

****

信息学院软件工程系

《计算机网络》实验报告

**题　　目 实验一　息的软件**

**班　　级 软件工程2020级卓越班**

**姓　　名 庾晓萍**

**学　　号 20420192201952**

**实验时间 2022年3月2日**

**2022年3月2日**

填写说明

1. 本文件为Word模板文件，建议使用Microsoft Word 2019打开，在可填写的区域中如实填写；
2. 填表时勿破坏排版，勿修改字体字号，打印成PDF文件提交；
3. 文件总大小尽量控制在1MB以下，最大勿超过5MB；
4. 应将材料清单上传在代码托管平台上；
5. 在实验课结束14天内，按原文件发送至课程FTP指定位置。

# 实验目的

通过完成实验，掌握物理层传输的原理；了解传输过程中的编解码、噪声、分辨率、波特率、调制和误码等通信概念；理解奈氏定理和香农定理。

# 实验环境

操作系统：Windows 10；

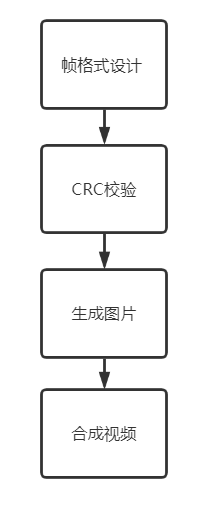
IDE：Visual Studio 2019

编程语言：C++；

# 实验结果

一、编码

0、主要流程

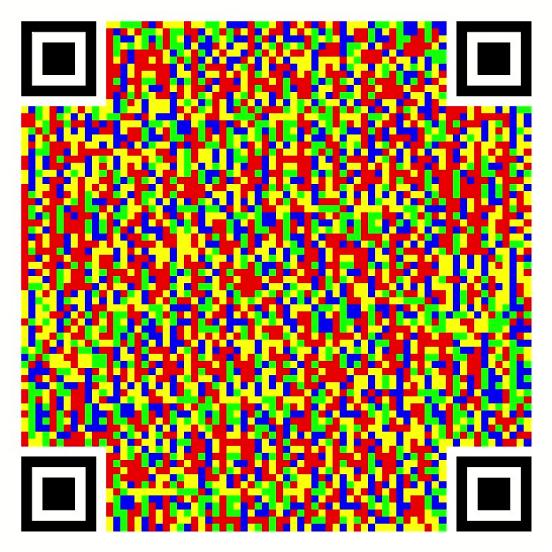


1、帧格式设计

（1）图片采用彩色编码，通过红、黄、蓝、绿四种颜色表示比特信息。（黄表示00，红色表示01……）。

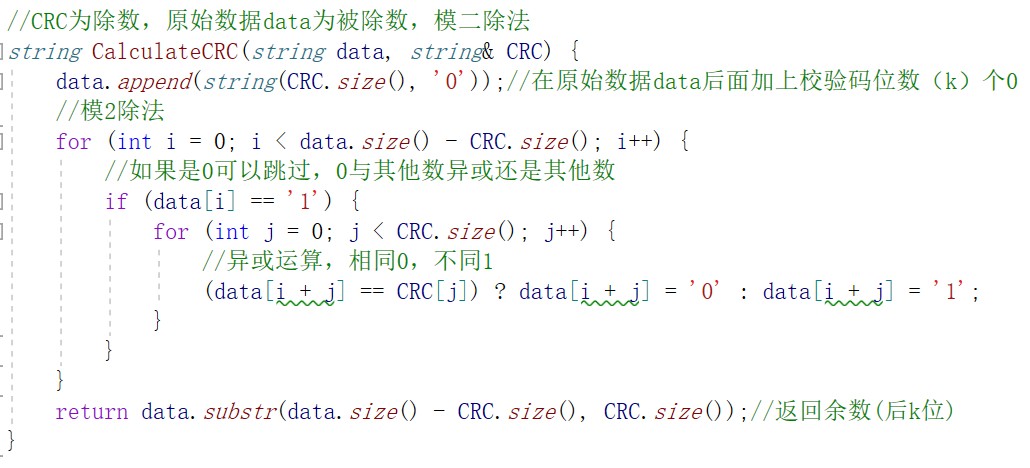
（2）二维码包含四个定位矩形，其他区域为数据区域。每张二维码包含5184（722）个像素格，其中576（4\*122）格被定位矩形占据。也就是说数据区的像素格为4608格，每格传递2bits信息，其中有32bits是最后加上的0。故每张二维码的信息为：2\*4608-32=9184bits。

（3）默认帧率为10帧/秒，可以在Main.cpp中更改。

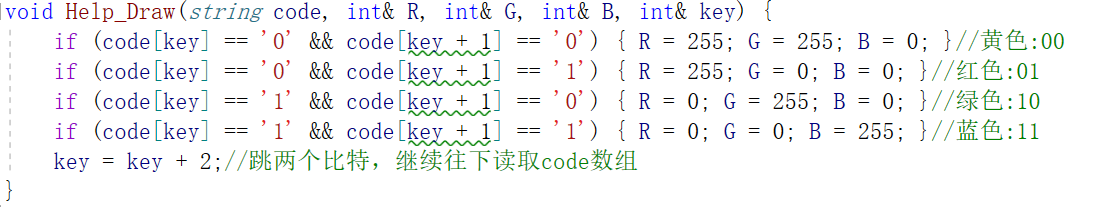


1. CRC效验核心代码

使用模二除法，算法类似高精度除法，但是不进位，每步的运算是异或。

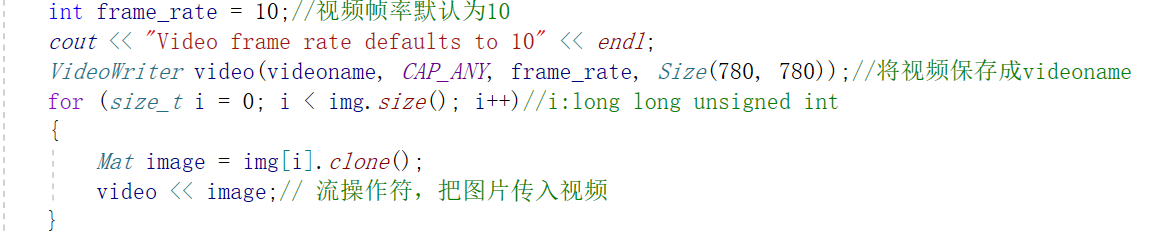


1. 图片生成核心代码

使用opencv的rectangle函数涂色。每两个比特信息填成一种颜色。 

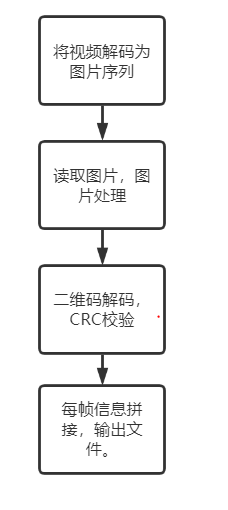
1. 将生成的图片序列合成为视频

使用opencv的VideoWriter函数，将视频保存在videoname路径。



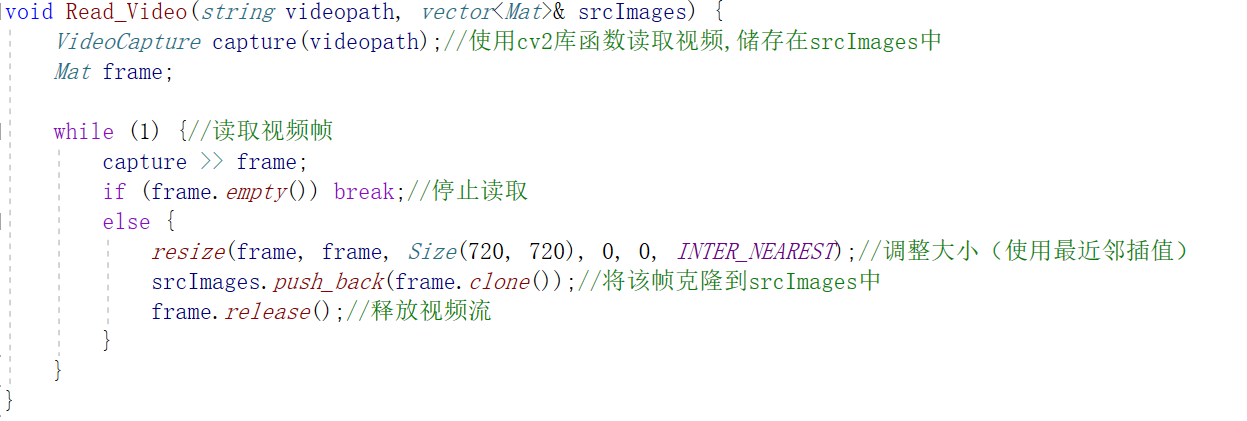
1. 解码

0、主要流程



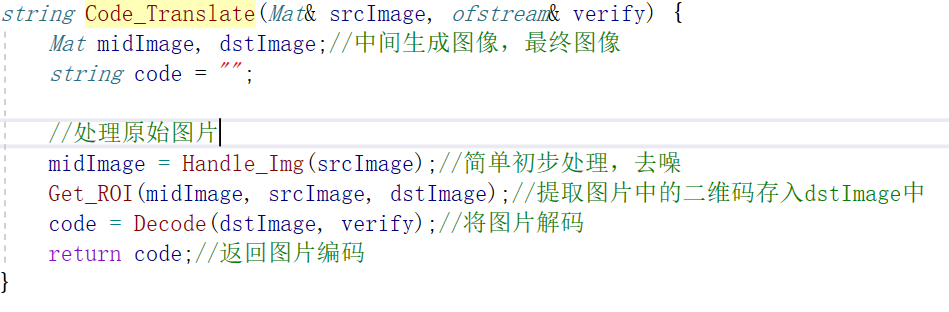
1、将视频解码为图片序列

使用opencv的VideoCapture函数，将得到的图片序列储存在srcImages中。



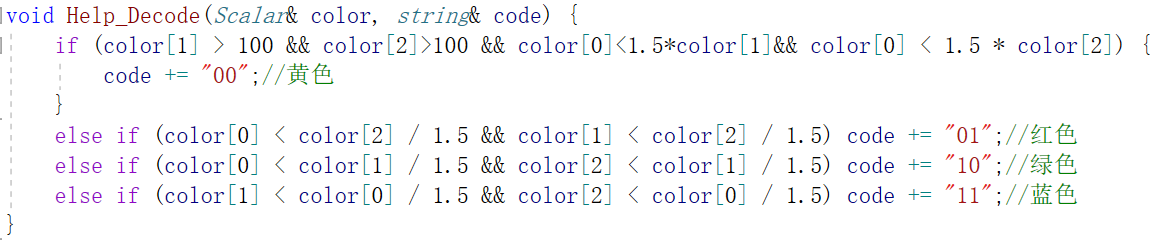
2、读取图片，图片处理

先进行简单处理，去除部分噪声（彩色转灰度、高斯滤波、二值化。）



1. 二维码解码，CRC校验

将识别到的颜色转化为对应的比特编码。



1. 将每帧信息拼接，输出二进制文件与校验文件。

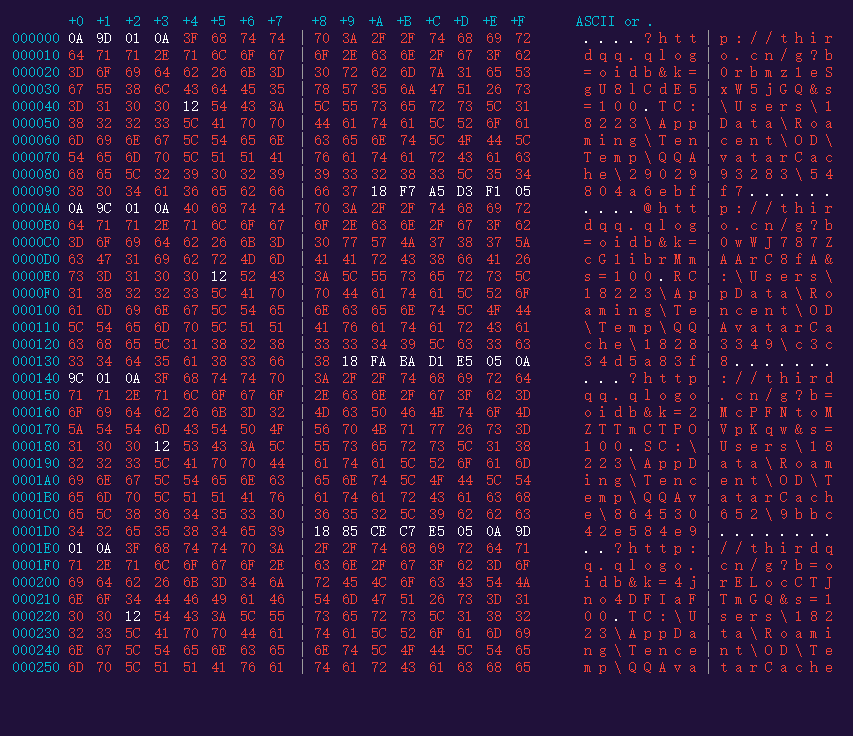


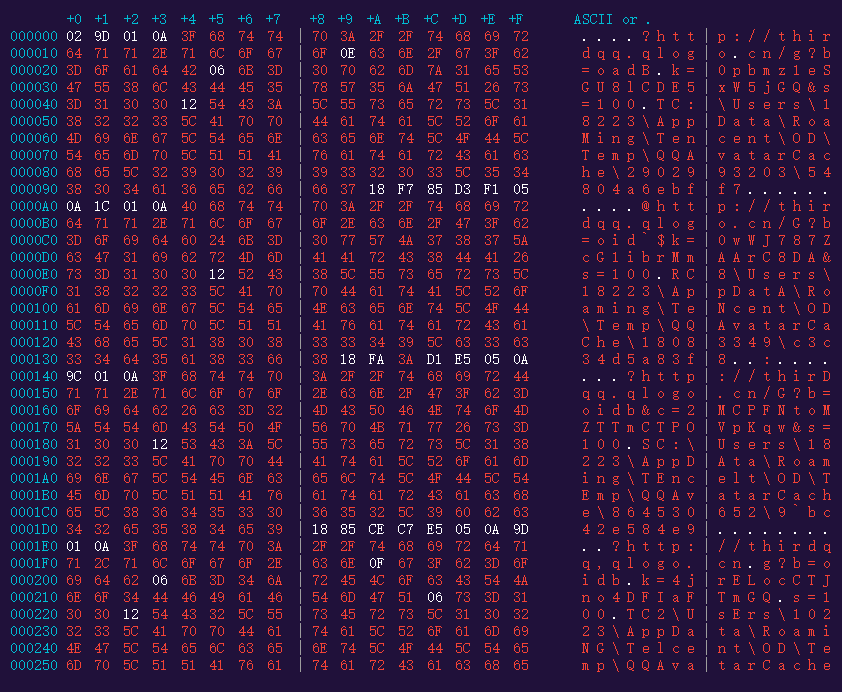
1. 实验结果
2. 传输速度

实验中，彩色二维码的分辨率是5184（722）个像素方块。帧率10帧/秒，每张二维码包含9184 bits信息。故波特率以及传输速率为10\*9184\*10-3=91.94Kb/s。

1. 准确度

下面第一幅图为输入的二进制文件，第二幅图为输出的二进制文件，信息大致可以完整传输，但是仍然有传输错误的部分。





1. 输出文件说明

在附加文件包“EA1-1952-庾晓萍”中，包含了附加的文件。其中input.bin 是输入的二进制文件，output.mp4是输出的彩色二维码视频文件， byphone.mp4是手机拍摄下的视频，output.bin是由手机拍摄的视频解码 得到的二进制文件，verify.bin是作为校验的二进制文件。

# 实验代码

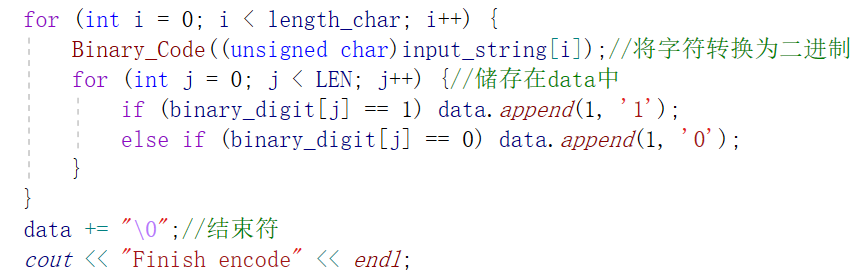
本次实验的代码已上传于以下代码仓库：

https://github.com/ryanregal/ExpOne\_ComputerNetwork

# 实验总结

一、实验课后问题

1、在实验中的，编解码算法是什么？

编码算法：核心部分时将字符转化为比特，具体如下图。先将文件内容读入字 符数组input\_string中，将字符转换为二进制储存在字符串data中。

解码算法：解码算法先调用Read\_Video函数读取视频，将图片储存在srcImages 数组中。再调用Code\_Translate函数将图片转换为代码。

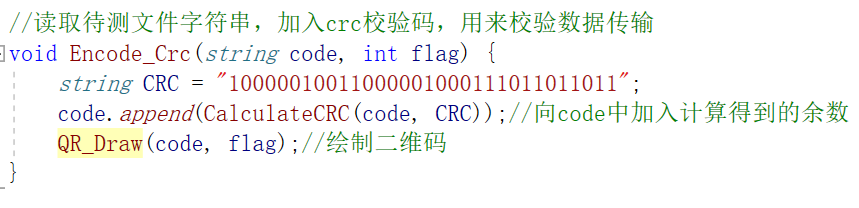


2、在实验中的，调制和解调算法是什么？其中，载体信号、调制信号是什

么？使用的算法属于调频、调幅还是调相？

答：调制是将信息搭载到载体上，本次实验中，就是把二进制文件写到一张图 片上，调制算法对应绘制二维码数据区的部分。在本次实验中，解调就是从二 维码中获取数据，得到二进制文件。载体信号是二维码，调制信号是从文件得 到的比特流。在实验中，是通过可见光的不同频率（像素块的不同颜色）来表 达不同信息的，所以算法属于调频算法。

调制算法：调制算法将比特流转化为二维码，主要是在比特流末尾加上CRC 校验码，然后调用绘制二维码。调制算法具体如下图：



解调算法：解调算法先获取图片，获取图片后进行灰度处理、二值化，然后根 据定位码定位，校正图片，最后得到一张二维码的数据区数据。解调算法在 Code\_Translate(Mat& srcImage, ofstream& verify)，具体如下图：



3、在实验中的，主要的噪声强度有多大，噪声来自哪些因素？

主要的噪声来自拍摄视频时手机的色差，背景的光线，以及拍摄图片变形。

4、你的编码算法分辨率是多少？（作为实验，不要求分辨率太大）

实验中，彩色二维码的分辨率是每张图5184（722\*722）个像素。

5、你的编码波特率是多少？传输率是多少？

默认帧率10帧/秒，每张二维码包含9184 bits信息。故波特率以及传输速率为 10\*9184\*10-3=91.94Kb/s。

6、按奈氏定理和香农定理，通信率上限是多少？

319.2Kb /s

1. 实验思考与反思

通过这一次的计网实验，学习到了很多关于通信的知识，在实验过程中遇到了一些问题，首先是opencv库配置问题，出现了dll报错，最后发现是链接器配置问题。然后是像素块的颜色判断，识别手机拍摄视频时出现了错误，没有成功编码，最后修改了RPG的判断条件。还有在对图片处理时，没有采用掩模的，可能导致背景图片对解码的干扰，可以再对代码进行改进。

另外也和老师讨论了二维码的定位矩形问题。实验中采用了四个定位矩形。四个定位矩形的优点在于可以对变形的拍摄图形更好地处理，代码中用opencv的getPerspectiveTransform函数对图片进行了透视变换。但是四个定位矩形也存在不能识别二维码方向的问题，实际中会在右下角的中间用一个小矩形替代。