigmaSpace=50