结合分数傅立叶变换及区域移位法进行图像编码

王小芳,聂守平,赵书安

(南京师范大学物理科学与技术学院,210097,江苏,南京)

[摘要] 提出了图像区域移位编码算法,并结合分数傅立叶变换,实现了对图像的多重加密编码. 计算机模拟表明该编码方法自由度大,保密性强.

[关键词] 图像编码, 分数傅立叶变换, 区域移位

[中图分类号] TN919.8, [文献标识码] A, [文章编号] 1001-4616(2005)02-0051-04

The Methods of Image Encryption Using Fractional Fourier Transform and Region Shifting Encode

Wang Xiaofang, Nie Shouping, Zhao Shu 'an

(School of Physical Science and Technology, Nanjing Normal University, 210097, Nanjing, China)

Abstract: An algorithm of image region shifting encode is proposed. Combing it and fractional Fourier transform, multi-channel image encryption encode is implemented. The results of computer simulation show that this method is flexible in the range freedom and strong in secrecy.

Key words: image encryption, fractional Fourier transform, region shifting encode

图像编码是图像处理的重要组成部分,就图像编码的目的而言,可以分为图像压缩编码和图像加密编码两类. 图像编码广泛应用于图像传输、图像存储、图像识别等领域. 对于图像加密[1],通过加密操作后,原来的数字图像变为类似于信道随机噪声的信息,这些信息在不知道密钥的情况下是不可识别的(除非进行有效破译),进而可以有效地保护图像数据. 图像加密编码可以在空间域实现,也可以在变换域实现. 空间域图像编码可以通过改变像素在图像中的位置、附加随机噪声来实现. 变换域可以通过离散余弦变换、傅立叶变换等来实现. 目前为止利用傅立叶变换编码的技术有单随机位相法、双随机位相法[2],都是通过产生一个或多个随机位相函数叠加在图像的频谱位相信息上,从而实现图像保密编码. 解码的关键是要知道所采用的随机位相函数.

本文同时采用图像空间域编码和频率域编码,实现图像加密. 空间域中提出了区域移位编码方法,通过编码前后区域的对应关系实现图像的区域移位. 在频率域对移位后的图像进行分数傅立叶变换[3],利用其分数阶作为图像再现的一个新的约束和保密的自由度,可建立新的图像编码技术. 区域移位易于硬件实现,分数傅立叶变换易于光学实现. 循环进行空间域和频率域编码,实现对图像的多重保密编码. 计算机模拟表明,这种编码方式自由度十分巨大,只有同时清楚区域移位密码和分数阶傅立叶变换密码才可能对图像解码.

1 图像编码原理

1.1 区域移位编码

如图 1 所示原始图像 A 大小为 $M \times M$,把它分成 N^2 个小的区域,每个区域的大小为 $S \times S$,每个区域的像素个数为 M^2/N^2.

如果我们按一定的规律改变每个区域在图像中位置,即图像 A 中第 g 个区域,在编码输出图像 B 中则可能是第 k 个区域,用数学式子表示为

收稿日期:2005-01-10.

作者简介:王小芳,女,1980—,硕士研究生,主要从事光电子技术和激光应用的学习与研究. E-mail:cindyheartsu@163.com

通讯联系人:聂守平,1967—,副教授,主要从事光信息处理的教学与研究. E-mail:nieshouping@email.njnu.edu.cn

$$B_k = A_g \tag{1}$$

这样就可以实现对图像的编码,这种编码称为区域移位编码. g 和 k 之间的关系可以通过一个大小为 N^2 的随机一维矩阵 $\boldsymbol{C} = (c_k)$ $(k = 0,1,\cdots,N^2-1)$ 来建立,即

$$g = c_k \tag{2}$$

下面我们建立原始图像 A 和编码后的 B 像素之间的一一对应关系. 根据所划分的图像区域的大小 S,可知像素 $B(i,j)$ 位于图像 B 中的第 k 个区域,即

$$k = N \times \mathrm{int}[i/S] + \mathrm{int}[j/S] \tag{3}$$

式中 int() 表示取整运算. 若取区域左上角为坐标原点,则像素 $B(i,j)$ 在区域 k 中位于第 p 行和第 q 列,记为 $B_k(p,q)$,即

$$p = \mathrm{MOD}(i/S),\quad q = \mathrm{MOD}(j/S) \tag{4}$$

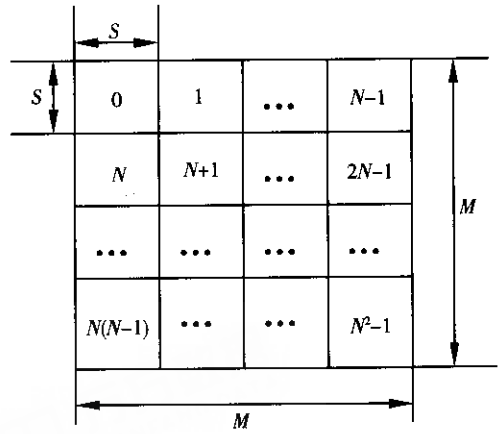

图1 图像分块

如果我们已经建立了图像 A 和编码图像 B 之间的区域关系,则可由(2)式得到像素 $B(i,j)$ 编码之前属于图像 A 中的区域 g. 同时由于我们采用的是区域移位法,即图像 B 的第 k 个区域的像素和图像 A 的第 g 个区域的像素是一一对应的,所以有

$$B_k(p,q) = A_g(p,q) \tag{5}$$

很容易求出像素 $A_g(p,q)$ 在整幅图像 A 中的坐标

$$i' = S \times \mathrm{int}(g/N) + p,\quad j' = S \times \mathrm{MOD}(g/N) + q \tag{6}$$

公式(2)至(6)建立了图像 B 中像素 $B(i,j)$ 和图像 A 中像素 $A(i',j')$ 的一一对应关系.

如果把大小为 128×128 像素的图像分成16个区域,则每个区域的大小为 32×32 像素,即 $M = 128, N = 4, S = 32$. 预先建立大小为16的一维随机矩阵 $\boldsymbol{C}$ 来表示原始图像和编码输出图像区域之间的对应关系,

$$\boldsymbol{C} = [13,8,7,15,1,2,5,12,9,4,0,3,6,10,11,14]$$

表1利用前述区域移位原理,计算了像素 $B(57,113)$ 在图像 A 中的对应像素 $A(121,25)$.

表1 图像 B 和图像 A 像素对应关系

图像 B			图像 A		
整幅图像像素坐标	图像 B 区域	区域中坐标	图像 A 区域	区域中坐标	整幅图像像素坐标
(57,113)	7	(25,17)	12	(25,17)	(121,25)

只要改变反映原始图像 A 和编码输出图像 B 之间的区域关系矩阵 $\boldsymbol{C}$,就可以实现图像 A 的不同编码,矩阵 $\boldsymbol{C}$ 有多少种排列,就可以得到多少种编码图像. 对于分成16个区域的图像则有16种编码图像.

显然区域移位编码过程是可逆的,即可以按照编码的逆过程实现图像的解码. 解码的关键是要使用完全相同的区域关系矩阵,只有知道预先确立的区域关系矩阵才能实现正确解码,这就是使用区域移位进行图像保密编码的原理.

图2(a)是原始图像,图2(b)和(c)是将图像分成16个区域和256个区域的编码图像. 图2(d)和(e)是将图像分成64个区域后采用两种不同的区域编码矩阵得到的图像编码.

1.2 分数傅立叶变换编码

为了简单起见,采用一维的情况. 输入图像 $f(x)$ 的分数傅立叶变换(FRT)定义[4]为:

$$f_a(x_a) = F_a\{f(x)\}(x_a) = \int_{-\infty}^{+\infty} f(x) B_a(x,x_a)\,\mathrm{d}x \tag{7}$$

其中 $B_a(x,x_a)$ 为核函数,$B_a(x,x_a) = A_\phi \exp[\mathrm{j}\pi(x^2\cot\phi - 2xx_a + x_a^2\cot\phi)]$,$a$ 为FRT的分数阶,x, x_a 分别是空域和分数域坐标,$\phi = a\pi/2$,A_ϕ 是仅取决于分数阶的常量,当 a 一定时 A_ϕ 也就确定了:$A_\phi = \sqrt{\dfrac{1-\mathrm{j}\cot\phi}{2\pi}}\exp(-\mathrm{j}2\pi)$. 这里需要指出的是,$-2 \leqslant a \leqslant 2$ 且 $a \neq 0$. 而当 $a = 1$ 时,即为一般傅立叶变换.

分数傅立叶变换是一种线性变换,二维情况中,在 x, y 方向可以同时分别进行不同分数阶的FRT变换. 分

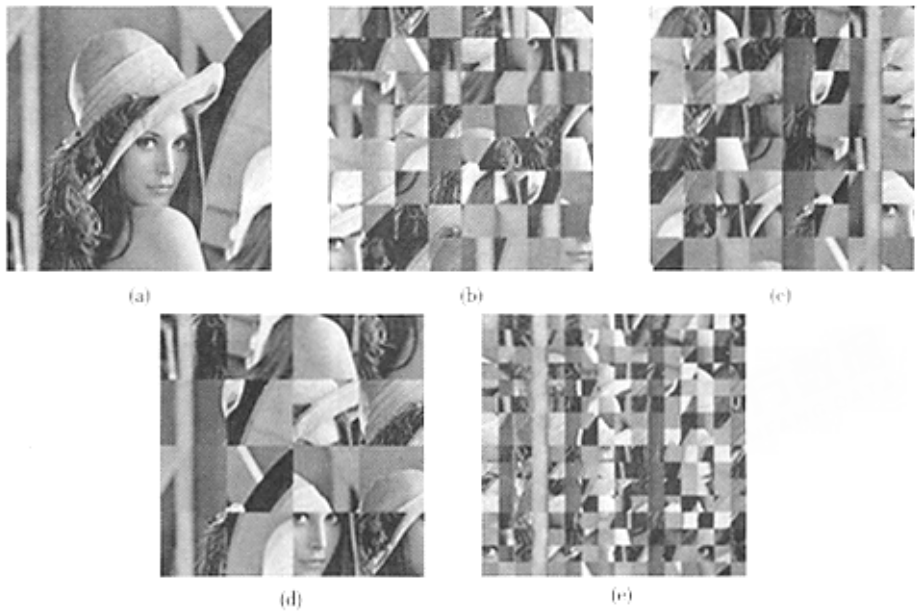

图 2 图像区域移位编码

数傅立叶变换是傅立叶变换的推广,因此同样也存在反变换. 当分数阶取 $-a$ 时即是分数阶为 a 的 FRT 变换的反变换.

这里利用不同的分数阶对图像进行编码,图 3 所示是对图 2(a) 所示的图像进行分数阶分别为 $a_x = a_y = 0.2$;$a_x = a_y = 0.5$;$a_x = a_y = 0.8$;$a_x = a_y = 1.0$ 的分数傅立叶变换振幅分布. 可以看出当分数阶 a_x 和 a_y 的值较小时,分数傅立叶变换的结果与原始图像具有相似性;当 a_x 和 a_y 的值为 1 时得到图像准确的傅立叶变换.

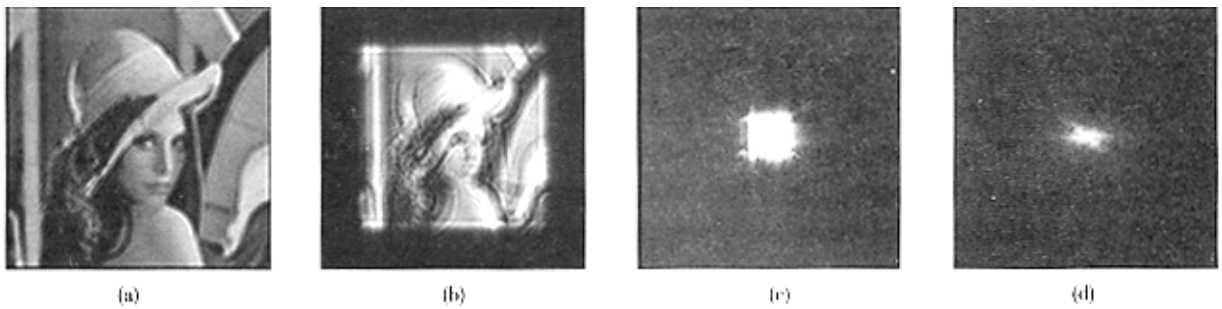

图 3 图像分数傅立叶变换编码

1.3 双重编码

双重编码即是循环使用区域移位编码和分数傅立叶编码. 将区域移位编码记为 $J_C()$,区域移位编码的反变换记为 $J_{-C}()$. 设输入图像用 $f(x,y)$ 表示.

首先,对图像进行区域移位变换,得到:

$$J_C\{f(x,y)\} \tag{8}$$

接着,对变换过后的图像进行分数傅立叶变换,可以得到:

$$F_a\{J_C\{f(x,y)\}\} \tag{9}$$

这样,也就得到最后编码结果为:

$$g(x,y) = F_a\{J_C\{f(x,y)\}\} \tag{10}$$

解码的过程也就是编码的反过程. 只要对(10) 式进行反变换就可以得到解码信息,可由下式表示:

$$f(x,y) = J_{-C}\{F_{-a}\{g(x,y)\}\} \tag{11}$$

(11) 式即为解码输出图像. 这里值得一提的是,上述进行分数傅立叶变换过程中,下标虽然是 a,实际上每一次变换都包含 x 和 y 两个方向,即有 a_x,a_y 两个变换密钥. 如果在(10) 式的基础上继续交替循环应用区域移位变换和分数傅立叶变换,那么编码程度也会随之加深.

另外,双重编码也可以先进行分数傅立叶变换后进行区域移位编码,即最后的编码结果为:

$$g'(x,y) = J_C\{F_a[f'(x,y)]\} \tag{12}$$

相应的解码过程为:

$$f'(x,y) = J_{-C}\{F_{-a}[g'(x,y)]\} \tag{13}$$

2 计算机模拟

利用双重编码原理(11)式对图2(a)的数字图像进行了加密和解密计算机模拟,结果如图4所示.

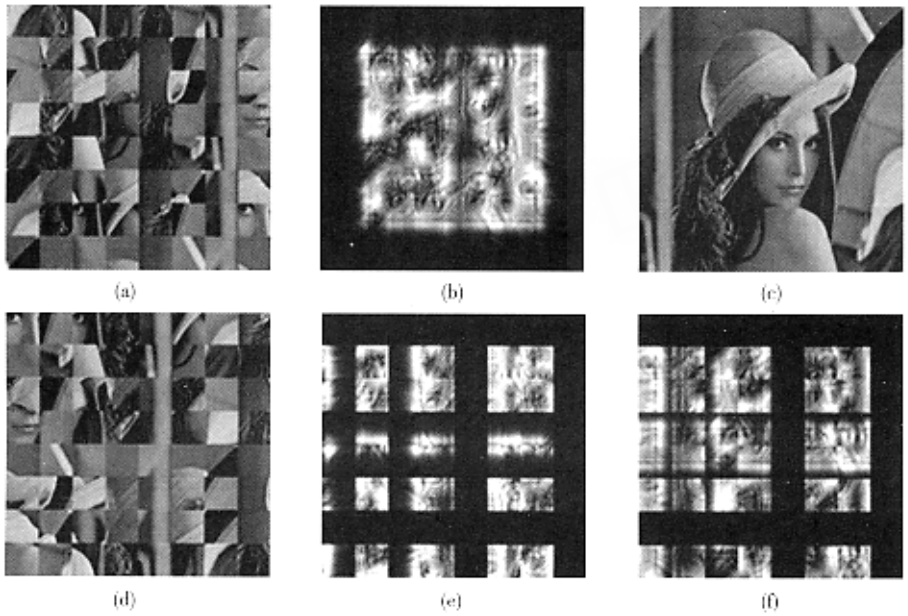

图4 双重图像加密编码与解码

图4中,(a)为对图像进行了64区域的移位编码;(b)是在(a)的基础上进行了FRT变换,这是编码后的振幅图像,其分数阶为 $a_x = a_y = 0.5$;(c)是正确解码结果;(d)是区域移位编码错误的情况下的解码结果;(e)是在FRT的分数阶错误的情况下的解码结果;而(f)是解码顺序错误,先进行区域移位反变换后进行FRT反变换.从图4中的解码效果可以看出,只有在完全知道编码密钥及编码顺序的情况下才能完全恢复图像,如果其中有一步出错,则不能恢复原始图像.

3 结论

本文在区域移位编码的基础上,结合分数傅立叶变换,实现了图像加密编码.这种方法是在空间域与频域都进行变换的双重编码.计算机模拟结果表明,该方法有很好的加密/解密效果和安全性.编码的自由度很广,而在解码的时候需要知道两种不同变换的密钥才可以完全解码原图像,加密程度很高.

[参考文献]

[1] Chung K L, Chang L C. Large encryption binary images with higher security[J]. Pattern Recognition Letters, 1998, 19(6): 461—468.

[2] Refregier P, Javidi B. Optical image encryption based on input plane and Fourier plane random encoding[J]. Optics Letters, 1995, 20(7): 767—769.

[3] Mendlovic D, Ozaktas H M. Fractional Fourier transforms and their optical implemnent I[J]. Journal of the Optical Society of America (A), 1993, 10(9): 1875—1881.

[4] Hennelly B, Sheridan J T. Optical image encryption by random shifting in fractional Fourier domain[J]. Optics Letters, 2003, 28(4): 269—271.

[责任编辑:丁蓉]