

实验指导书 2

by 李蔚洋

Arnold 变换，又称猫脸变换(catmapping)。

一种对图像的置乱方法。

狭义Arnold 变换

狭义的Arnold变换原理如下：

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \bmod(N)$$

拆开看：

$$\begin{cases} x' = (x + y) \bmod(N) \\ y' = (x + 2y) \bmod(N) \end{cases}$$

注意：

1. Arnold变换,实质上是对于图像的各个像素点进行移动。
2. 置乱的实质是像素点新位置与旧位置的映射，且该映射一一对应。
3. 可以把Arnold变换理解为先作x轴方向的错切变换，再作y轴方向的错切变换，最后的模运算相当于切割回填操作：

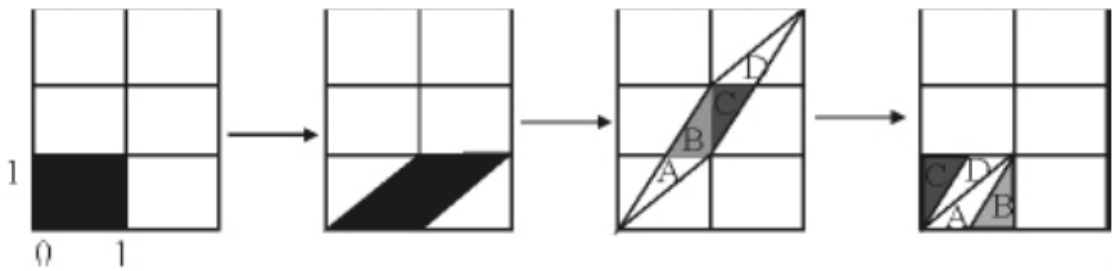


图 4 Arnold 变换

4. 一次Arnold变换是不够达到置乱效果的，因此我们要进行一定次数的迭代，将上一次的输出作为下一次输入。
5. Arnold变换有周期性。

N	4	6	8	10	12	14	16	18
周期T	3	12	6	30	12	24	12	12
N	20	60	100	128	200	256	512	1024
周期T	30	60	150	96	150	192	384	768

广义Arnold 变换

广义Arnold变换：

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{bmatrix} 1 & a \\ b & ab+1 \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \bmod(N)$$

拆开：

$$\begin{cases} x' = (x + ay) \bmod(N) \\ y' = (bx + (ab + 1)y) \bmod(N) \end{cases}$$

不难发现，当a=b=1时，为狭义Arnold 变换。

恢复

思考下怎么恢复原图？

一种方法是利用周期性，另一种是把像素点移回原来的位置。

对原来的变换矩阵求逆，得到逆变换公式。

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} 2 & -1 \\ -1 & 1 \end{bmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \bmod(N)$$

拆开：

$$\begin{cases} x = 2x' - y' \bmod(N) \\ y = -x' + y' \bmod(N) \end{cases}$$

程序设计注意事项

1. 需要两个函数，分别实现置乱和恢复。注意本次实验会有矩阵操作，请使用对应工具帮助编程。
2. 必须实现狭义Arnold变换，也可以稍加改动写广义Arnold变换，没什么难度。
3. 编程语言自选，但是需要注意的是，本次实验是对图片进行操作，在实验报告时需要读取图片、生成图片等操作，因此需要使用到各语言对应的图像处理工具包。

c++中的opencv的环境配置略有困难，推荐各位同学使用python，使用cv2、matplotlib、numpy等包，以免在环境配置上花费太多时间。