



# 中国科学技术大学 本科生毕业论文答辩

## 基于样例的图像修复 Exemplar-Based Image Completion

张博栋

PBI0210189

导师：张荣 副教授

2014年6月


# 目录

- 图像修复背景
- 基于样例的图像修复基本算法
- 改进方法
- 搭建平台介绍
- 实验修复结果

# 一、图像修复背景介绍

- 图像修复是利用图像中未破损区域的信息，来修复图中破损区域，并且使得看起来具有真实的效果，图像修复具有广阔的应用范围，如视频中的文字进行去除，剔除照片中不想要的物体，去除图像划痕等。



- 
- 图像修复背景
  - **基于样例的图像修复基本算法**
  - 改进方法
  - 搭建平台介绍
  - 实验修复结果

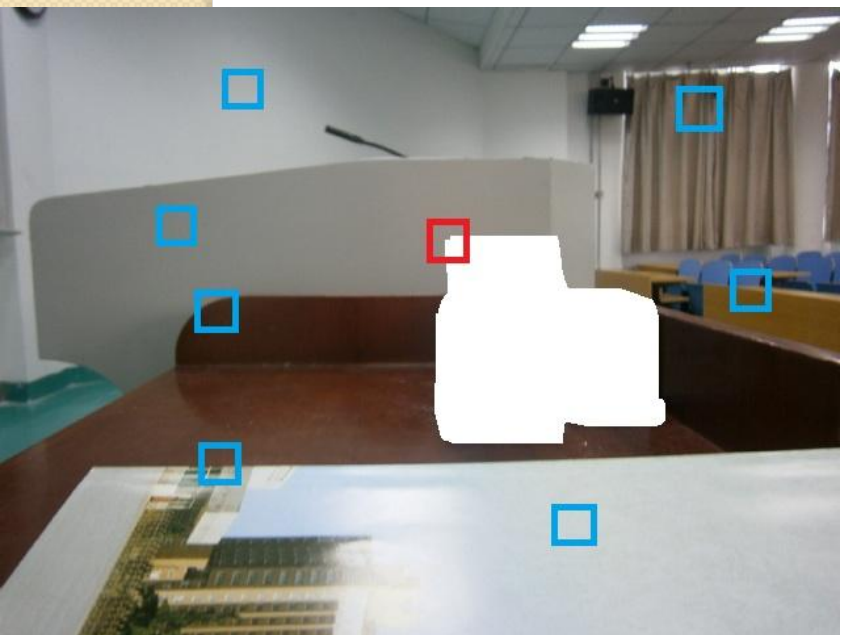
## 二、基于样例的图像修复



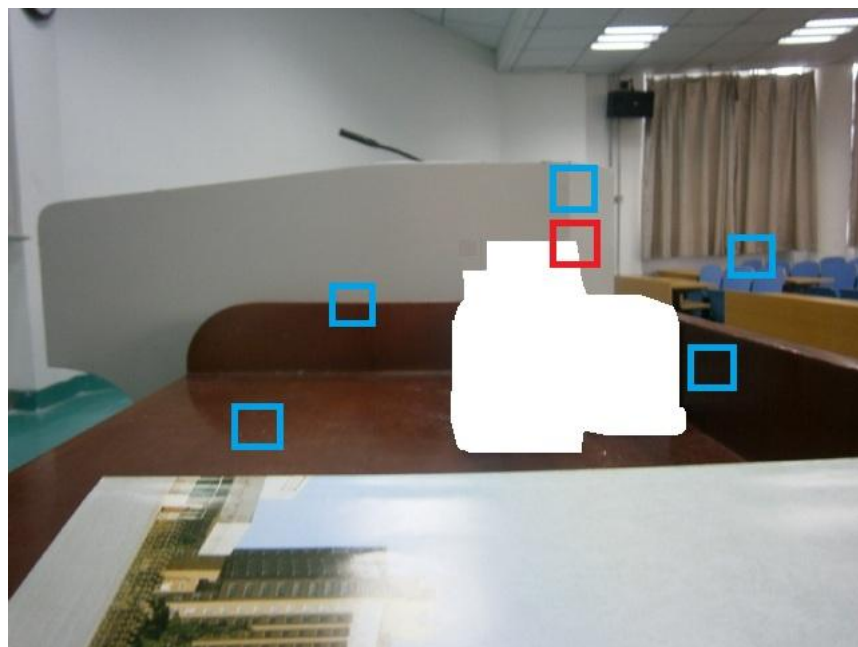
原图



Mask图 用来标明源区域、待修复区域



将最相似的源块像素复制到待修复块



不停迭代直到修复完成

# 源区域中选取最佳源块的标准

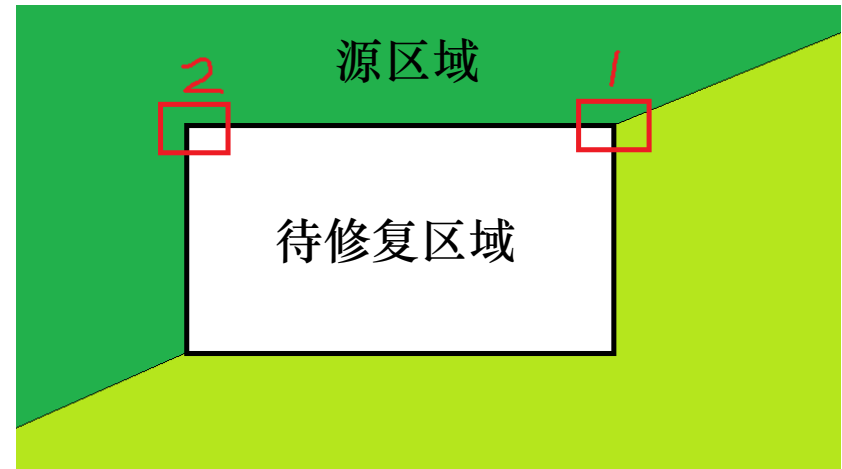
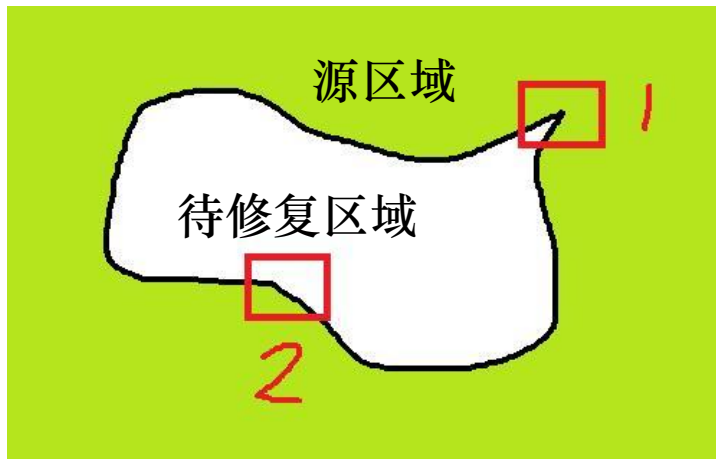
颜色最相近（对应像素点的差的平方和最小） $\Psi_p = \arg \min_{\Psi_q \in \Phi} d(\Psi_p, \Psi_q)$

## 如何决定先修复哪一块

- 含有源区域多的块易被先修复
- 含有结构线的块易被先修复

$$P(p) = C(p)D(p) \quad \text{其中 } C(p) = \frac{\sum_{q \in \Psi_p \cap (I - \Omega)} C(q)}{|\Psi_p|} \quad D(p) = \frac{|\nabla I_p^\perp \cdot \mathbf{n}_p|}{\alpha}$$

$|\Psi_p|$  是  $\Psi_p$  的面积， $\alpha$  是标准化因子， $\mathbf{n}_p$  是与边界切向垂直的单位向量， $\nabla I_p^\perp$  是梯度，在初始时， $C(p)=0, \forall p \in \Omega$ ， $C(p)=1, \forall p \in \Phi$ 。





# 基于样例的图像修复的缺点


- 效率低，寻找源块时间长
- 结构复杂的图像修复质量难以保证
- 最佳源块未必是能成功修复图像的块



蓝色块右侧像素复制给红色块，导致错误的结果



基于样例的图像修复结果

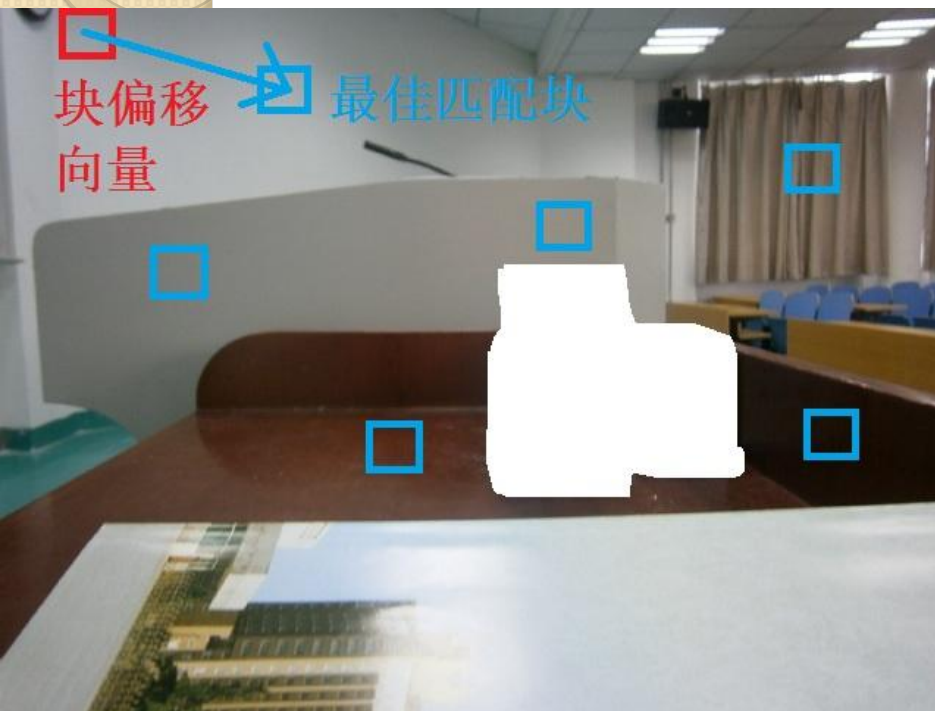
- 
- 图像修复背景
  - 基于样例的图像修复基本算法
  - **改进方法**
  - 搭建平台介绍
  - 实验修复结果



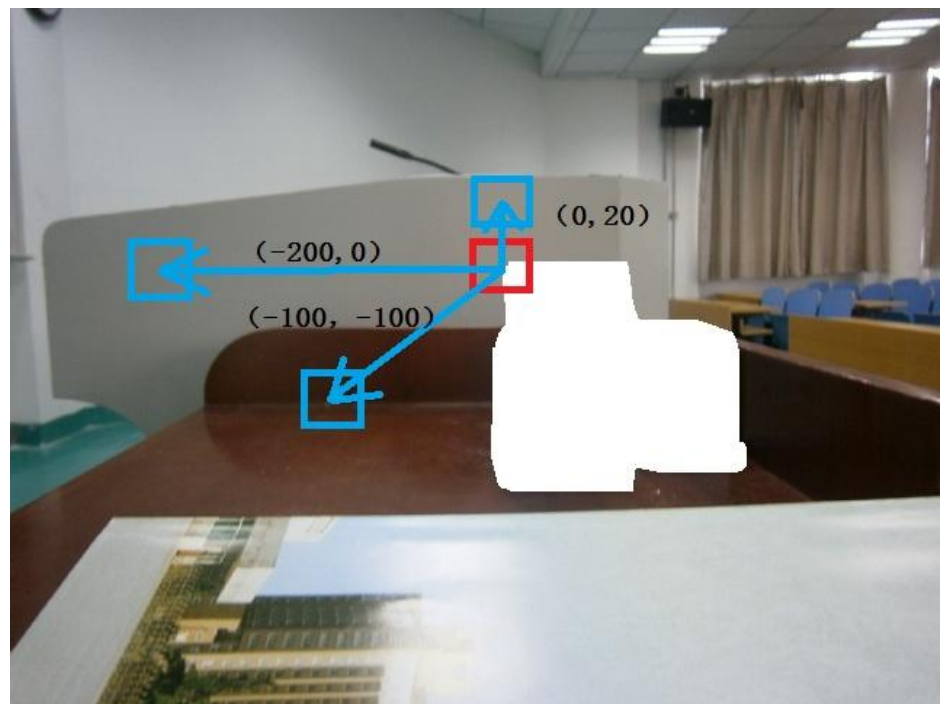
# 改进I: 根据图像块统计特性寻找最佳源块

- 目的：大幅降低寻找源块的时间，同时保证修复质量不会下降
- 修复前通过运算得到最具有代表性的块偏移向量，修复过程中利用向量进行寻找源块
- 由于寻找源块数量较少，时间缩短

- 修复前通过统计每个源块的最佳匹配块，得到块偏移向量的统计特性，其中含有较高频率的块偏移向量。
- 大部分向量频次为0，较高频率的频次可达100以上。
- 利用前1000个高频块偏移向量进行修复



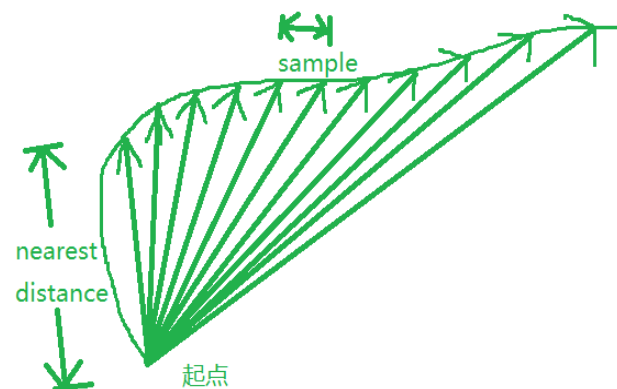
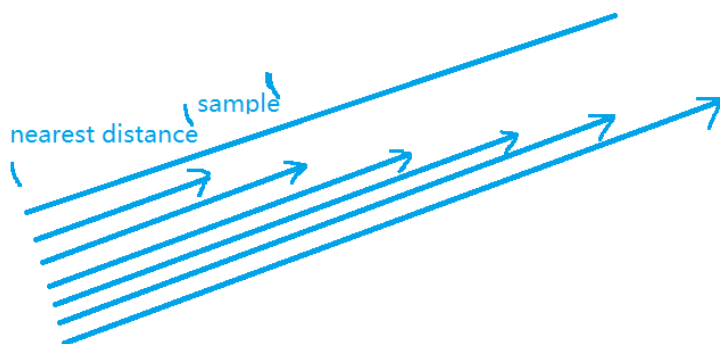
修复前进行块匹配统计



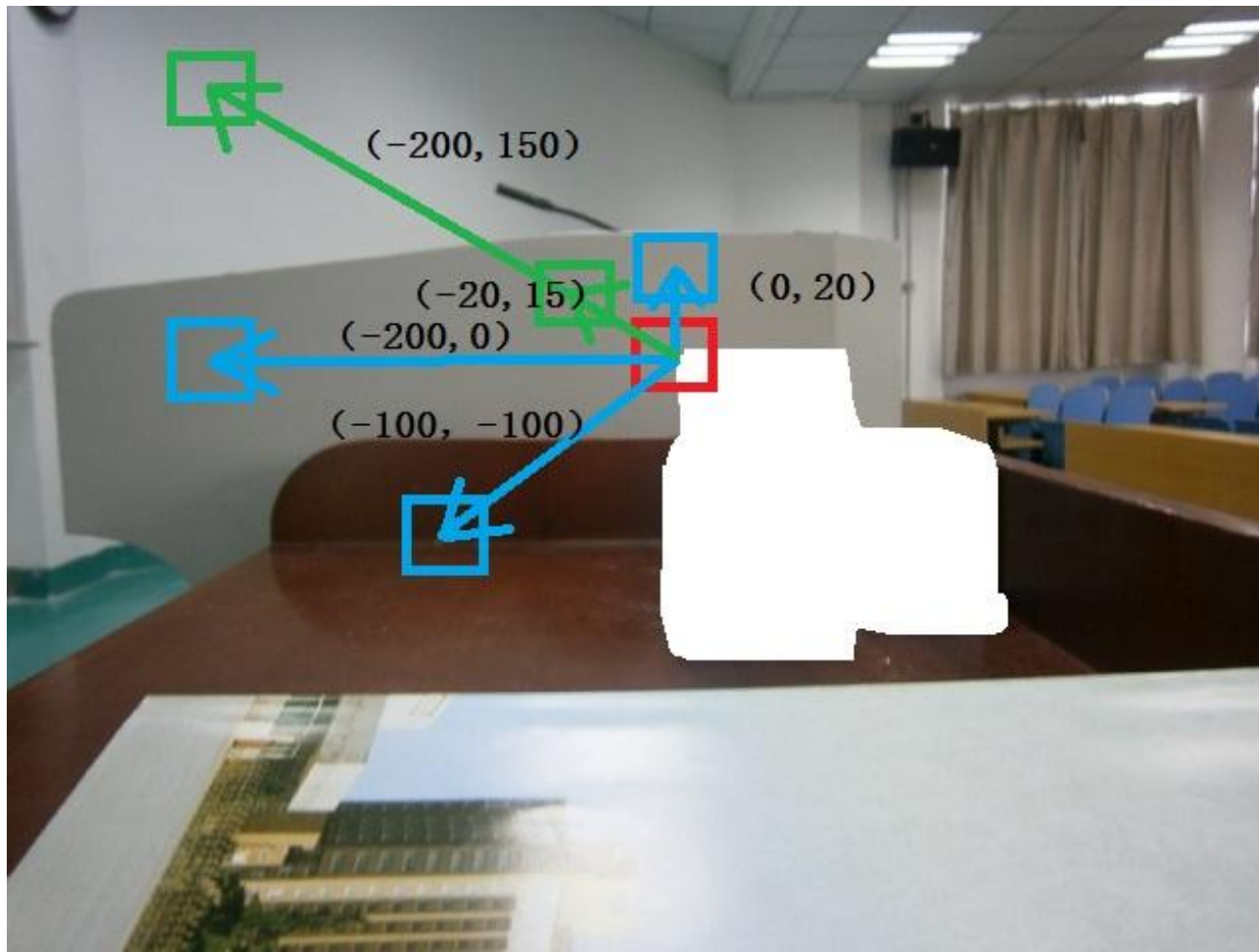
假设得到的高频块偏移向量为  
 $(-200, 0)$ 、 $(0, 20)$ 、 $(-100, -100)$   
则修复时寻找的源块如图所示  
(其他块不再寻找)

## 改进2：用户交互画线

- 为了防止高频块匹配向量并不是能有效修复图像的向量，通过交互加入额外向量纳入块匹配向量中。

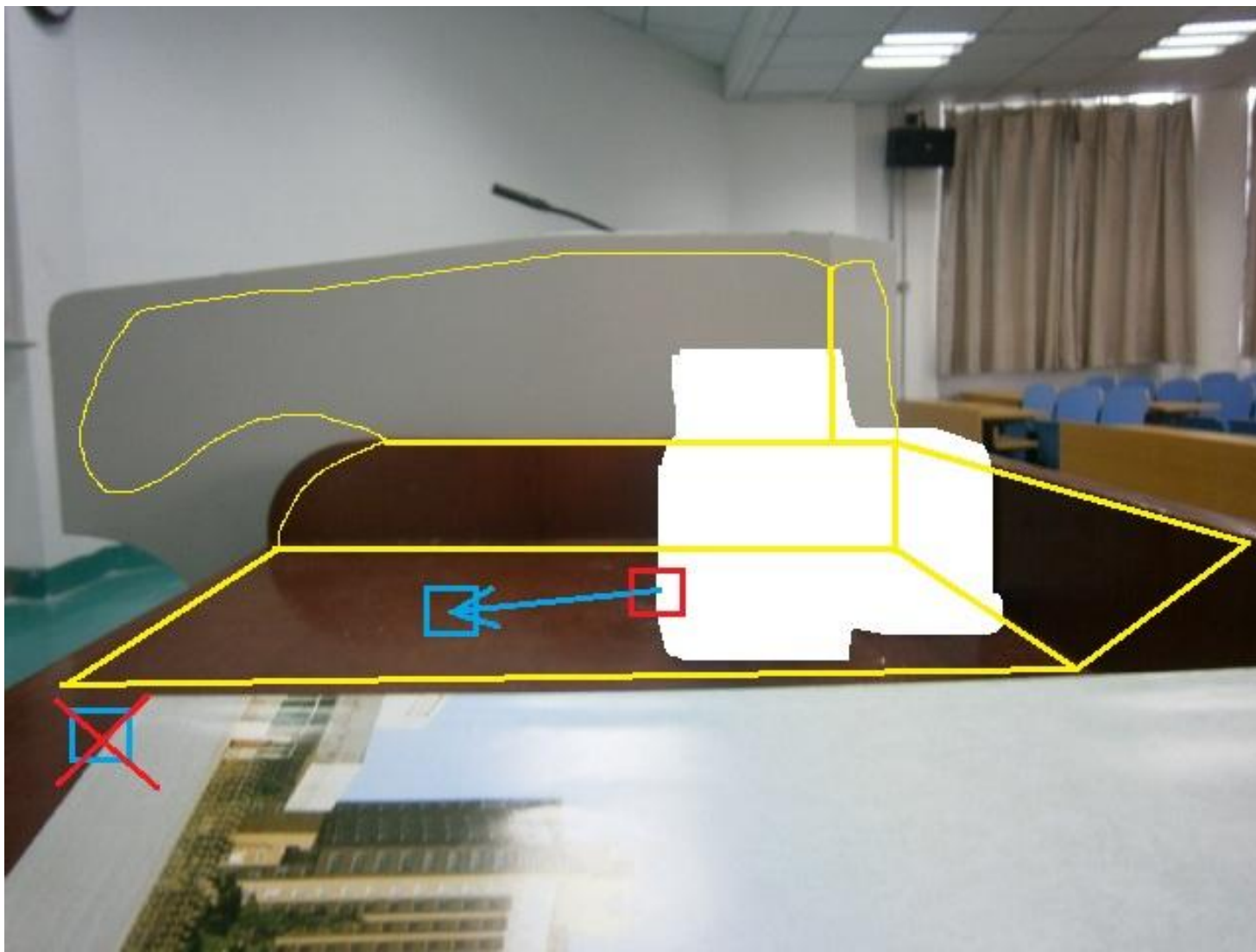


- 假设得到的高频块偏移向量为  
 $(-200, 0)$ 、 $(0, 20)$ 、 $(-100, -100)$   
交互得到的向量为  $(-20, 15)$ 、 $(-200, 150)$  则修复时寻找的块如下



# 改进3：将图像分为不同区域

- 通过用户交互将图像分为不同区域，在寻找最佳源块时，使得同区域源块优先作为最佳源块。






# 最终结果

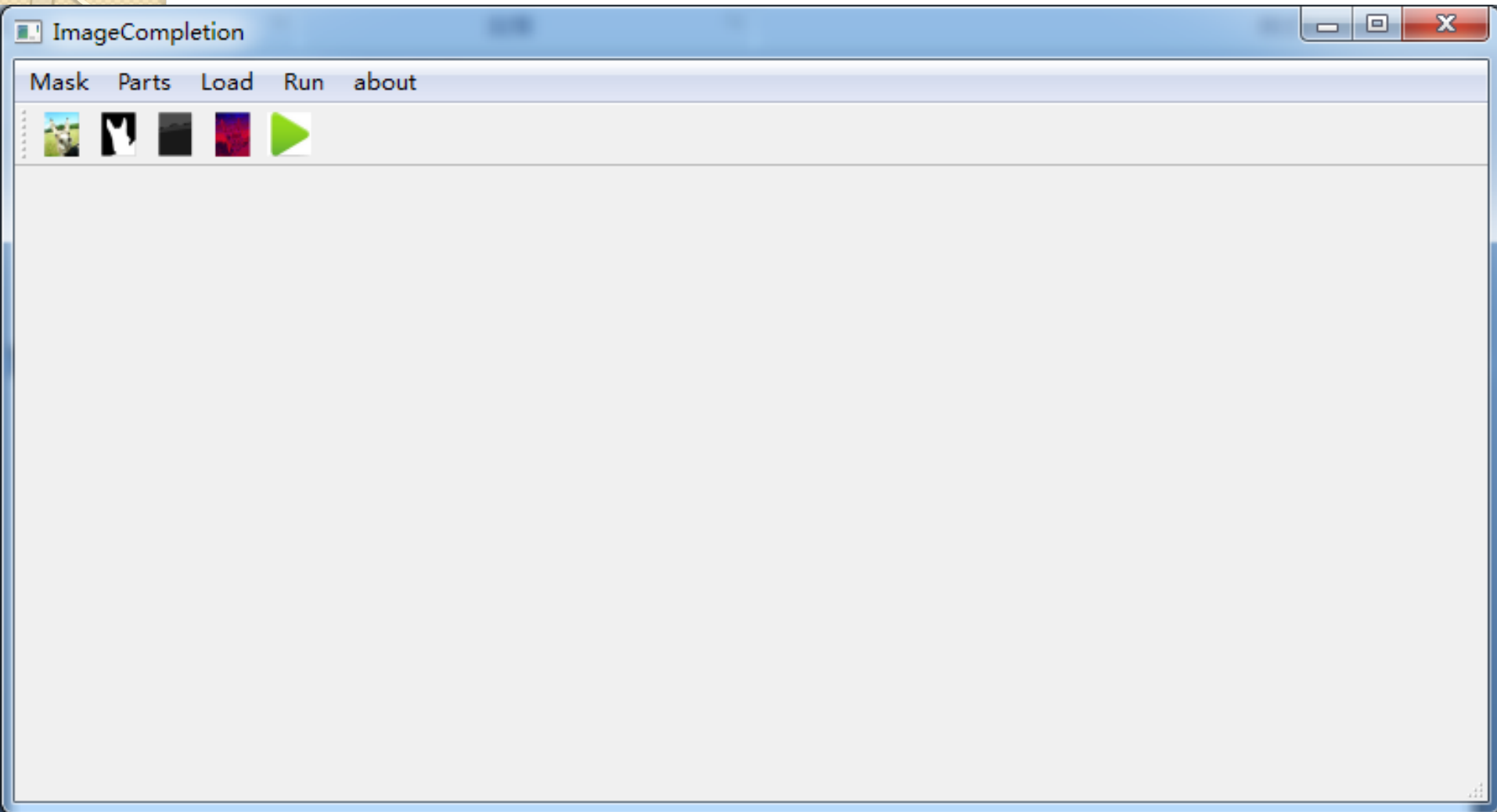




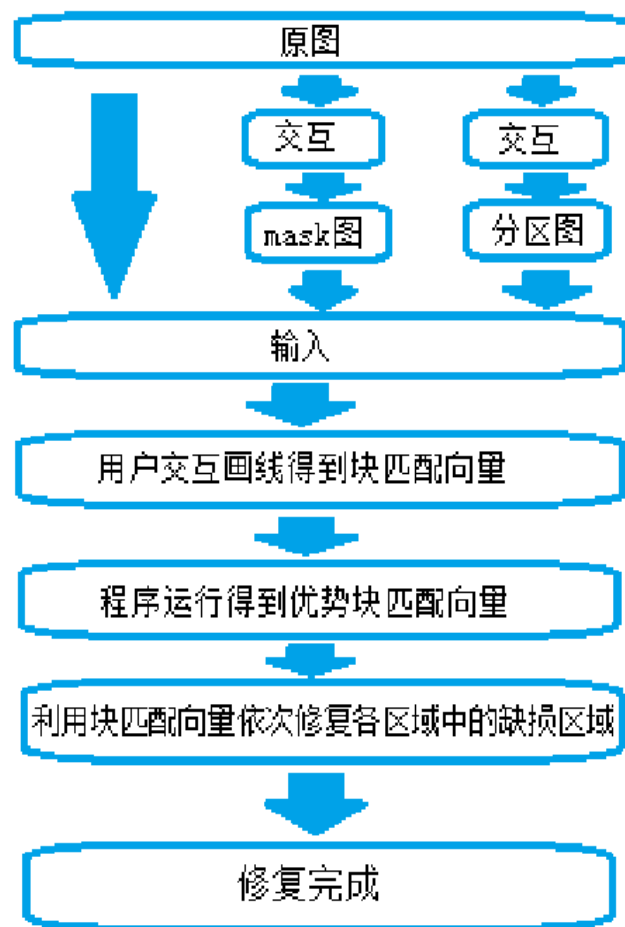
- 
- 图像修复背景
  - 基于样例的图像修复基本算法
  - 改进方法
  - **搭建平台介绍**
  - 实验修复结果

# 程序界面

平台：Visual Studio+QT+OpenCV



# 流程图



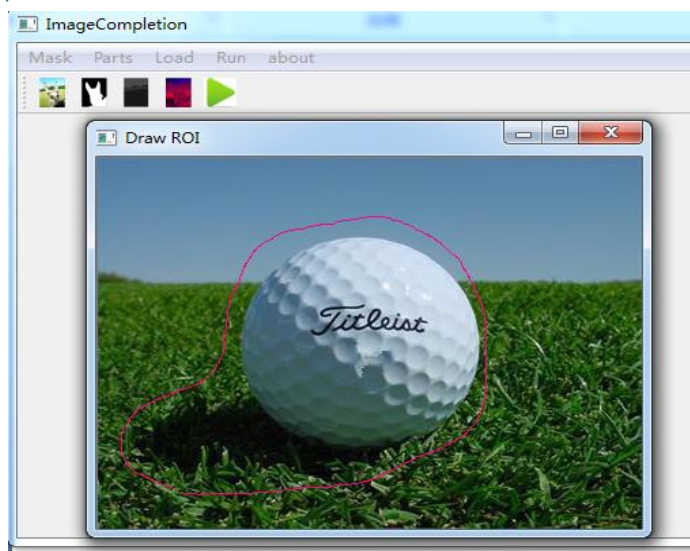
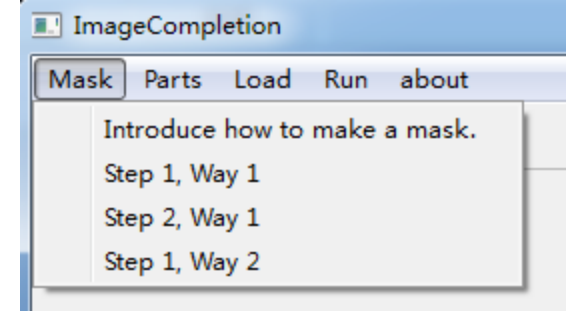
# 得到mask图



原图



利用填涂方式



Mask图

利用画线方式

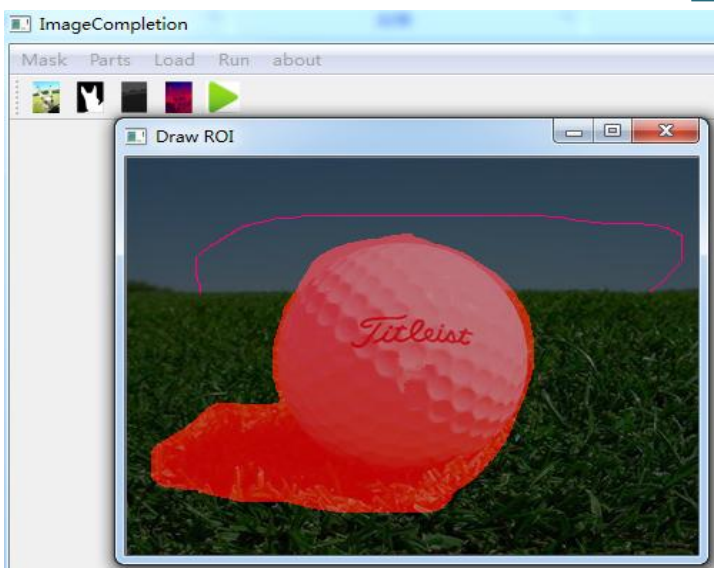
# 得到分区图



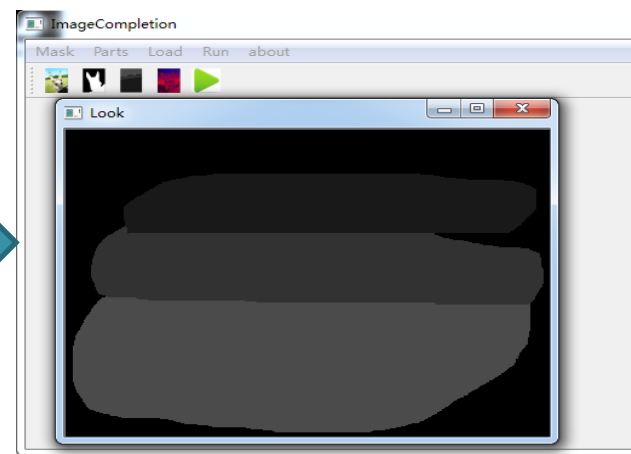
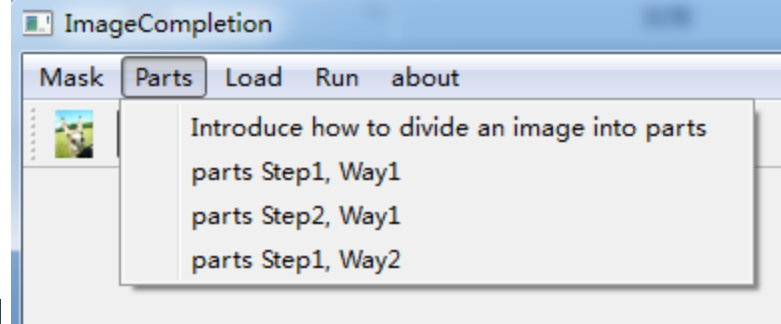
原图



利用填涂方式

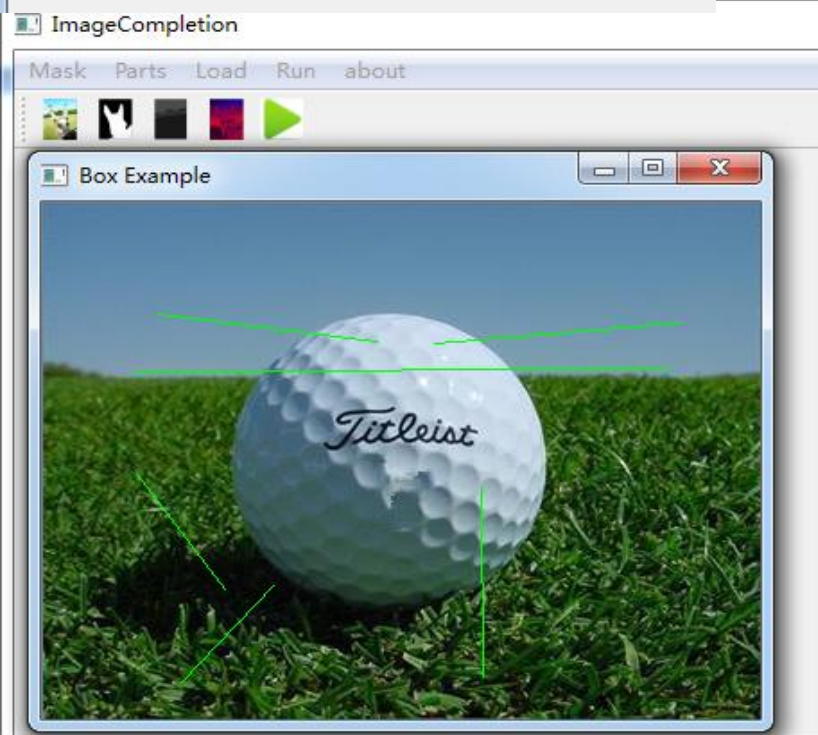
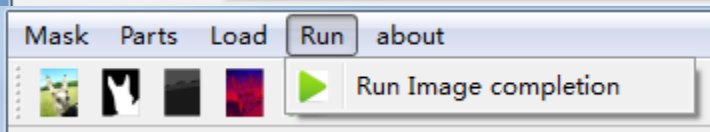
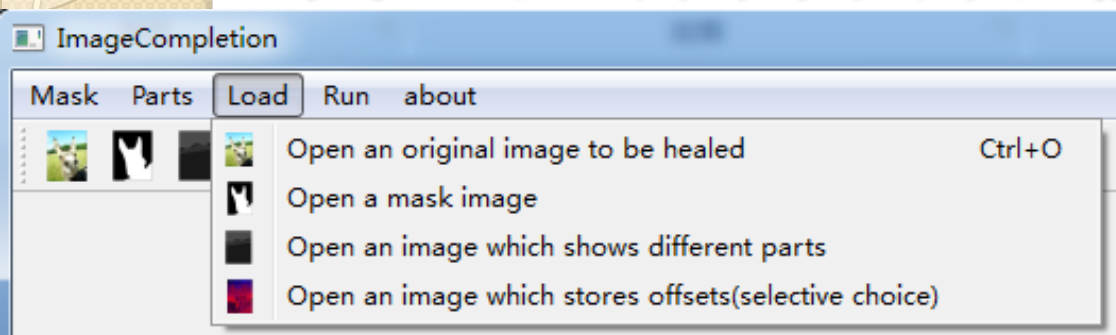


利用画线方式

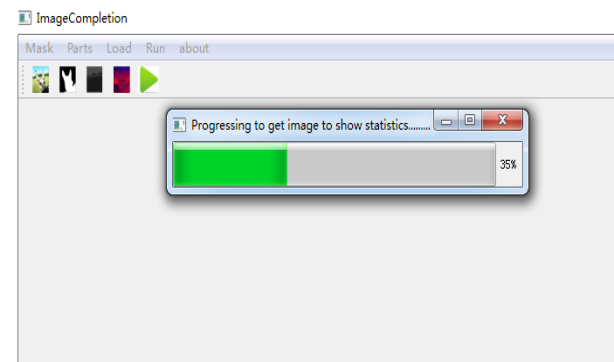


分区图

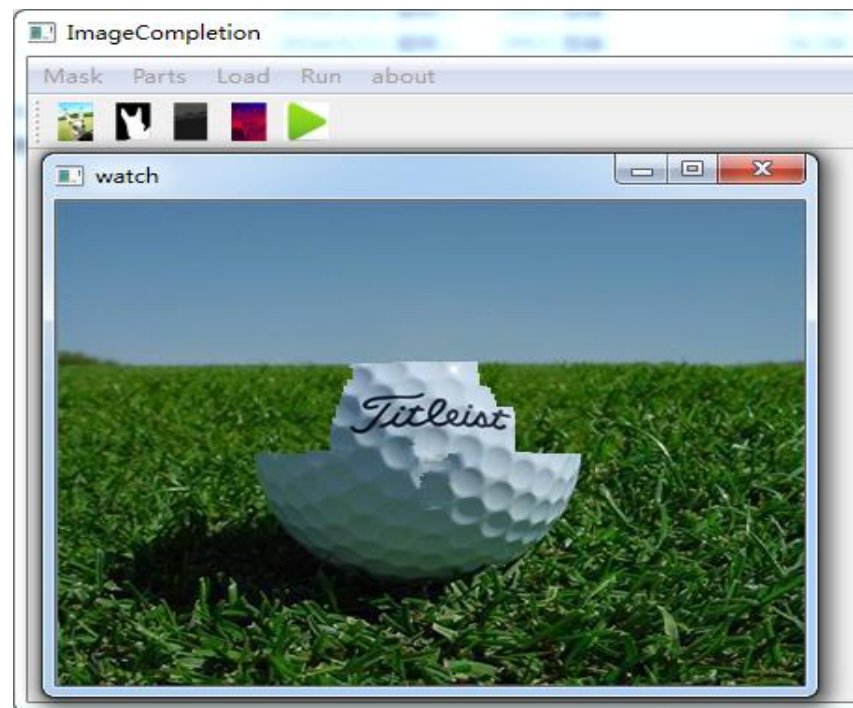
# 载入图像并开始修复



用户交互画线




得到块匹配统计特性



利用窗口观察修复过程



- 
- 图像修复背景
  - 基于样例的图像修复基本算法
  - 改进方法
  - 搭建平台介绍
  - **实验修复结果**



原图



Photoshop  
自动修复结果



传统的基于样例的图像修复结果



本算法修复结果





原图



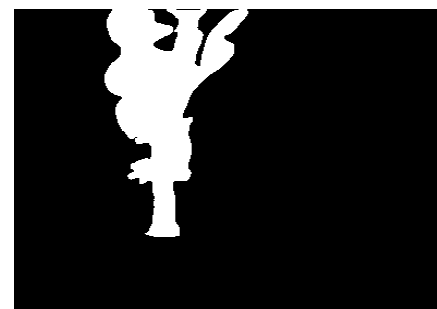
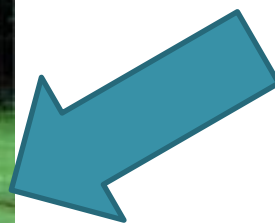
Mask图



中间结果图



最终结果图



Mask图





不使用分区算法得到的结果



使用分区算法得到的结果



原图



修复结果图





原图



修复结果图





原图



修复后的图

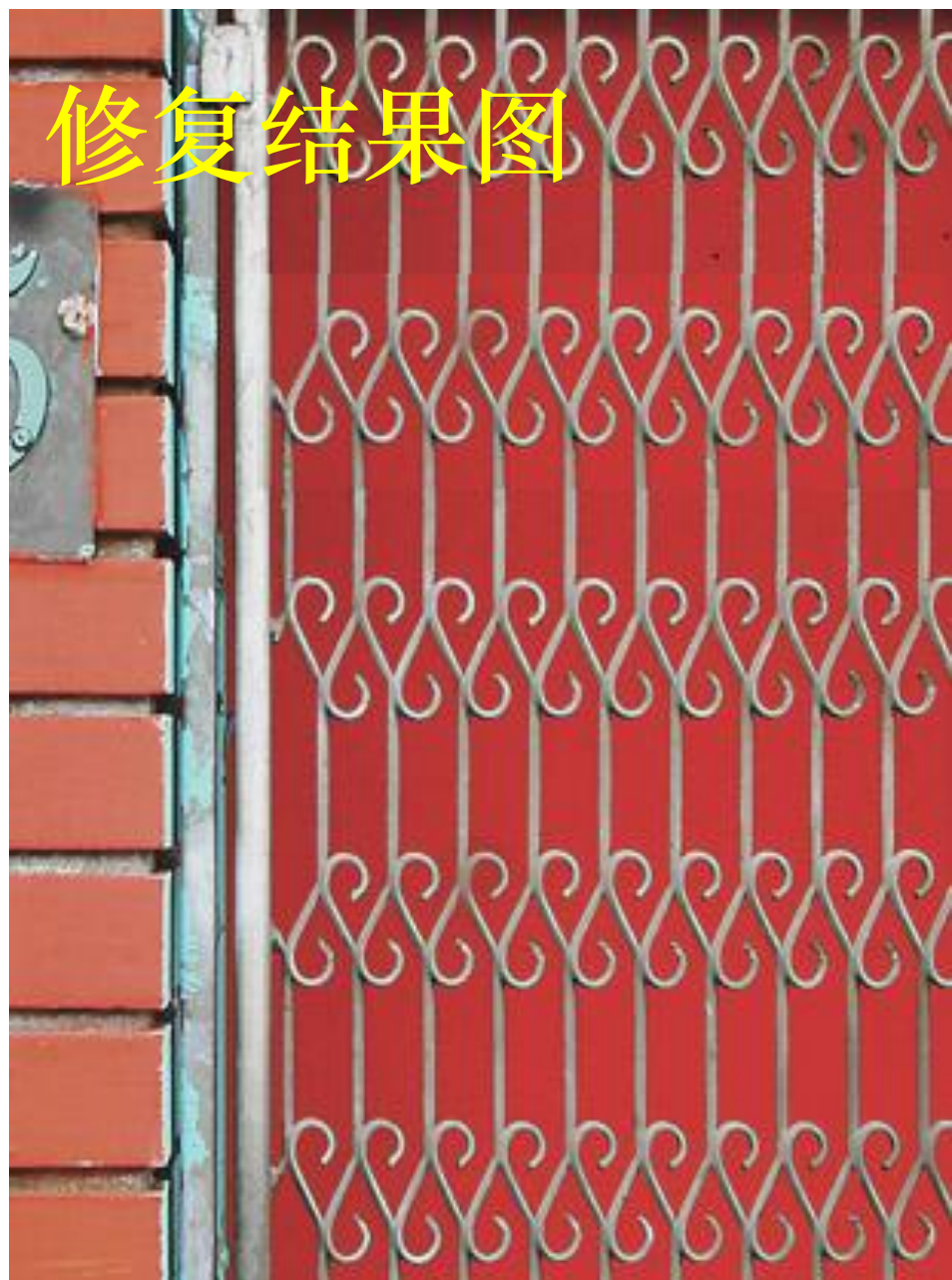




原图



修复结果图





原图



修复结果图





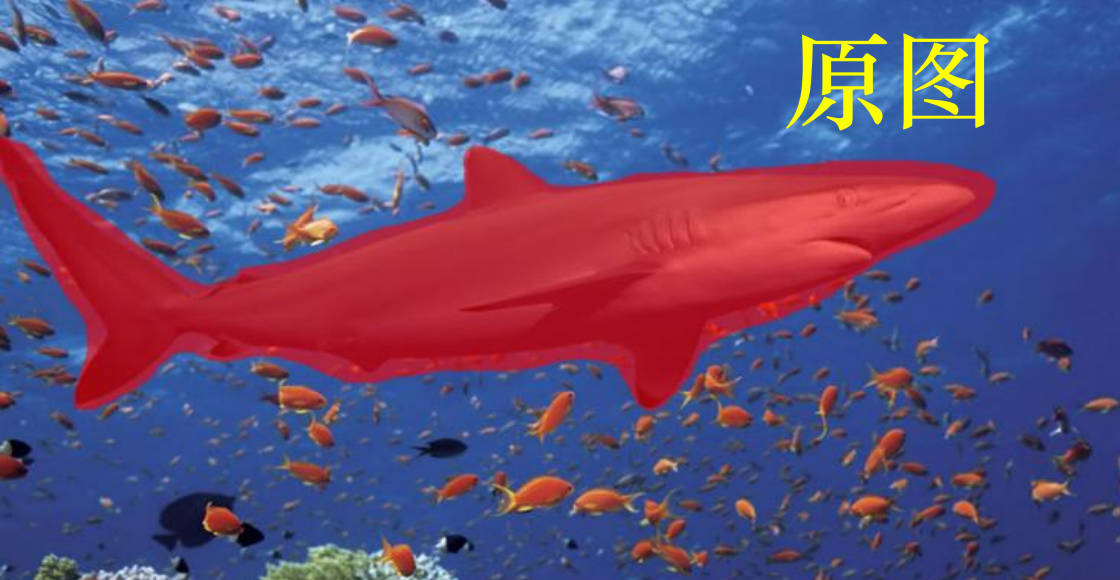
原图



修复结果图







原图



修复结果图



原图



修复后的图





原图



修复后的图



原图



修复后的图





原图



修复后的图





原图



修复后的图





# 原图



# 修复后的图





谢谢各位老师！