

浙江大学



课程名称:	多媒体安全
姓 名:	
学 院:	计算机学院
专 业:	计算机科学与技术
学 号:	
指导老师:	黄劲
完成时间:	2023 年 4 月 17 日

实验一: E_BLIND/D_LC 系统测试

一、实验目的

1. 了解 E_BLIND/D_LC 系统的基本原理, 理解 False Positive 和 False Negative 的概念, 掌握 Linear Correlation 的计算。
2. 考察 E_BLIND/D_LC 系统检测值的分布, 给出检测值的概率分布图或密度分布图。
3. 测试 8-bit 灰度值截断对系统检测准确率的影响。

二、实验内容与要求

1. 实现 E_BLIND/D_LC 系统。
2. 设计一张水印, 嵌入强度 $\alpha = 1$, 使用该水印测试 E_BLIND/D_LC 系统应用于不同封面时的检测准确率, 计算 False Positive/Negative Rate, 并绘制如课本图 3.6 的检测值分布图或密度图。要求封面数量不少于 40 张 (建议使用课程提供的标准数据集作为测试封面)。
3. 设计不少于 40 张不同的水印, 使用固定的嵌入强度 $\alpha = 1$, 选择一张黑白像素比例 (即灰度值接近 0 或 255 的像素数量占总像素数量的比例) 不高于 30% 的封面, 测试不同的水印嵌入同一封面时 E_BLIND/D_LC 系统的检测准确率, 计算 False Positive/Negative Rate, 并绘制如课本图 3.6 的检测值分布图或密度图。
4. 选取一张黑白像素比例不低于 50% 的原始封面, 重复子实验 3, 比较检测准确率并分析原因。

三、实验环境

语言版本: MATLAB R2020b

四、实验过程

4.1 E_BLIND/D_LC 系统实现

此部分主要是实现 E_BLIND 水印生成和 D_LC 水印检测。

为实现 E_BLIND 水印生成, 我们需要采用一些随机分布函数随机出一系列水印模式。由于提供的测试集中, 最大大小的图片为 5312*5312, 我们使用 randn() 函数生成均值为 0、方差为 1 的 5312*5312 的 double 类型矩阵, 作为水印模式。

定义函数 E_BLIND() 根据随机数种子 seed 返回水印模式 watermark, 实现如下:

```
1 function [watermark] = E_BLIND(seed)
2
3     rng(seed);
4     size = 5312;
5     watermark = randn(size, size);
6 end
```

生成的水印需要植入封面。首先将水印模式裁剪到图片大小，然后采取逐元素简单相加方式即可得到含水印的图片。另需要一个水印强度参数 α 控制水印的鲁棒性。

定义函数 embed() 根据封面 cover，水印模式 watermark，水印强度 alpha 返回含水印的封面 markedWork，实现如下：

```
1 function [markedWork] = embed(cover, watermark, alpha)
2
3     [sizeX, sizeY] = size(cover);
4     watermark = watermark(1:sizeX, 1:sizeY);
5     markedWork = cover + alpha * watermark;
6 end
```

D_LC 水印检测的原理是：在理论上，随机生成的水印与随机的封面的线性相关系数不会很大（接近 0），因此可以通过待检测图片与原水印的线性相关系数的大小判定原图片是否存在水印。定量的检测函数是：

$$z_{lc}(c, w_r) = \frac{1}{N} c * w_r = \frac{1}{N} \sum_{x,y} c[x,y] * w_r[x,y]$$

如果 z_{lc} 的绝对值大于人为设定的检测阈值 τ_{lc} ，则认为图片中存在水印，并根据 z_{lc} 的符号判定是正或负水印。否则，认为未植入水印。

定义函数 D_LC() 根据待检测图像 img 和水印模式 watermark 返回 z_{lc} 值，实现如下：

```
1 function [result] = D_LC(img, watermark)
2
3     [sizeX, sizeY] = size(img);
4     watermark = watermark(1:sizeX, 1:sizeY);
5     result = sum(sum(img .* watermark)) / sizeX / sizeY;
6 end
```

以上三个函数就可实现 E_BLIND/D_LC 系统的大部分工作，只需人为给出 τ_{lc} 等其他参数以及判定逻辑即可。

此外, 再定义一个函数 truncate() 用以作 8-bit 灰度截断, 实现如下:

```
1 function [work] = truncate(work)
2
3     [sizeX, sizeY] = size(work);
4     for i = 1:sizeX
5         for j = 1:sizeY
6             work(i, j) = round(work(i, j));
7             if work(i, j) > 255
8                 work(i, j) = 255;
9             end
10            if work(i, j) < 0
11                work(i, j) = 0;
12            end
13        end
14    end
15 end
```

4.2 相同水印应用于不同封面的检测

设定水印种子 seed 为 19260817, 取给定数据集中文件名字典序最小的 40 张图片作为测试封面。得到的密度图如下:

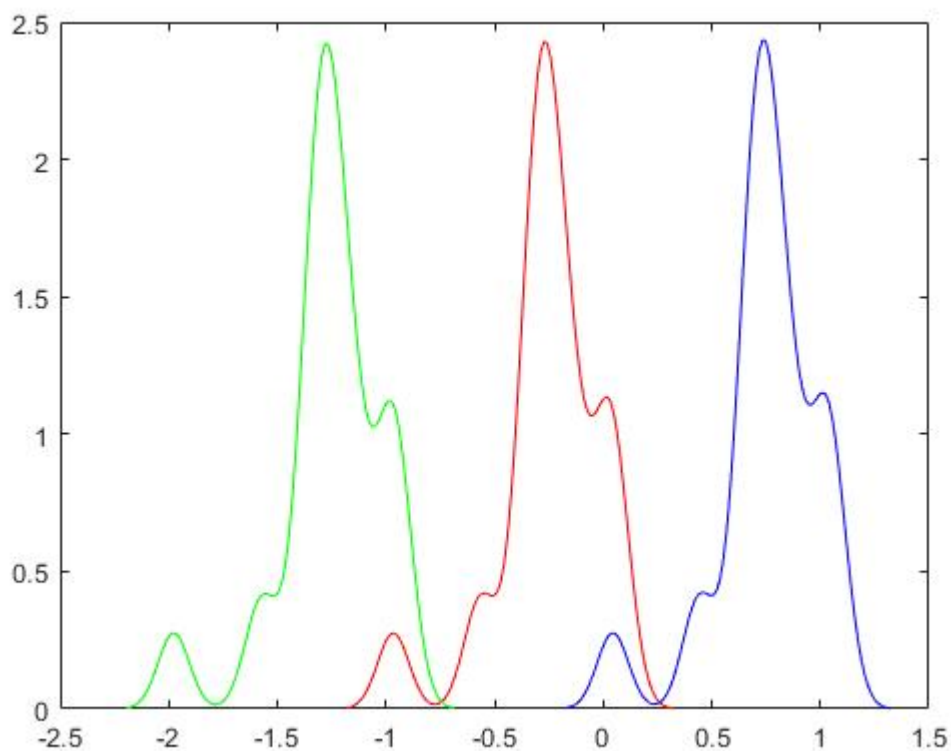


图 1 水印 19260817 应用于不同封面的 z_{lc} 密度图

图中, 绿色曲线为负水印检测值, 红色曲线为无水印, 蓝色曲线为正水印。

在阈值 $\tau_{lc} = 0.6$ 下, 检测准确率为 91.25%, 假阳性率为 2.5%, 假阴性率为 6.25% (以上数据均为正负水印的平均值)。

4.3 不同水印应用于相同封面的检测（黑白像素比例低时）

选取数据集中的 couple.bmp 作为测试封面，水印种子为 1 到 40。得到的密度图如下：

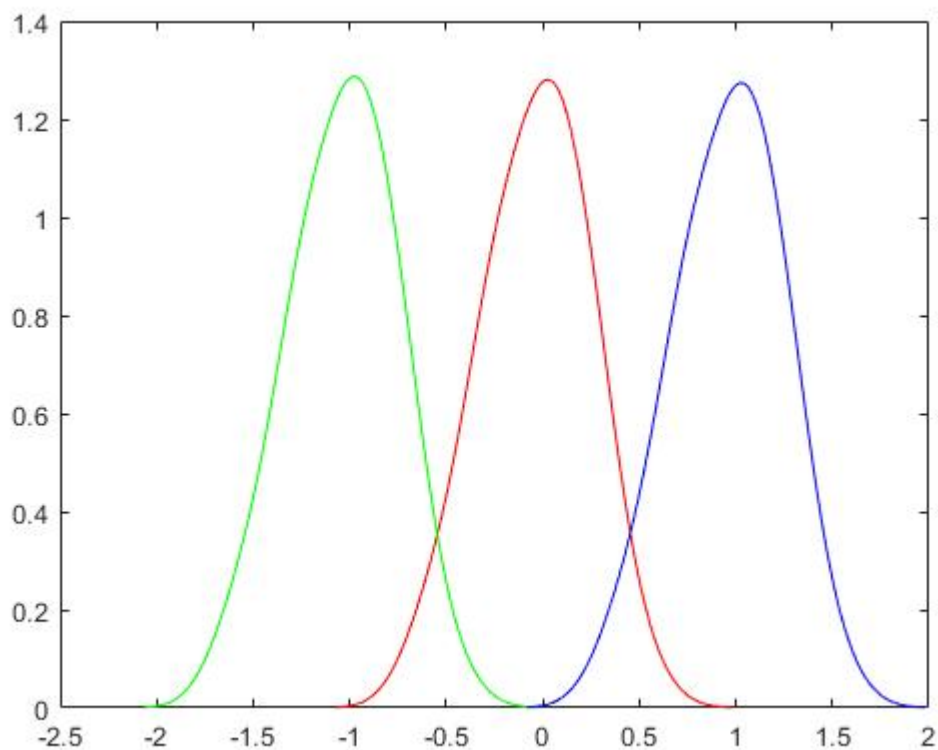


图 2 不同水印应用于 couple.bmp 的 z_{lc} 密度图

图中，绿色曲线为负水印检测值，红色曲线为无水印，蓝色曲线为正水印。

在阈值 $\tau_{lc} = 0.6$ 下，检测准确率为 93.75%，假阳性率为 1.25%，假阴性率为 5%（以上数据均为正负水印的平均值）。

如果对加水印的图像应用 8-bit 灰度截断, 则密度图:

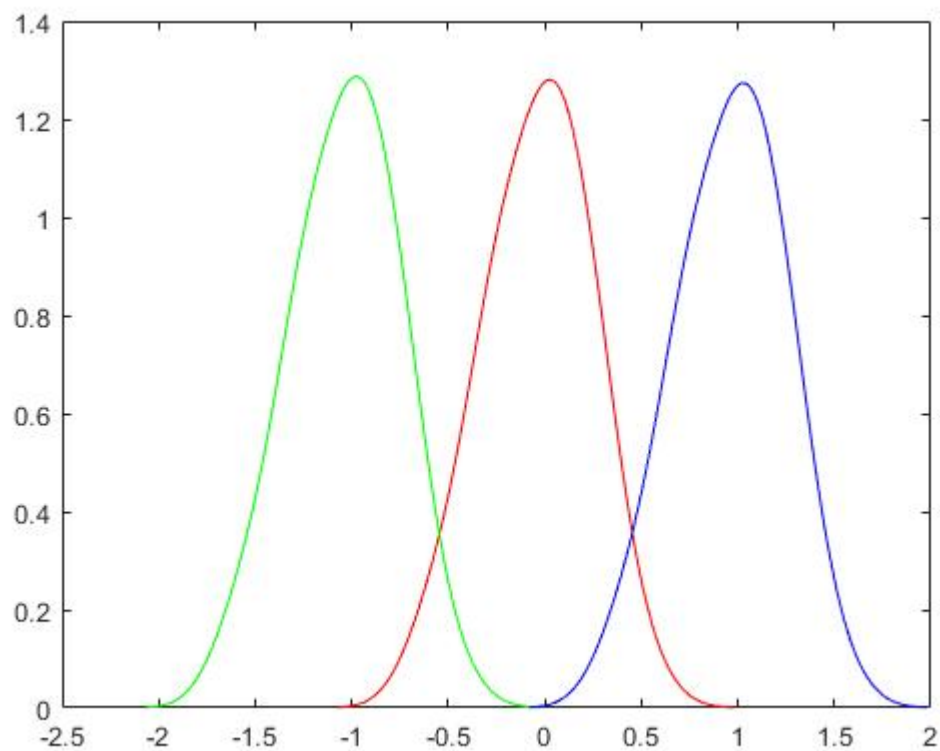


图 3 不同水印应用于 8-bit 截断的 couple.bmp 的 z_{lc} 密度图

此时的检测准确率仍然为 93.75%, 1.25%, 假阴性率为 5% (以上数据均为正负水印的平均值)。

4.4 相同水印应用于不同封面的检测（黑白像素比例高时）

选取数据集中的 rec.bmp 作为测试封面，水印种子为 1 到 40。得到的密度图如下：

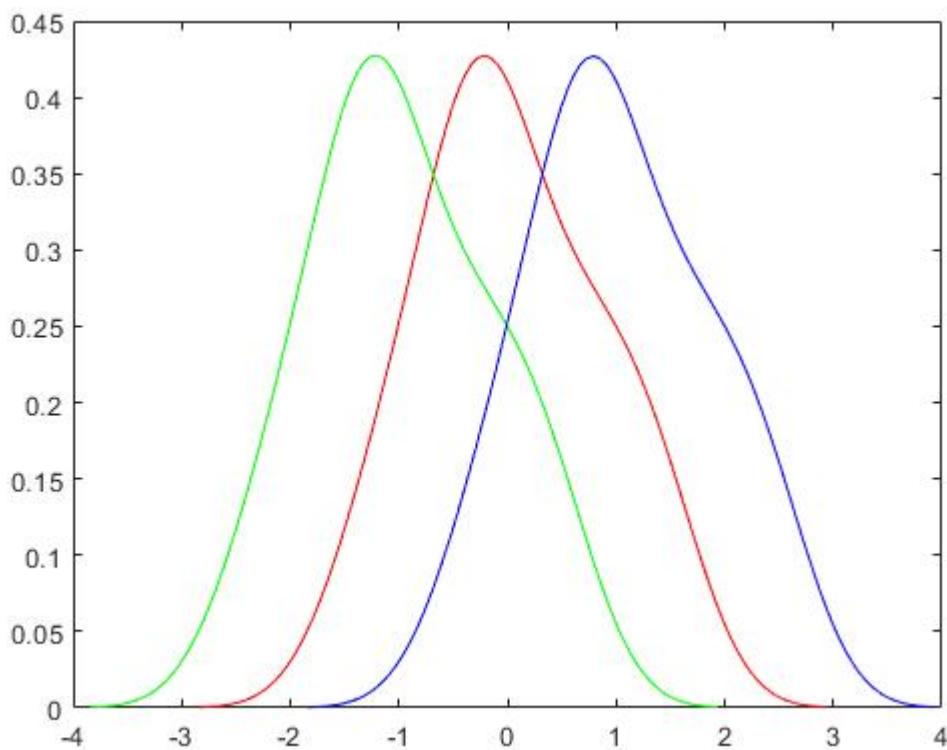


图 4 不同水印应用于 rec.bmp 的 z_{lc} 密度图

图中，绿色曲线为负水印检测值，红色曲线为无水印，蓝色曲线为正水印。

在阈值 $\tau_{lc} = 0.6$ 下，检测准确率为 43.75%，假阳性率为 23.75%，假阴性率为 32.5%（以上数据均为正负水印的平均值）。

如果对加水印的图像应用 8-bit 灰度截断, 则密度图:

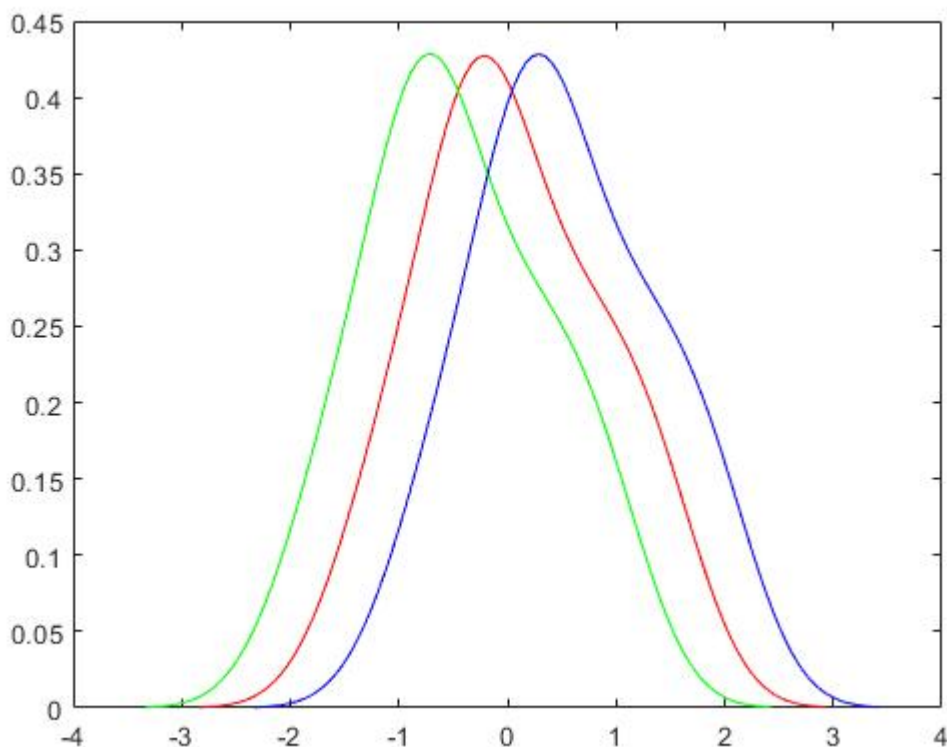


图 5 不同水印应用于 8-bit 截断的 rec.bmp 的 z_{lc} 密度图

此时的检测准确率为 17.5%, 假阳性率为 23.75%, 假阴性率为 58.75% (以上数据均为正负水印的平均值)。

五、实验分析与结论

5.1 检测准确率分析

E_BLIND/D_LC 系统的检测准确率较为依赖水印和图片的线性相关系数, 如果两者的相关系数本身就很低则检测准确率会严重降低。

这点导致该系统在某些非自然图片中严重不适用, 如数据集中的 rec.bmp (比较自然图像下的图 2 和 rec.bmp 下的图 4)。在这张色块明显且完全非随机的图像上, 该系统的检测准确率从 90% 左右陡降到 40%。这也意味着该方法难以应用到动画等人工绘制及生成图像较多的领域。

5.2 8-bit 灰度截断对检测准确率的影响

从图 2 和图 3 的对比可以看出, 8-bit 灰度值截断对自然图片影响不大; 自然图片很少存在极值 (0 或 255) 像素点, 并且取整操作基本上也是添加一个随机向量, 从而截断造成的信息损失相对少。对

couple.bmp, 检测准确率在截断前后没有大的变化。

从图 4 和图 5 的对比可以看出, 8-bit 灰度值截断对极值像素多的图片影响相当大。极值像素很可能在添加水印后超过 8-bit 范围, 从而截断造成的信息损失极大 (可以认为是把约 $\frac{1}{4}$ 的水印向量值强制置为 0, 导致水印效果降低)。从而, 在截断后, z_{lc} 密度曲线存在明显的朝 0 移动, 导致假阴性率陡升。不过, 假阳性率没有大的变化。

六、实验感想

使用 MATLAB 进行实验, 难度整体不高, 难度主要是在选取合适的样本图像, 便于后续分析上。例如, 我也尝试过 brain.bmp 作为 task 3 的样本, 但该图片得到的数据很难解释 E_BLIND/D_LC 系统原理导致的某些特性。最后还是选择了 100% 黑白像素的非自然图像 rec.bmp。

通过本实验, 我对水印算法有了初步的认识, 大体上理解了水印和检测的方法及原理, 有助于后续对课程的学习。