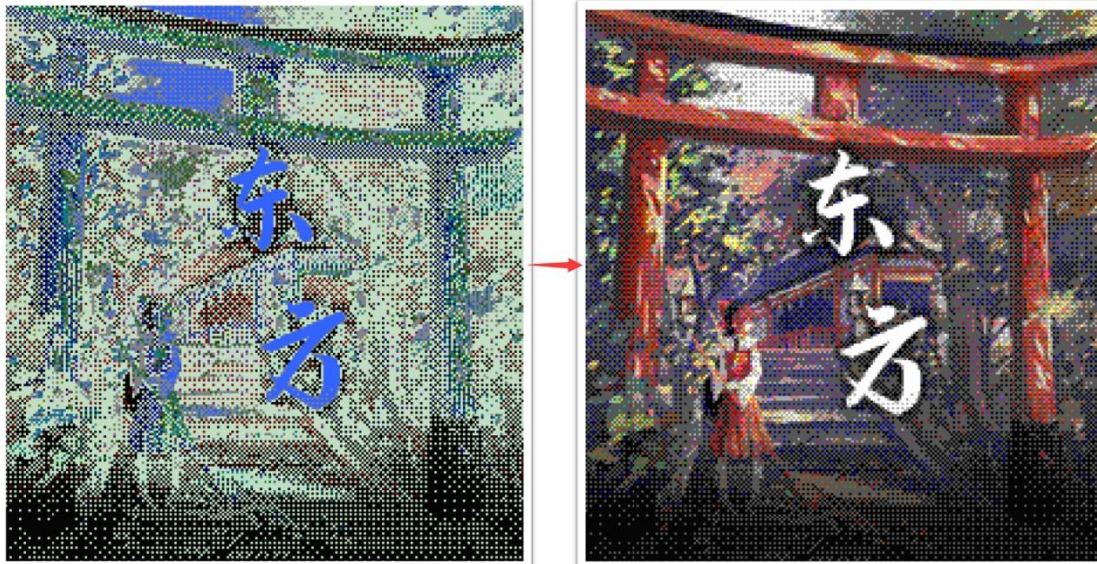


# 直接写屏法的仿色图片显示

## 优点



不言而喻的显示效果良好：

使用的显示模式为最基础的 **VGA 模式**，MED/BI 均可使用；

因为图片源文件只是 16 色的 4 位 BMP，图片大小可以很小；

靠 16 色的拟合，显示效果可以远高于正常 16 色，不过略低于 256 色。~~（远离 dos 阴间调色板的快乐，你值得拥有）~~

可以简单的应用直接写屏，理论在 BC 下可每秒显示出 120 张图片。~~（再也不用等待图片慢悠悠的加载，与老师深情对视）~~

## 食用说明

**注意：**

**本显示机制需要结合 PS 共同食用（降低代码难度）**

源代码文件夹中储存着头文件 IMAGE.h 和源文件 IMAGE.c

正确的将它们放到你工程中相应的位置（如不知道怎么操作，请看祖传文件《如何建立起 BC 工程》一文）之后，在开头 `#include <image.h>`，然后正常调用函数就好。

具体函数如下（可忽略）：

1. `int bmp_convert(char *bmp, char *dbm);`

传入参数：要转化的 bmp 地址，及保存的 dbm 地址

功能：将一个 bmp 文件的完整裸图像数据提取到 dbm 中保存

2. `int show_dbm(int x,int y,char *dbm,int nowpage);`

传入参数: 图片显示的 x, y 坐标, nowpage 为要显示的显示页编号, dbm 传入相应的 dbm 地址

功能: 将图片在所选显示页的对应位置放出

3. `void cir_bar(int x1,int y1,int x2,int y2,int color);`

传入参数: 左上角坐标(x1,y1), 右下角坐标(x2,y2), 颜色 color

功能: 画一个圆角矩形框

4. `void movetopage(int x1, int y1, int x2, int y2,int page1 ,int page2);`

传入参数: 左上角坐标(x1,y1), 右下角坐标(x2,y2), 显示页编号 page1 与 page2

功能: 将对应区域的图像数据, 从 page1 显示页传到 page2 显示页

5. `void set_color(int color_no,int red,int green,int blue);`

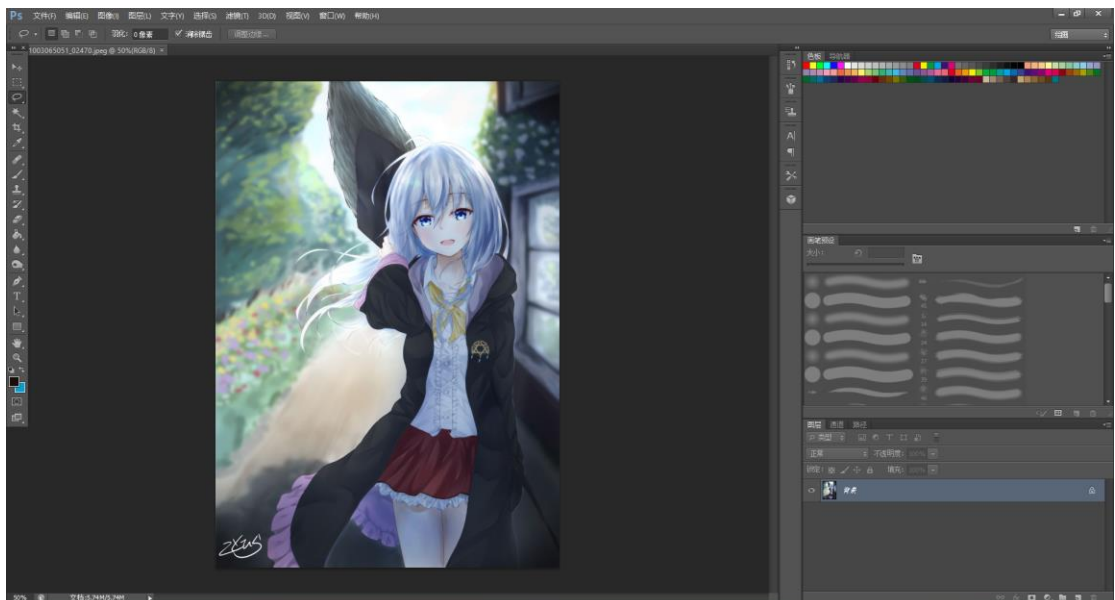
传入参数: 所选颜色号, 红色分量大小, 绿色分量大小, 蓝色分量大小 (分量数值范围在 0-63)

功能: 将所选颜色编号的颜色寄存器中的 RGB 值修改为传入值, 从而改变显示的颜色

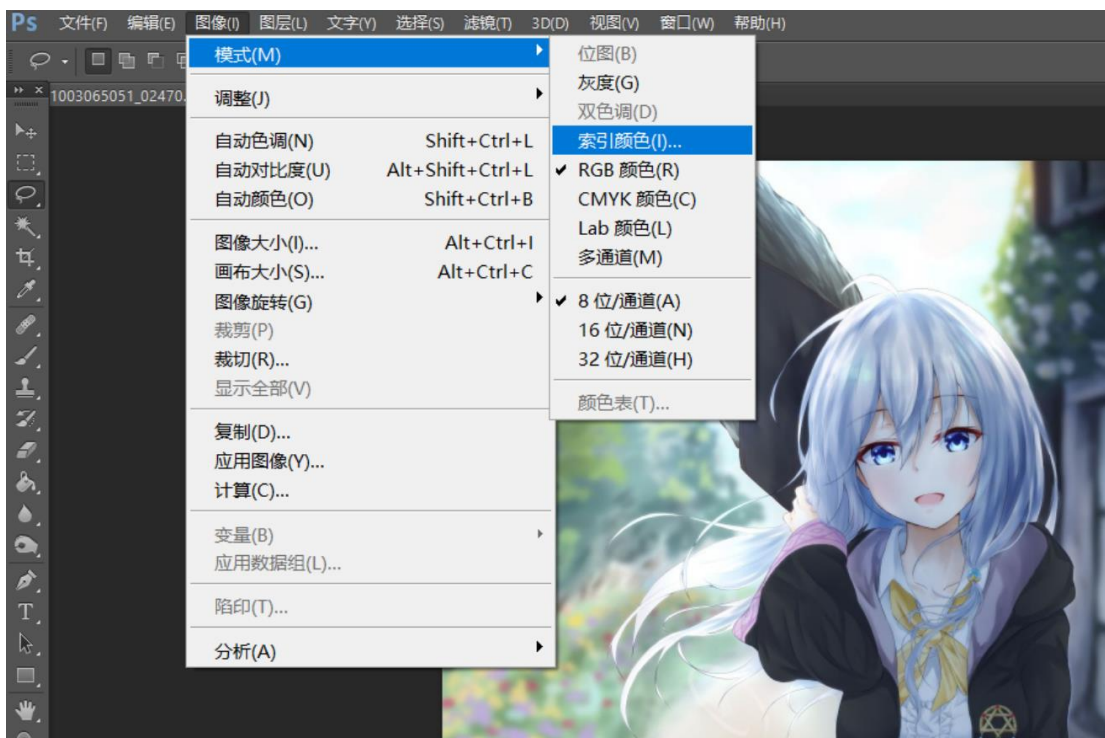
6. 后面四个函数结合起来可以解决闪屏问题, 有兴趣的同学可以自行摸索。

首先打开你的编译器: PS ()

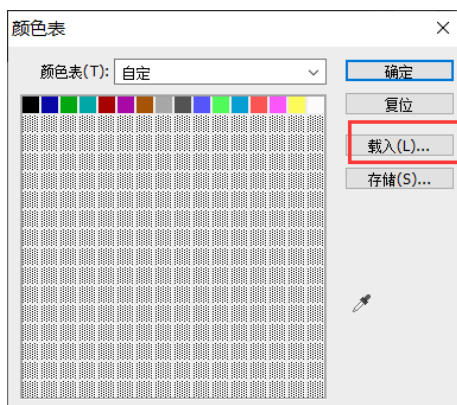
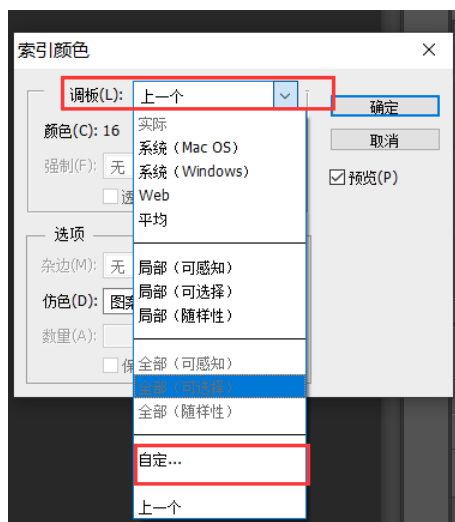
首先打开任何一张图片:



然后修改图片模式为索引颜色



在弹出的对话框中选中自定义的调色板（第一次修改完毕以后，以后再转化直接选择‘上一个’就可）



选择载入的文件名为 dos.act，这是我做的贴近 DOS 调色板颜色的一个颜色表，、在压缩包中，自行解压即可，选中它导入。

名称	修改日期
源代码	2022/1/12 21:38
dos.act	2021/2/16 13:19



仿色设置可以自行选择，会有不同的仿色效果，建议选择 '图案'。

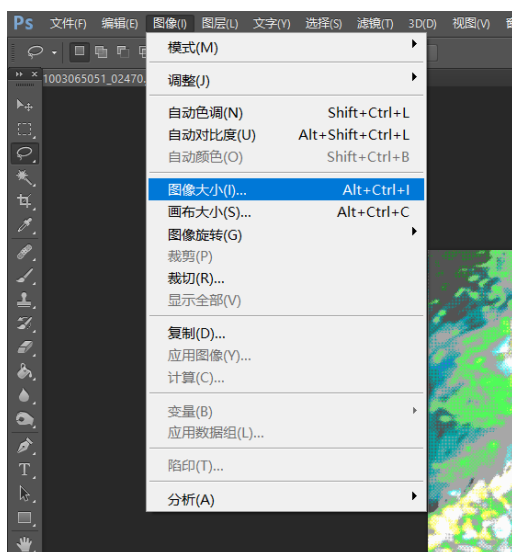


于是图片变成了这样，不同的图片色域不同，拟色效果会有不同，有的效果会更好，有的会更差：

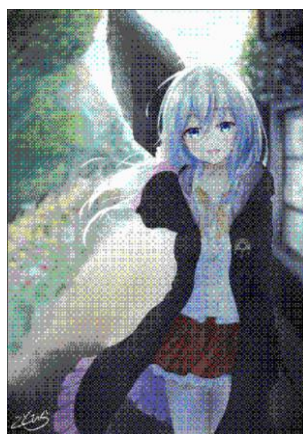




图片的分辨率对其显示效果也有影响，像 VGAMED 分辨率为 640\*350，需要压一下图片的分辨率：

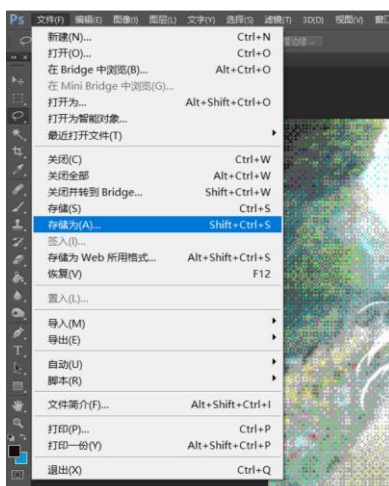


所以实际显示效果受分辨率制约，就会变成这样（此时宽为 350 像素）：



现在上图也就是 BC 中实际能显示出的效果。

下面把它保存为 4 位 BMP 就好了：



选择格式为 BMP:

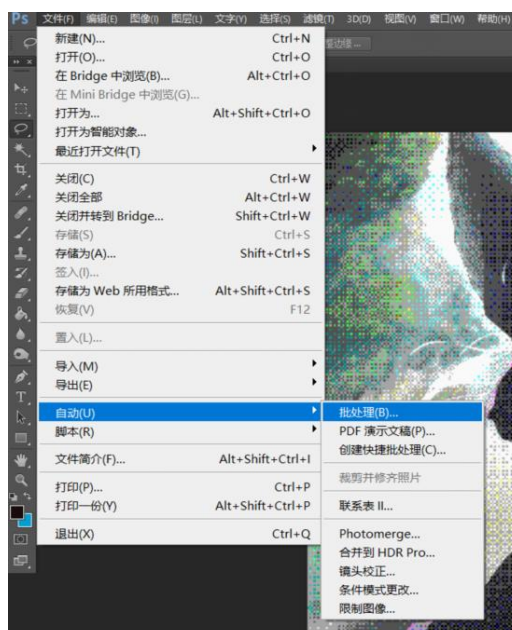
格式(F): BMP (\*.BMP;\*.RLE;\*.DIB) ▼

深度选中为 4 位:



这样就完成了一个仿色成功的 4 位 BMP 制作:

完成以上流程全靠人工相当繁琐, 在相当多的图片需要处理的时候, 可以选择 PS 自带的脚本处理器, 进行批处理工作, 这里就不进行详细解说了。



下面使用程序里的函数对该 BMP 文件进行二次加工。

首先调用 `bmp_convert(char *bmp,char *dbm)`，第一个参数输入 BMP 文件地址，第二个参数输入要生成的 dbm 文件地址。(dbm 是自己瞎定义的图像原始数据，其机理可以忽略)

调用 `show_dbm(int x,int y,char *dbm,int nowpage)`即可在在(x,y)输出图片(x 需为 8 的倍数)，char \* dbm 处填入的 dbm 文件地址。

这样就完成了图片的显示。



啊对了，`set_color(int color_no,int red,int green,int blue)`这个函数，可以指定颜色编号，修改其对应的 rgb 值，即实现自由选择颜色的功能，让软件界面色彩更加丰富。具体实现机理可以参考《C 语言的窗口式图形界面设计》