**江苏科技大学**

**实 验 报 告**

课 程： 信息隐藏与数字水印

课 题： WAV音频的LSB信息隐藏

学 院： 计算机学院

组 员： 182210710119 陈四贵

182210710101 曹慧珺

182210710104 樊静雯

182210710140 赵富星

班 级： 1822107101

指导老师： 李会格

目 录

[一、 实验目的 1](#_Toc74923623)

[二、 实验环境 1](#_Toc74923624)

[三、 实验原理 1](#_Toc74923625)

[四、 实验步骤 1](#_Toc74923626)

[五、 实验截图 4](#_Toc74923627)

[六、 实验分析暨客观评价 7](#_Toc74923628)

[七、 心得体会 7](#_Toc74923629)

[八、 实验分工介绍 7](#_Toc74923630)

# 实验目的

了解WAV文件格式，掌握利用WAV格式音频文件实现LSB信息隐藏原理，设计并实现一种基于WAV文件的LSB信息隐藏算法。了解归一化相关系数NC的概念，利用NC对水印图像和提取图像水印进行比较。

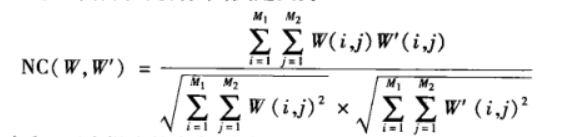
# 实验环境

软件环境：Windows PC机一台、 matlab R2018a

图像参数：512\*384 BMP

# 实验原理

WAV是 Microsoft Windows本身提供的音频格式,该格式通常保存一些没有压缩的音频。对于数字音频,其最低位或者最低几位的改变,对于整个声音没有明显的影响,因此替换掉这些不重要的部分,可以隐藏秘密信息。

如果在音频信号中嵌人的水印为图像,则评价提取的水印与原始水印的相似性,可采用归一化相关系数(NC)作为评价标准,其定义为

式中:W为原始水印,w'为提取的水印,它们的大小为M× M2o

# 实验步骤

①在WAV音频中隐藏一组随机生成的二进制数字

1. clc;
2. clear;
3. fid = fopen('1.wav', 'r');
4. oa = fread(fid, inf, 'uint8');
5. fclose(fid);
6. n =5;
7. d = randsrc(1,n,[1 0]); %产生随即序列即水印
8. M = oa;
9. **for** i=45:45+n-1
10. M(i) = bitset(M(i),1 ,d(i-44));
11. end
13. fid = fopen('music\_new\_secret.wav','wb');
14. fwrite(fid,M ,'uint8');
15. fclose(fid);
16. figure;
17. [y1,Fs] = audioread('1.wav');
18. x1 = (0:length(y1) - 1)/Fs;
19. subplot(2, 1, 1);
20. plot(x1, y1);
21. axis([0 max(x1) -1 1]);
22. xlabel('Time /(s)');ylabel('Amplitude');title('original audio');
24. [y2,Fs] = audioread('music\_new\_secret.wav');
25. x2 =(0:length(y2)-1)/Fs;
26. subplot(2, 1, 2);
27. plot(x2, y2);
28. axis([0 max(x2) -1 1]);
29. xlabel('Time / (s)');ylabel('Amplitude');title('information hiding audio');
30. figure;
31. subplot(2, 1, 1); plot(x1, y1); axis([0 0.6 -0.05 0.05]);
32. xlabel('Time / (s)');ylabel('Amplitude');title('original audio');
33. subplot(2, 1, 2); plot(x2, y2); axis([0 0.6 -0.05 0.05]);
34. xlabel('Time / (s)');ylabel('Amplitude');title('information hiding audio');
35. fprintf('嵌入的水印：');

②提取隐藏信息

1. fid = fopen('music\_new\_secret.wav', 'r');
2. oa = fread(fid, inf, 'uint8');
4. m = oa;
5. n = length(oa) - 44;
6. **for** i = 1 : n
7. w(i) = bitget(m(44 + i), 1);
8. end
9. fclose(fid);
10. disp(w)  %将提取到的信息打印出来
11. rate = length(find(w ~= d)) / length(d); %rate表示为误码率

③在WAV音频中隐藏一个二值图像

1. clc;
2. clear;
3. fid = fopen('hidden\_music.wav', 'r');
4. oa = fread(fid, inf, 'uint8');
5. fclose(fid);
7. n = length(oa) - 44;
8. io = imread('hidden\_image.bmp');
9. [row, col] = size(io);
10. wi = io(:);
11. **if** row \* col > n
12. error('音频载体太小，请更换载体');
13. end
14. info\_hd\_audio = oa;
15. **for** k = 1 : row \* col
16. info\_hd\_audio(44 + k) = bitset(info\_hd\_audio(44 + k), 1, wi(k));
17. end
19. fid = fopen('secret\_music.wav', 'wb');
20. fwrite(fid, info\_hd\_audio, 'uint8');
21. fclose(fid);
23. figure;
24. [y1, Fs] = audioread('hidden\_music.wav');
25. x1 = (0:length(y1) - 1) / Fs;
26. subplot(2, 1, 1);
27. plot(x1, y1);
28. axis([0 max(x1) -1 1]);
29. xlabel('Time / (s)');ylabel('Amplitude');title('original audio');
31. [y2, Fs] = audioread('secret\_music.wav');
32. x2 = (0:length(y2) - 1) / Fs;
33. subplot(2, 1, 2);
34. plot(x2, y2);
35. axis([0 max(x2) -1 1]);
36. xlabel('Time / (s)');ylabel('Amplitude');title('information hiding audio');

39. figure;
40. subplot(2, 1, 1); plot(x1, y1); axis([0 0.6 -0.05 0.05]);
41. xlabel('Time / (s)');ylabel('Amplitude');title('original audio');
43. subplot(2, 1, 2); plot(x2, y2); axis([0 0.6 -0.05 0.05]);
44. xlabel('Time / (s)');ylabel('Amplitude');title('information hiding audio');

④提取WAV音频中隐藏的图像信息

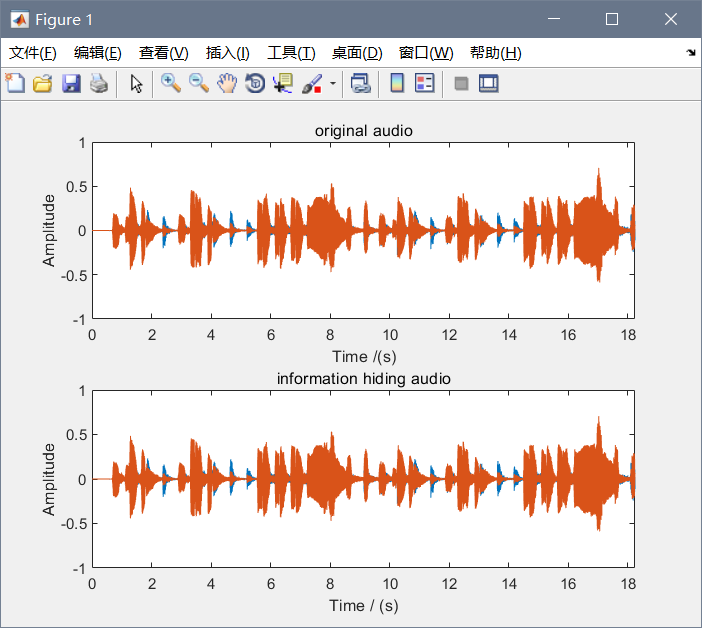
1. clc;
2. clear;
3. orig\_pic = imread('hidden\_image.bmp');

6. fid = fopen('secret\_music.wav', 'r');
7. oa1 = fread(fid, inf, 'uint8');
8. fclose(fid);
9. n1 = 40000;
10. **for** i = 1 : n1
11. w1(i) = bitget(oa1(44 + i), 1);
12. end
13. w1 = w1';
14. m1 = reshape(w1, 200, 200);
15. figure;
16. subplot(1, 2, 1);imshow(orig\_pic);title('original picture');
17. subplot(1, 2, 2);imshow(m1);title('not attacked extracted picture');
18. imwrite(m1, 'school\_extract.bmp', 'bmp');

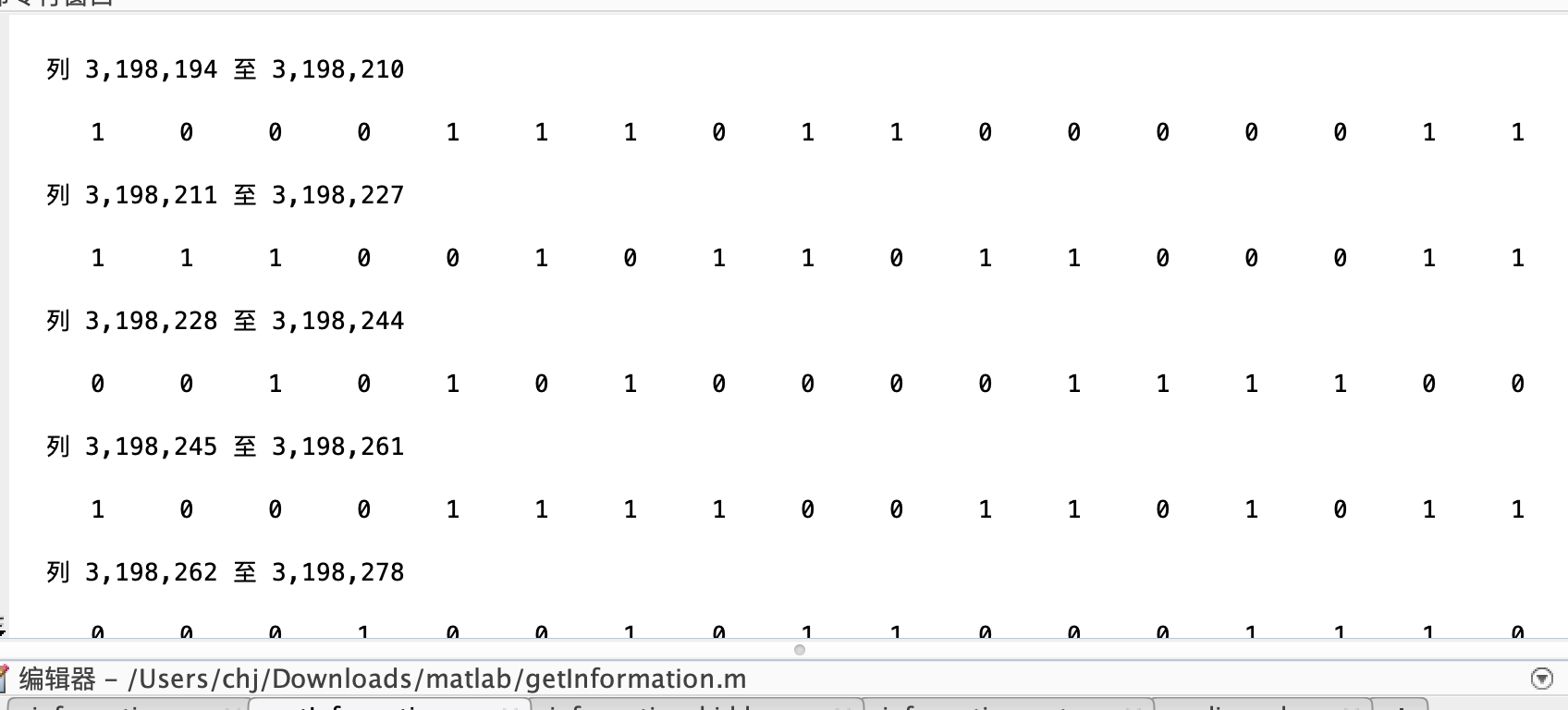
21. fid = fopen('music\_new\_secret\_attacked.wav', 'r');
22. oa2 = fread(fid, inf, 'uint8');
23. fclose(fid)
24. n2 = 40000;
25. **for** i = 1 : n2
26. w2(i) = bitget(oa2(44 + i), 1);
27. end
28. w2 = w2';
29. m2 = reshape(w2, 200, 200);
30. figure;
31. subplot(1, 2, 1);imshow(orig\_pic);title('original picture');
32. subplot(1, 2, 2);imshow(m2);title('attacked extracted picture');
33. imwrite(m2, 'school\_extract\_attacked.bmp', 'bmp');

# 实验截图

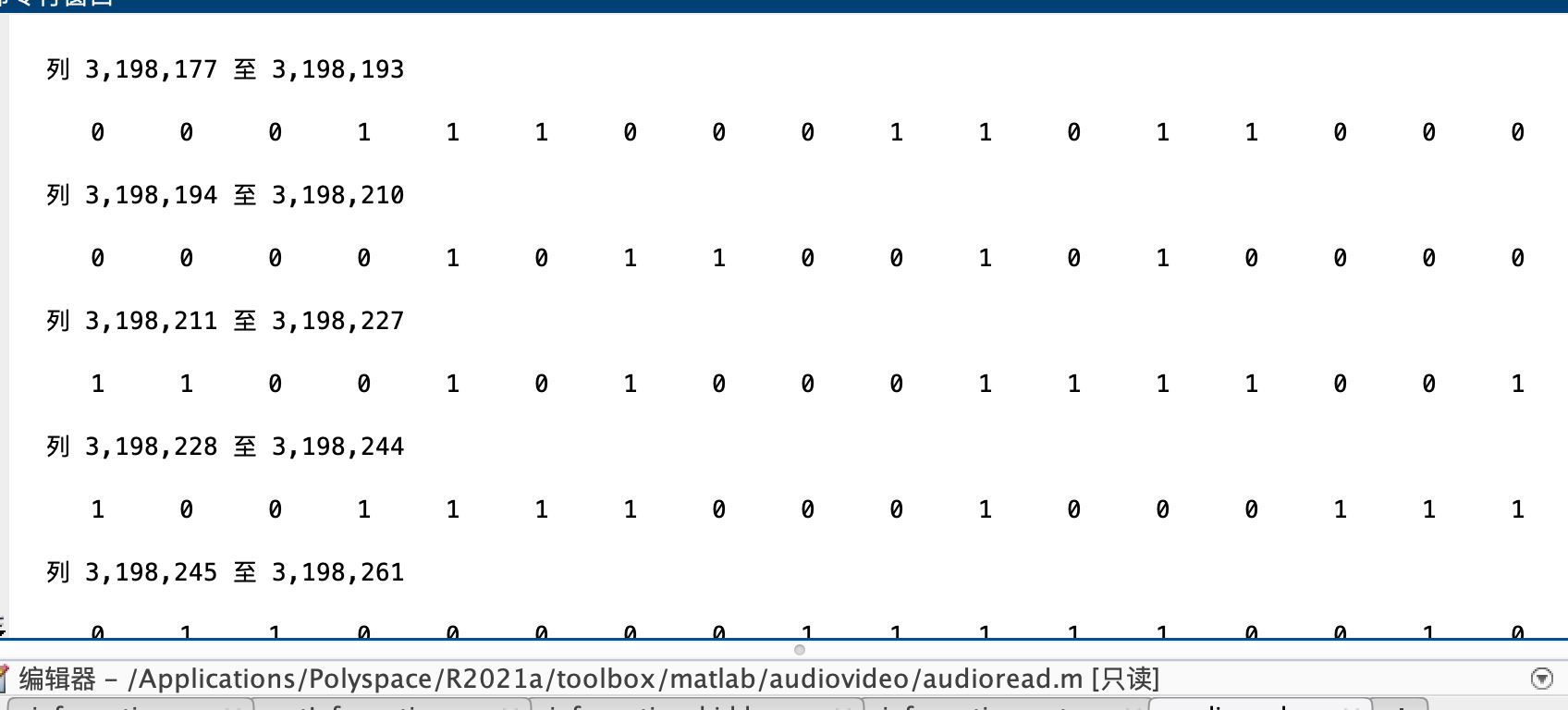
1、生成的波形图：



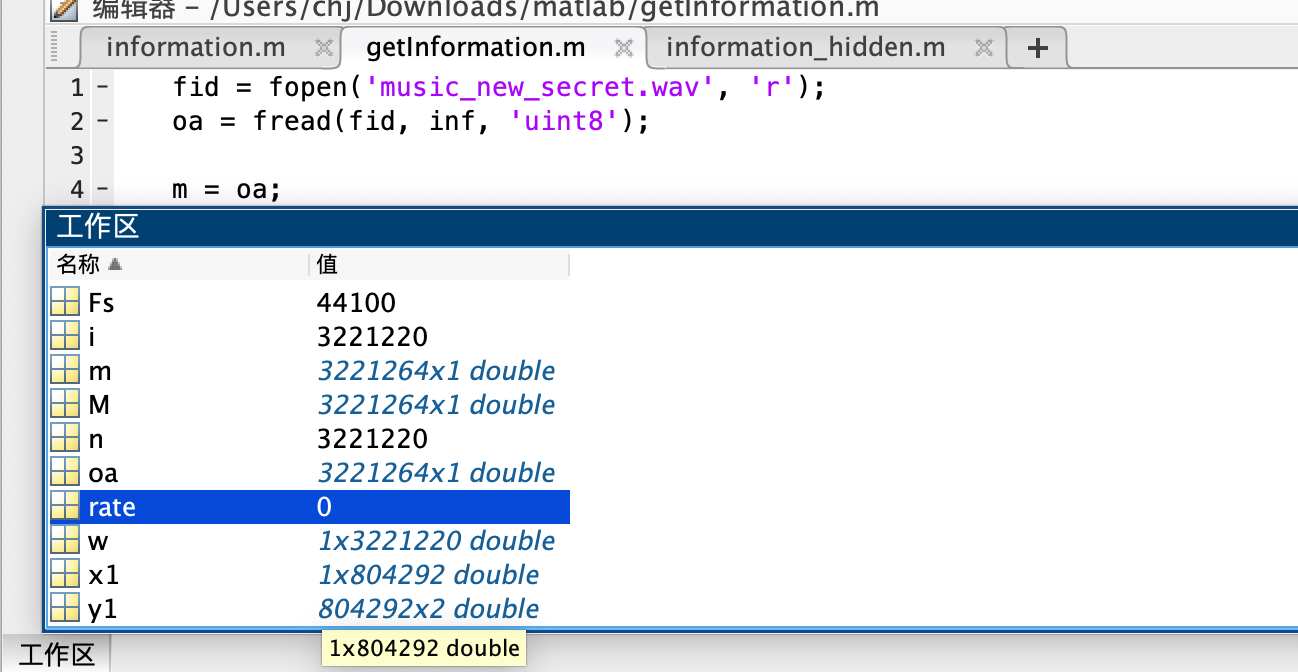
隐藏的信息为：



2、提取到的隐藏信息

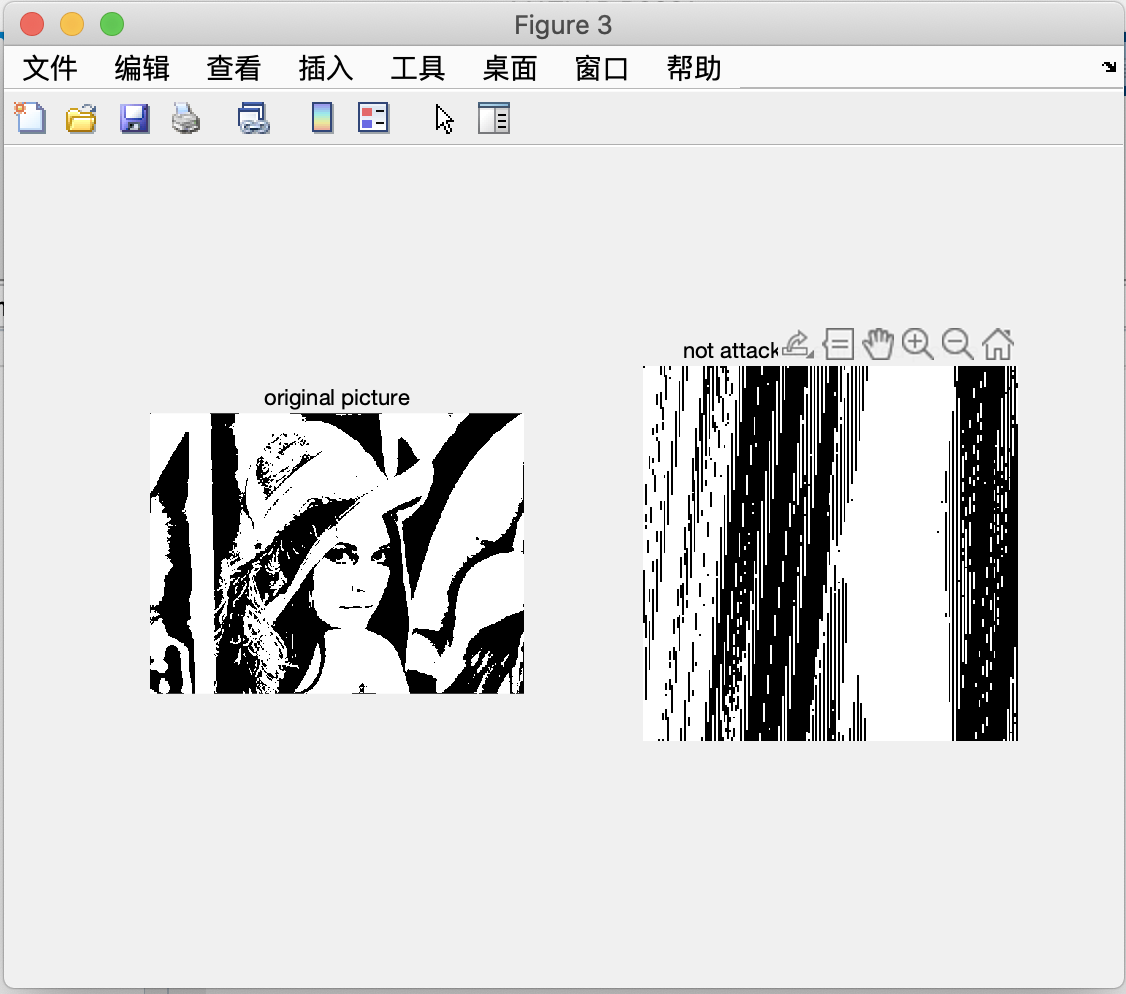


可以发现，提取到的信息和隐藏的信息存在一定的区别，粗略计算后误码率rate = 0，可以看出误差并不大。



3、在WAV音频中隐藏一个二值图像

用于隐藏的图像和提取出的图像比较：



# 实验分析暨客观评价

可以发现，虽然我们按照标准的LSB嵌入算法往wav文件里嵌入二值图像，但嵌入的图像与提取出的图像差距较大，但也能依稀看出轮廓，调整了几次图像后也是如此。

经过研讨，我们认为可能是音频文件不是标准的用于实验的音频所导致的，就好像二值图像信息隐写，图像块中黑色像素多于白色像素嵌1，白色像素多于黑色像素嵌0，但若不是使用标准的实验用图像（如lena），其像素分布本就不均匀，使用这种算法恐怕很难找到合适的嵌入点。

LSB算法采用流加密分别对数据本身和嵌入过程进行加密，其安全性完全依赖于密钥，信息嵌入和提取算法简单，速度快，容量较大，具有一定的实用性，但其对信道干扰及数据操作的抵抗力很差，诸如信道干扰、数据压缩、滤波、重采样、时域缩放等操作都会破坏编码信息。

# 心得体会

1、在进行信息隐藏与数字水印的相关实验时，最好使用标准的实验用的图像、音频等载体，否则，使用非实际大规模应用的算法时，实验的效果很有可能会受影响。

2、为了保证嵌入信息的安全性，我们一般并不会采用直接嵌入比特流的方式，而是会采用哈希函数、伪随机数生成器等方法进行层层加密之后再嵌入。

# 实验分工介绍

本组成员均认真积极地合作完成了本次实验，在实验选材、实验设计、实验分析等各个环节均有参与，收获颇多，无偷懒行为。