**《现代密码学》实验报告**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **年级、专业、班级** | | **20级信息安全专业01班** | | | **姓名** | **蔡欣彤** |
| **实验题目** | **Arnold变换** | | | | | |
| **实验时间** | **2023.5.1** | | **实验地点** | **DS3304** | | |
| **实验成绩** |  | | **实验性质** | **☑验证性 □设计性 □综合性** | | |
| 教师评价：  □算法/实验过程正确； □源程序/实验内容提交 程序结构/实验步骤合理；  □实验结果正确； 语法、语义正确； 报告规范；  其他：  评价教师签名： | | | | | | |
| 一、实验目的  通过编程实现Arnold变换，加深对扩散和混淆设计原则的理解，并了解Arnold变换的周期性。 | | | | | | |
| 二、实验项目内容  自己创建明文信息（需正方形矩阵形式）或者选用一张正方形灰度图像，选择一个密钥，即迭代次数，编写Arnold变换实现程序，实现置乱和恢复的操作； | | | | | | |
| 1. 实验过程或算法（源程序）   1.编程实现.  使用Python平台，先对数据进行一个预处理，再对图片进行置乱，再恢复。  import numpy as np import random,cv2 from PIL import Image def img():  data = cv2.imread('baijingting1.png')  image = cv2.resize(data, dsize=(200,200), fx=1, fy=1, interpolation=cv2.INTER\_LINEAR)   mat= np.array(image)  r,g,b = [image[:,:,i] for i in range(3)]  img\_gray = r\*0.299+g\*0.587+b\*0.114#灰度处理  image2 = Image.fromarray(img\_gray)  image2.show()  a=1  b=1  N=200  an=np.zeros([200,200])  for i in range(0,150):#200\*200矩阵150轮变换复原  for y in range(0,200):  for x in range(0,200):#变换每个像素点  xx=(a\*x+b\*y)%N  yy=(a\*x+(a\*b+1)\*y)%N  an[xx][yy]=img\_gray[x][y]  img\_gray[:][:]=an[:][:]  print(i)  if (i+1)%50==0:#每50轮输出一次变换图像  image2 = Image.fromarray(img\_gray)  image2.show()  if \_\_name\_\_=="\_\_main\_\_":  img() | | | | | | |
| 四、实验结果及分析和（或）源程序调试过程  对原图片处理后：    迭代次数为200.  进行前50次置乱：    再进行50次置乱：    最后50次置乱得到原来的照片。 | | | | | | |