

**数字媒体技术**

**实验报告**

题 目 编解码平台及应用

学 院 计算机科学与技术

专 业 大数据

学 号 L170300901

学 生 卢兑玧

任 课 教 师

哈尔滨工业大学计算机科学与技术学院

2021.3

**实验三:** **编解码平台及应用**

1. **实验内容或者文献情况介绍**

跨媒体的情感分类

– 根据图像、视频、音频，分别提取其特征，根据特征对其表达的情感进行分类

– 是否可以在各种多媒体的特征表示之间建立转换关系，从而可以用图像特征来检索具有同样情感属性的音频和视频，或者用任意一种表达其属性。

– 情感属性可以换为：正能量性，积极性等，并进一步给出定量分数

要求

– 最后提交实验报告（包括原理介绍、论 文阅读、测试数据集和测试结果等）和 源代码以及可执行程序

– 具体参见模板。

-对真彩色BMP图像进行颜色量化，将颜色数目减少为 256种颜色

• GIF图像都是通过颜色量化生成，然后比较256种颜色图像与原始图 像之间的PSNR值

– 对二值图像进行JBIG压缩(专门针对二值图像)

• 尝试运行JBIG代码进行编解码：

<https://www.cl.cam.ac.uk/~mgk25/jbigkit/>

• 将可执行程序的操作说明写清楚！

– 使用FFMPEG平台对视频进行编码、转码操作

• 访问ffmpeg官网，学会使用ffmpeg进行视频操作

对真彩色BMP图像进行颜色量化，将颜色数目减少为 256种颜色

BMP(\*BMP, \*.RLE)：

是指在DOS和视窗兼容电脑上使用的"标准视窗Bitmap"图像。 该BMP文件格式用于视窗的基础画面,支持RGB模式和Indexed模式。

初期BMP文件没有压缩,但自Windows 3.0之后,可以使用RLE压缩方式压缩文件。 BMP文件有黑白和4,8,24比特格式,分别表现2色,16色,256色和特鲁色。 乐视和阿尔法频道不支持。

Compuserve GIF:

它是GrahicsInterchange Forbat 的缩写, 其设计是为了尽量减少文件传输时间。 以Betwap方式的压缩形式,是专门支援8Bet256颜色的Indexed模式的文件格式,在网页制作中经常使用。

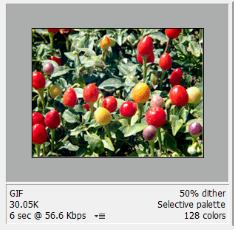
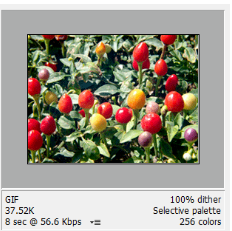
GIF文件有两种类型,根据编码方法分为87A和89A。 89A是最新版本,将多张图片合二为一,可以产生像动一样的效果,多用于在网络及多媒体环境下实现简单的视频。 (potshop支持GIF格式,但不支持视频GIF文件。)

GIF根据显示图像的方式分为Non-Interlaced和Interlaced方式,共有4种方式可以储存。

Graphics Interchange Format

GIF也是为加快传输速度而压缩存储的方式之一,它的特点在于在웹上多用,可以在图像文件内将多张图像放在一个文件中。

与JPG文件相比,压缩率稍差,而传输速度快,图像损伤小。遗憾的是GIF最多只能表现256种颜色。



<根据压缩率GIF文件的画质变化>

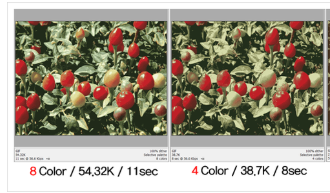
GIF文件容量控制色数和dither值。

dither价格是柔和表现颜色和颜色界线的数据。

随着颜色数量的减少,dither价格下降,画质会变粗糙,但容量会减少。

保存GIF时,选项上可选择256 / 128 / 64 / 32 / 16 / 8 / 4 / 2

像单纯的插图一样,如果规定颜色数量,根据颜色数量指定的话,容量就会大幅减少。

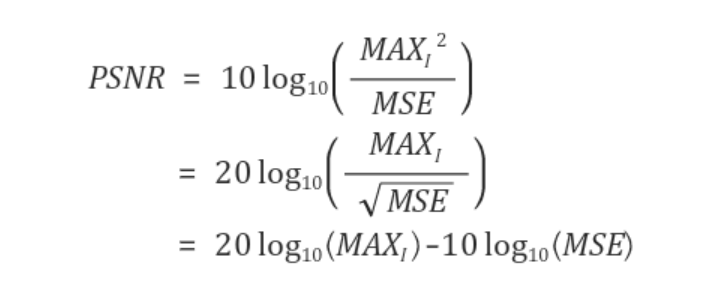


很好奇一下子把照片图片缩小到极端颜色的数量会怎么样,所以试了一次。

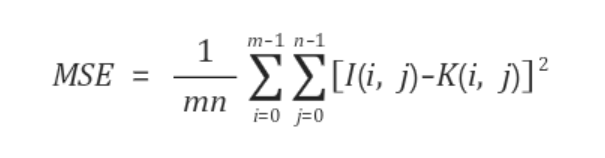
dither价格保持不变,只减少了颜色数,结果就出来了。

PSNR

PSNR计算如下:



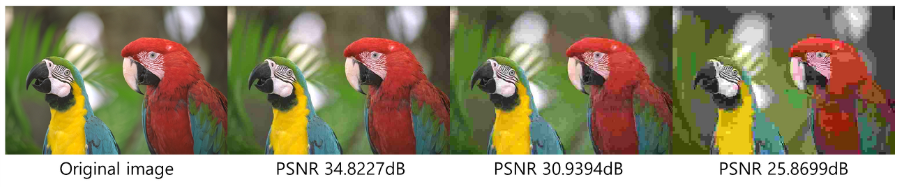
这里 MAXI太阳的后裔是该影像的最低价,8位数的灰色影像是255。 另外,MSE的平均误差可以计算如下。



在这里,I是m x n尺寸的灰石形象,K是指I中包含杂音的形象,即歪曲的形象。 由于官方1的分母中有MSE,所以MSE(平均平方差)越小,就拥有越大的PSNR。 因此,品质好的形象会拥有较大的PSNR,品质不好的形象会拥有较小的PSNR。

PSNR与图像质量之间的关系

让我们来寻找一下在同一原图中被不同程度JPEG压缩损失的图像的PSNR。



从JPEG压缩损失图片的PSNR可以看出,压缩率越大,PSNR也越小。

34.8227dB -> 30.9394dB -> 25.8699dB

我们可以亲眼看到,越是往右,歪曲现象就越严重。

34.8227dB (质量不错) -> 30.9394dB (质量差一点) -> 25.8699dB(质量差一点)

1. **算法简介及其实现细节**

对二值图像进行JBIG压缩

JBIG是为了压缩李真的形象,特别是Facsimily的形象而开发的。 但是它也用于其他用途。 比组合4(packs)标准提高了20%甚至50%。 在特殊情况下,压缩率提升了30倍。 1993年规格化。

JBIG采用IBM专利的一种算术编码方式。 这一方式被命名为"Q-coder"。 这一方式就是以前面的bit或画的前排bit计算概率。 根据扫描顺序,需要发生压缩或压缩动作,因此后面的螺栓不能参照压缩(reference)。 增加若干超大头(5%),可支持有序传输功能。

JBIG采用的好处如下:

对Bitonal及Grayscale形象卓越的无损失压缩功能

越小的文件越快速的亿岁及 Retrieve功能

对格雷斯凯尔形象的更有效的压缩功能

具有在更高的分辨率上存储压缩图像的能力,故OCR认识具有更好的准确度

Progressive Capability - 逐渐将代码图像进行装饰时,原图像的阻隔上图版本将首先打开,然后加倍分辨率进行装饰。

Progressive Encoding -

在拥有一个数据库的应用程序中,可有效构建具有不同分辨率的Serve输出功能。另外,还可以开发低速和高速通讯链接的图像浏览器应用程序。 即, 低分辨率的局域网可以高速传输或显示,相对高分辨率的局域网可以低速链接。

JBIG的主要用途是替代CCITT Group 3/4中使用的低效压缩算法。

对所打印文本或扫描Line Art图片,JBIG的压缩率比Group4的压缩率高出10至数十个百分点。

电脑生成的图像打印文本比Group4更出色500%。

使用Grayscale对图片进行明暗处理或去污处理,图像质量提高2~3倍。

JBIG对NOIZ非常敏感,在阁下的形象内NOIZ增加时可减少压缩率。

JBIG可代替无损失JPEG使用。 JBIG在每像素2~5Bit人的形象中,创造出比无损失JPEG更高的品质,对拥有6~8像素数据的形象产生相同结果。

更小20~60%的文件可以节省费用及快速传送文件。

JBIG的使用领域:

Document Imaging:基于纸张的电子扫描,存储和管理

存取账户

网络应用或所有网络应用

A0大小的工学纪录片储存。由于可使用多重激光,使用者可提前查看障碍商图,在浏览器上显示自己想要的分辨率。

JPEG压缩的基本原理

– 将视觉上重要的部分和不太重要的部分分离(DCT变换)

– 把重要部分活下来,把不重要的部分损失掉

减少数据量(量化)

– 通过颜色型号的变换,分为重要部分和不重要部分

李(RGB->YIQ)

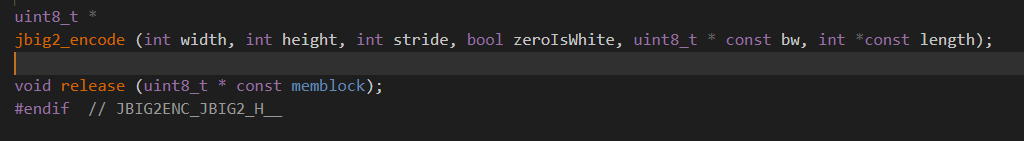
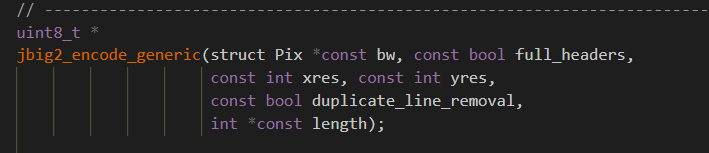
327KB(335,684byte) 44.9KB(46,024byte)

视频放大缩小字体功能 放大缩小字体功能

导出函数

DLL 库导出的函数有三个（除下列方法之外，还有原代码提供的jbig2\_encode\_generic方法：



jbig2\_encode 方法用于编码传入的字节数组。其参数有六个：

1. Width：图像宽度

2. Height：图像高度

3. Stride：一行像素所占的双字(DWORD)数

4. ZeroIsWhite：将BW的0视为白色

5. BW：图像的字节数组（Byte[]），此字节数组是原始黑白图像的二进制数据，默认情况下，二进制位1表示白色，0表示黑色

6. Length：字节数组长度

7. 传出参数：编码后的 JBIG2 字节数组

1. **实验设置及结果分析（包括实验数据集）**

使用FFMPEG平台对视频进行编码、转码操作

1.Demuxing

:从文件(DCF)中提取录像,音频数据流

2. deconding

:使用特定视频光盘获取原始视频数据

3. Encoding

:从原视频数据转换成特定视频光盘

2. 墨星(Muxing)

:将转换的数据(录像)装入文件(DCF)

基本FFmpeg使用方法(命令语)

ffmpge《输入选项》,《输入文件》,《打印选项》,《打印文件》

输入选项

- 指定输入文件:-i input.ts

- 强制指定形式:-f mkv

ex) ffmpeg -i input.ts -c:v libx264 output.mkv

软件编码

-c:v libx264 (H.264)

-c:v libx265 (H.265, HEVC)

参考。利用FFmpeg剪视频

ex) ffmpeg -i input.avi -ss 600 -t 120 -vcodec copy -acodec copy clip.avi

-ss 10分钟(600秒)开始部分文件

-t 2分钟(120秒)左右选择部分

将10分钟的avi文件分成一半的时候

ffmpeg -i input.avi -vcodec copy -acodec copy -t 300 before.avi

-t 300(秒)指5分钟之前没有指定起点,所以从开始到5分钟为止。

ffmpeg -i input.avi -vcodec copy -acodec copy -ss 300 after.avi

-ss 300(秒)5分钟开始的意思,因为结尾部分没有指定,所以从5分钟开始一直到最后。

参考。利用FFmpeg的视频合成

把合页视频的名单制作成文本文件。

$ cat > list.txt

file a.mkv

file b.mkv

$ ffmpeg -f concat -i list.txt -c copy output.mkv

ffmpeg -i <source>.mp4 -c:a copy -c:v copy <target>.mkv

-i是为指定输入文件而设定的,-c是设定codeo,但设定video时是v,设定audio时是a。 一般会指定codex的名字,会使用原样的codex,所以交给copy就可以了。 如果在目的文件名称上使用mkv,可以自动转换成mkv形式。