游戏画面离不开插图，除底图外对象多为不规则轮廓。直接插入长\*宽大小矩形图片会使背景覆盖底图，因此要想办法扣出对象，只插入轮廓内的图像。



猫身后背景并不需要

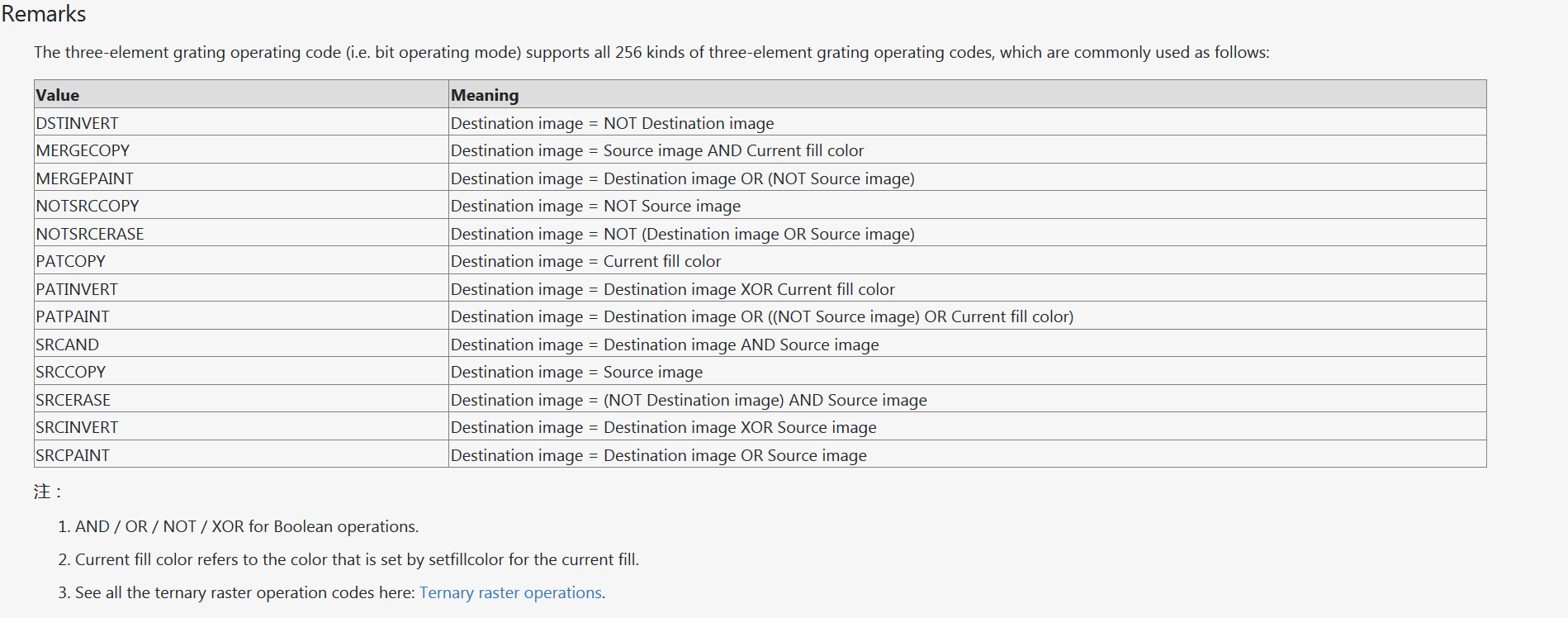
通过PS或手机p图软件，可以得到如下两种裁剪出的猫猫图像（测试需求所以扣得很潦草）



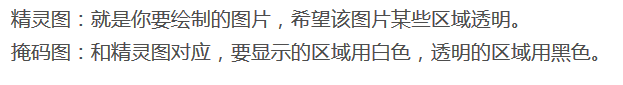
初步想法为使用图1.即可解决仅插入对象画面并去掉多余背景的问题。然而经过测试发现在IMAGE（a, b）建立的a\*b区域内变成了黑色背景，仍然覆盖了草地底图，问题没有解决。这时候猜测easyX的putimage()函数操作流程为删除绘图界面在a\*b内原有图像并插入路径导入的新图像，导致猫猫的背景区域为绘图程序界面的默认黑色（为空，无图层）。

带着疑问回顾putimage()函数，发现了一直在缺省使用的dwRop变量。

通过easyX给出的帮助网页，了解到这个变量，即三元光栅，是可以解决如上问题的关键。



尝试过所有Value后并没有实现目标效果，于是思考逻辑逆运算的组合，以实现背景区域为destination图像、猫猫区域为source图像。思考不出来于是上网搜索并得到了网友点播：核心思路为掩码图与精灵图的重叠显示



我的理解：

掩码图通过黑白判定，替计算机完成需求对象与背景的划分，并通过NOTSRCERASE参数对应逻辑，得到纯黑猫猫对象以及底图草地的反色（255-草地颜色），并以此作为新得到的在a\*b区域上的destination image。程序紧接着putimage精灵图，精灵图的选取应为图2. 类白色背景的扣后图（原因在于此时精灵图作为source image，白色才有运算的意义，可以与此时destination image对应的“底图草地反色”取反，重新得到“底图草地原色”，若完全透明则第二次取反无意义，结果仍为“底图草地反色”）

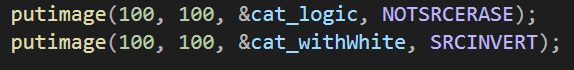


猫猫的掩码图



猫猫的精灵图（蓝色以凸现白背景）

通过



即可实现期望效果



如图所示，

仅猫猫成功插入并保留了除它之外的草地

## 测试源代码

#include <Windows.h>

#include <graphics.h>

#include "CatMouse\_Button.h"

int main(void) {

initgraph(1000, 900);

BeginBatchDraw();

IMAGE map(1000, 900);

loadimage(&map, L".\\pictures\\map\_grassland.jpg", 1000, 900);

IMAGE cat\_withWhite(100, 100);

loadimage(&cat\_withWhite, L".\\pictures\\cat\_withWhite.png", 100, 100);

IMAGE cat\_logic(100, 100);

loadimage(&cat\_logic, L".\\pictures\\cat\_logic.png", 100, 100);

while (1) {

putimage(0, 0, &map);

putimage(100, 100, &cat\_logic, NOTSRCERASE);

putimage(100, 100, &cat\_withWhite, SRCINVERT);

FlushBatchDraw();

}

EndBatchDraw();

Sleep(2000);

closegraph();

}

参考网址： <https://blog.csdn.net/sinat_32588261/article/details/78063174>

以及easyX三元光栅的注释： https://docs.easyx.cn/en-us/putimage