

Lab3 Poisson Image Edit

By PB17111585 张永停

一、实验内容

- 实现 Poisson Image Editing 算法
- 实现多边形光栅化的扫描线转换算法
- 使用Eigen库求解大型稀疏方程组
- 使用 OpenCV
- 使用矩阵预分解来实时拖动区域显示结果

二、算法描述

（一）Poisson Image Editing

图片可以由像素点的颜色和梯度来刻画，若想要图片在视觉上因剪切造成的冲突感最小，应尽量维持被复制图片的梯度不变，以及被粘贴图片的颜色过度自然。

允许改变被粘贴图片绝对量：颜色，不允许改变相对量：梯度。

本算法通过将复制的图片的梯度复制到被粘贴的图片中来实现图片剪切。

$$\begin{cases} \min_f \iint_{\Omega} |\nabla f|^2, & f|_{\partial\Omega} = f^*|_{\partial\Omega}, \\ \nabla f = 0 \text{ over } \Omega, & f|_{\partial\Omega} = f^*|_{\partial\Omega}, \end{cases}$$

离散形式为

$$\forall p \in \Omega, |N_p| f_p - \sum_{q \in N_p \cap \partial\Omega} f_q^* + \sum_{q \in N_p} v_{pq}$$

使用Eigen的Cholesky分解来解方程

（二）扫描线算法

首先比较得知多边形y轴坐标的最小和最大值

然后依次扫描多边形区域，求出多边形与扫描线的交点并排序

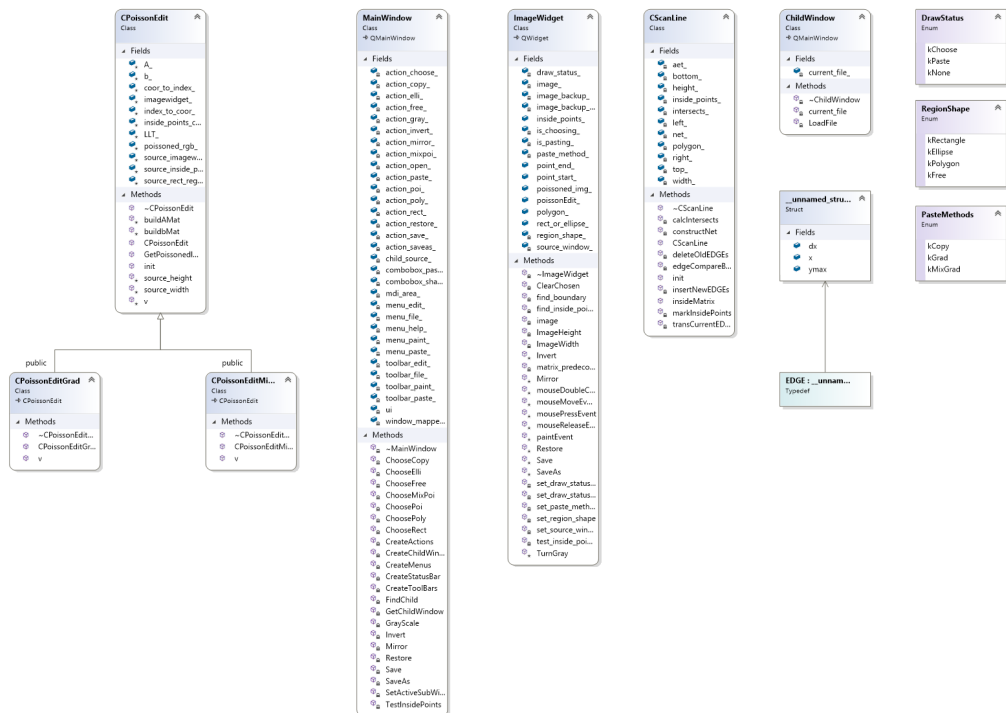
对交点配对，每对顶点之间的点为内点

特殊情况见代码

（三）实施拖动

由于Poisson方程系数矩阵与目标区域位置无关，故可对系数矩阵进行预分解（这里采用Cholesky分解），从而拖动时可以比较快的求解方程

三、代码框架



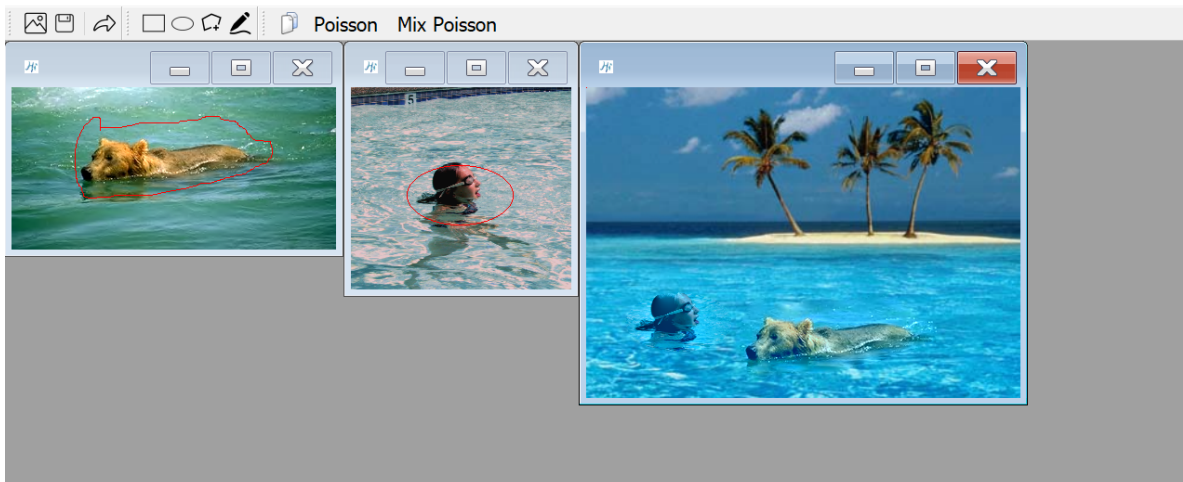
四、实验结果

- 允许使用矩形、椭圆、多边形、随笔画来选择区域
- 粘贴时可以使用直接像素粘贴、Poisson粘贴、MixPoisson粘贴
- 使用时，先在需要复制的图片上选择，再将鼠标移至被粘贴图片所在窗体，再从工具栏选择所需粘贴方法

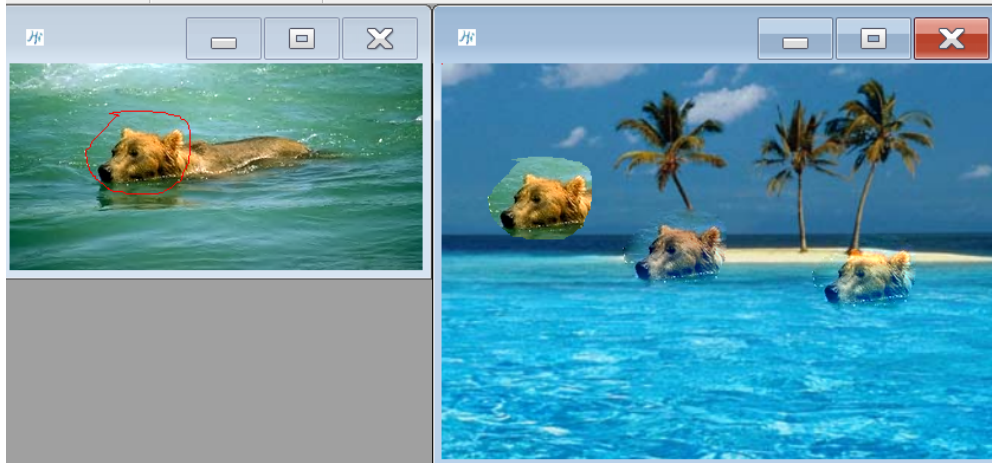
（一）泊松粘贴

⚡ Poisson Image Edit

File Edit Paint Paste Method



（二）从左至右分别为：直接像素粘贴，Poisson，MixPoisson



可以较为清晰的看到，MixPoisson的效果略好于Poisson

（三）图片大小

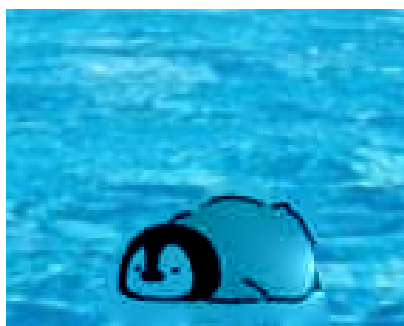
很遗憾大图片拖动极其耗费时间，尤其是过大时，几乎没有反应。

对于小图片，比如狗头，拖动几乎没有延迟。

（四）内存泄漏

emmmmm这个我真的尽力了。。。但我真的查不到哪里的问题，上网搜索说是opencv的mat类在cvt和copy的时候不会自动释放，但我手动释放了呀还是显示内存泄漏TAT，如果助教知道是哪里的问题可以告诉我一下嘛

（五）彩蛋一只



五、实验感想

opencv的rgb是反着，长宽也是反的，调了好久233333

感谢csdn