



A Fast Algorithm of Exponent Moments in Image Reconstruction

Wu Yu^{*}, Ping Zi-liang, Li Yin

Department of Communication and Information Engineering, Century College, Beijing University of Posts and Telecommunications, Beijing, China

Email address:

wuyuq2003@163.com (Wu Yu), pingziliang@ccbupt.cn (Ping Zi-liang), liyin@ccbupt.cn (Li Yin)

^{*}Corresponding author

To cite this article:

Wu Yu, Ping Zi-liang, Li Yin. A Fast Algorithm of Exponent Moments in Image Reconstruction. *Science Discovery*. Vol. 4, No. 5, 2016, pp. 292-296. doi: 10.11648/j.sd.20160405.15

Received: August 24, 2016; **Accepted:** September 12, 2016; **Published:** October 13, 2016

Abstract: Image moments as a very effective highly concentrated image feature differ from image simple geometric features. Using image moments as a global feature to describe image feature information is a key problem in the image information domain of image recognition, image transformation, image classification, image transmission and image compression. On the basis of exponent moments, the image can be reconstruction approximately. And the exponent moments calculation can be more fast and more accurate using 2-D Fast Fourier Transform.

Keywords: Radial Harmonic Fourier Moments, Exponent Moments, 2-D Fast Fourier Transform

一种指数矩重构图像的快速算法

吴娱^{*}, 平子良, 李殷

通信与信息工程系, 北京邮电大学世纪学院, 北京, 中国

邮箱

wuyuq2003@163.com(吴娱), pingziliang@ccbupt.cn(平子良), liyin@ccbupt.cn(李殷)

摘要: 图像矩作为一种非常有效的高度浓缩的图像特征, 区别于图像简单的几何特征。利用图像矩作为图像的全局特征来描述图像的特征信息, 是图像识别、图像变换、图像分类、图像传输、图像压缩等图像信息处理技术领域的一个关键问题。基于指数矩, 可以近似的重构原图像。在此基础上, 利用二维快速傅里叶变换能更快速、精确的完成指数矩的计算。

关键词: 圆谐-傅里叶矩, 指数矩, 二维快速傅里叶变换

1. 引言

矩不变量是一种高度浓缩的信息特征, 能非常有效的描述图像特征。国内外有大量关于图像矩的研究, 如 Zernike矩、Legendre矩、圆谐-傅里叶矩等, 并成功地应用于模式识别、指纹辨认、天气预报、医疗诊断、遥感技术、地震探测等多个领域中[1]。本文计算基于指数构建

的圆谐-傅里叶矩, 即指数矩 (CEMs), 用一种快速算法验证其在图像描述中的性能, 表明该算法是一种比传统算法更快速且精确的指数矩计算方法。

2. 指数矩的定义和计算

圆谐-傅里叶矩[2-4]在极坐标下,基函数 $P_{nm}(r, \theta)$ 是由径向函数 $T_n(r)$ 以及角向傅里叶因子 $\exp(jm\theta)$ 组成:

$$P_{nm}(r, \theta) = T_n(r) \exp(jm\theta) \quad (1)$$

其中, n 为非负数, m 为整数, r 的取值范围 $0 \leq r \leq 1$, θ 的取值范围 $0 \leq \theta \leq 2\pi$ 。

$$T_n(r) = \begin{cases} \sqrt{\frac{1}{r}} & n=0 \\ \sqrt{\frac{2}{r}} \sin(n+1)\pi r & n \text{ 为奇数} \\ \sqrt{\frac{2}{r}} \cos(n)\pi r & n \text{ 为偶数} \end{cases} \quad (2)$$

圆谐-傅里叶矩是将函数图像投影在圆谐-傅里叶矩的基函数上得到的系数,它在极坐标下的表达式可写为:

$$\phi_{nm} = \frac{1}{2\pi} \int_0^{2\pi} \int_0^1 f(r, \theta) T_n(r) \exp(-jm\theta) r dr d\theta \quad (3)$$

其中, ϕ_{nm} 是图像函数 $f(r, \theta)$ 的圆谐-傅里叶矩。

指数矩是以更为简洁的复指数函数代替三角函数,从而实现的一种基于复指数构建的圆谐-傅里叶矩。

根据欧拉公式:

$$e^{-j\theta} = \cos \theta + j \sin \theta \quad (4)$$

可知,正弦函数和余弦函数可以表示为复指数函数的形式,将圆谐-傅里叶矩径向基函数中的三角函数用复指数形式表示,即

$$A_n(r) = \sqrt{\frac{2}{r}} \exp(j2n\pi r), n = -\infty, \dots, 0, \dots, \infty \quad (5)$$

其中, n 为整数, r 的取值范围 $0 \leq r \leq 1$ 。

用复指数形式的径向基函数替换三角形式的径向基函数可以得到:

$$Q_{nm}(r, \theta) = A_n(r) \exp(jm\theta) \quad (6)$$

其中, n 为整数, m 为整数, r 的取值范围 $0 \leq r \leq 1$, θ 的取值范围 $0 \leq \theta \leq 2\pi$ 。

对图像函数 $f(r, \theta)$, 称其在基函数为 $Q_{nm}(r, \theta)$ 上的展开系数 E_{nm} 为 (n, m) 阶指数矩 (Complex Exponent Moments, CEMs), 即

$$E_{nm} = \frac{1}{2\pi} \int_0^{2\pi} \int_0^1 f(r, \theta) \sqrt{\frac{2}{r}} \exp(j2n\pi r - jm\theta) r dr d\theta \quad (7)$$

3. 指数矩重构图像及仿真分析

传统的关于指数矩的计算是在直角坐标系下进行的。利用极坐标到直角坐标的转换公式[5-7], 指数矩在直角坐标系下的表达式:

$$E_{nm} = \frac{1}{4\pi} \iint_{x^2+y^2 \leq 1} f(x, y) \sqrt{\frac{2}{r_{x,y}}} \exp(-j2n\pi r_{x,y}) \exp(-jm\theta_{x,y}) dx dy \quad (8)$$

$$\begin{aligned} \text{其中, } x &= r \cos \theta, y = r \sin \theta, \\ r_{x,y} &= \sqrt{x^2 + y^2}, \theta_{x,y} = \arctan \frac{y}{x}, dx dy = r dr d\theta \end{aligned}$$

对于 $N \times N$ 的图像 $f(i, j)$, 对其进行归一化, 把像素坐标 (i, j) 变换为外接单位圆内的坐标 (x_j, y_i) , 其中圆心位于图像中心。则:

$$c_1 = \frac{\sqrt{2}}{N}, c_2 = \frac{N+1}{\sqrt{2}N}, x_j = c_1 j - c_2, y_i = c_2 - c_1 i。$$

(x_j, y_i) 满足 $0 \leq x^2 + y^2 \leq 1$, 每个像素所在的小区域的面积为 $\Delta x \Delta y = \frac{\sqrt{2}}{N} \times \frac{\sqrt{2}}{N}$ 。每个像素所在的小区

域内, 积分变量和被积函数的采样点为 (x_j, y_i) , 像素 (i, j) 点的图像的函数值为 $f(i, j)$, 指数矩的基函数值的表达式:

$$Q_{nm}(i, j) = A_n(r_{i,j}) \exp(jm\theta_{i,j}) \quad (9)$$

其中:

$$r_{i,j} = \sqrt{(c_1 j - c_2)^2 + (c_2 - c_1 i)^2}, \theta_{i,j} = \arctan \frac{c_2 - c_1 i}{c_1 j - c_2}。$$

在直角坐标下指数矩离散形式的表达式:

$$E_{nm} = \frac{1}{2\pi N^2} \sum_{i=1}^N \sum_{j=1}^N f(i, j) A_n^*(r_{i,j}) \exp(-jm\theta_{i,j}) \quad (10)$$

对图像函数 $f(i, j)$, 计算出指数矩后, 利用有限个指数矩近似重构图像函数 $f(i, j)$ 。

在极坐标系下, 利用有限个指数矩去近似重构图像函数的表达式:

$$f(r, \theta) \approx \sum_{n=-n_{\max}}^{n_{\max}} \sum_{m=-m_{\max}}^{m_{\max}} E_{nm} A_n(r) \exp(jm\theta) \quad (11)$$

直角坐标下重构图像函数的表达式:

$$f(i, j) \approx \sum_{n=-n_{\max}}^{n_{\max}} \sum_{m=-m_{\max}}^{m_{\max}} E_{nm} A_n(r_{i,j}) \exp(jm\theta_{i,j}) \quad (12)$$

在MATLAB仿真平台下, 指数矩重构图像仿真结果如图1所示:

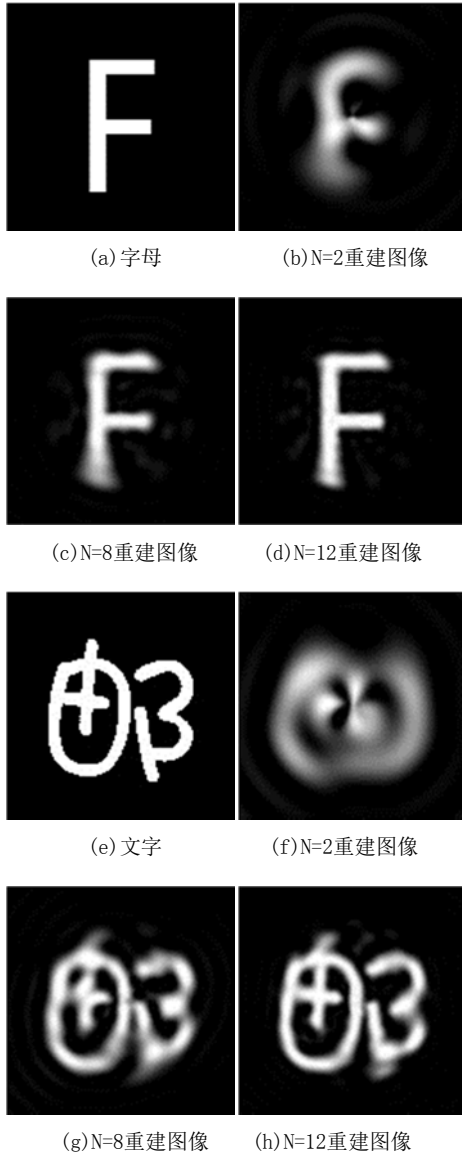


图1 对字母、文字分别利用指数矩在不同阶数下重建图像。

图1是分别对字母和文字图像利用不同阶数的指数矩重建的图像。图1(a)是原始字母图像, 图1(b)、(c)和(d)是利用阶数分别为2、8和12的指数矩对图1(a)进行重建的图像。图1(e)是原始文字图像, 图1(f)、(g)和(h)是利用阶数分别为2、8和12的指数矩对图1(e)进行重建的图像。

4. 二维快速傅里叶变换FFT计算指数矩

将直角坐标 (x, y) 转换为像素坐标 (i, j) 的表达式:

$$i = -\left\lfloor y \times \frac{N}{\sqrt{2}} \right\rfloor + \frac{N}{2} + 1 \quad (13)$$

$$j = \left\lfloor x \times \frac{N}{\sqrt{2}} \right\rfloor + \frac{N}{2} \quad (14)$$

其中, $\lfloor x \rfloor$ 是取整运算, 取大于等于 x 的整数。

极坐标下图像函数表达式[8, 9]:

$$f_p(r, \theta) = f\left(-\left\lfloor r \times \frac{N}{\sqrt{2}} \times \sin \theta \right\rfloor + \frac{N}{2} + 1, \left\lfloor r \times \frac{N}{\sqrt{2}} \times \cos \theta \right\rfloor + \frac{N}{2}\right) \quad (15)$$

指数矩的积分区域为单位圆内部, 采用变量等距离离散的方法将两个变量都均匀分成 M 份, 把积分式离散为求和运算。首先, 将单位圆内的区域沿径向 r 平均分为 M 份, 每份都是同心圆环, 沿半径 r 方向的间隔为 $\Delta r = 1/M$, 再将每个同心圆环沿角向 θ 平均分为 M 份, 每份的角向间隔为 $\Delta \theta = 2\pi/M$, 其中 M 是固定值, 是在仿真实验预先设定的一个非负整数。在每个小区域上, 任选一点计算积分函数的函数值, 选取每个小区域在径向和角向的起始点位置计算积分函数的函数值, 每个小区域沿径向和角向的起始点位置如下:

$$r_u = \frac{u}{M}, u = 0, 1, \dots, M-1 \quad (16)$$

$$\theta_v = \frac{2\pi v}{M}, v = 0, 1, \dots, M-1 \quad (17)$$

然后取 (r_u, θ_v) 计算离散的积分函数 $f_p(r_u, \theta_v)$, 得到离散积分函数 $f_p(r_u, \theta_v)$ 的表达式如下:

$$f_p(r_u, \theta_v) = f\left(-\left\lfloor r_u \times \frac{N}{\sqrt{2}} \times \sin \theta_v \right\rfloor + \frac{N}{2} + 1, \left\lfloor r_u \times \frac{N}{\sqrt{2}} \times \cos \theta_v \right\rfloor + \frac{N}{2}\right) \quad (18)$$

转化为求和的形式:

$$E_{nm} = \frac{1}{M^2} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} f_p(r_u, \theta_v) \sqrt{\frac{r_u}{2 \times M}} \exp(-j \frac{2\pi}{M} nu) \exp(-j \frac{2\pi}{M} mv) \quad (19)$$

令:

$$G_p(r_u, \theta_v) = f_p(r_u, \theta_v) \sqrt{\frac{r_u}{2 \times M}} \quad (20)$$

得:

$$E_{nm} = \frac{1}{M^2} \sum_{u=0}^{M-1} \sum_{v=0}^{M-1} G_p(r_u, \theta_v) \exp(-j \frac{2\pi}{M} nu) \exp(-j \frac{2\pi}{M} mv) \quad (21)$$

观察式(21), E_{nm} 为 $G_p(r_u, \theta_v)$ 的傅里叶变换,

即为 $f_p(r_u, \theta_v)$ 的指数矩。

根据上述分析,把利用二维快速傅里叶变换计算图像的指数矩的过程总结为以下三点:

- (1) 计算极坐标下的图像函数 $f_p(r_u, \theta_v)$;
- (2) 计算函数 $G_P(r_u, \theta_v)$;
- (3) 将函数 $G_P(r_u, \theta_v)$ 进行二维快速傅里叶变换,其结果就是在极坐标下计算的图像函数指数矩。

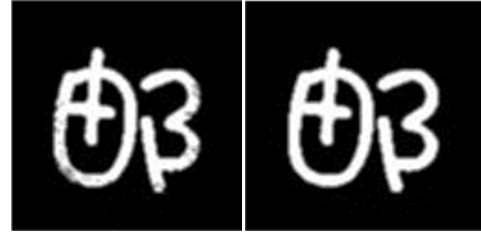
5. 快速傅里叶算法的仿真实验

利用有限个指数矩近似重建图像,重建图像的表达式:

$$\begin{aligned} f_{rec}(i, j) &\approx \sum_{n=-n_{\max}}^{N_{\max}} \sum_{m=-m_{\max}}^{M_{\max}} E_{nm} Q_{nm}(x_j, y_i) \\ &= \sum_{n=-n_{\max}}^{N_{\max}} \sum_{m=-m_{\max}}^{M_{\max}} E_{nm} A_n(r_{i,j}) \exp(jm\theta_{i,j}) \end{aligned} \quad (22)$$

其中, $f_{rec}(i, j)$ 是利用有限个指数矩近似重建的图像函数, N_{\max} 是径向基函数的最大阶数, M_{\max} 是角向基函数的最大阶数, E_{nm} 为指数矩。

在MATLAB仿真平台下,指数矩重构图像仿真结果如图2所示:



(g) N=8重建图像 (h) N=12重建图像

图2 对字母、文字分别利用指数矩FFT算法在不同阶数下重建图像。

图2是分别对字母和文字图像利用不同阶数的指数矩FFT算法重建的图像。图2(a)是原始字母图像,图2(b)、(c)和(d)是利用阶数分别为2、8和12的指数矩对图2(a)进行重建的图像。图2(e)是原始文字图像,图2(f)、(g)和(h)是利用阶数分别为2、8和12的指数矩对图2(e)进行重建的图像。

6. 传统算法与快速傅里叶算法的重建图像比较

6.1. 两种算法的重建图像误差比较

由仿真实验可以看出,在相同阶数下,利用二维快速傅里叶算法计算指数矩重建的图像质量要优于传统算法。

为了更好的对比原图像和重建图像的图像相似度,定义了原图像与重建图像之间的图像重建误差 ε 的表达式[10]:

$$\varepsilon = \frac{\sum_{i=1}^M \sum_{j=1}^N [f(i, j) - f_{rec}(i, j)]^2}{\sum_{i=1}^M \sum_{j=1}^N [f(i, j)]^2} \quad (23)$$

其中, $f(i, j)$ 是原图像函数, $f_{rec}(i, j)$ 是重建图像函数。

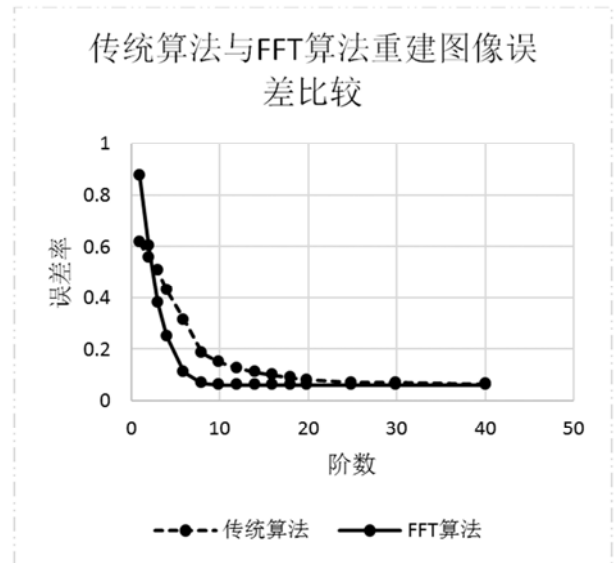


图3 传统算法与FFT算法重建图像误差比较。

图3显示了利用传统与FFT算法重建图像阶数与误差率大小 ϵ 的关系。可以看出, 利用FFT算法计算指数矩的重建图像在10阶的时候误差率就能达到0.06, 而传统算法要在21阶才能达到。利用FFT算法计算指数矩的重建图像在较低的阶数下就能较好的重构图像。通过对比实验结果证明了利用FFT算法计算指数矩是一种计算精度更高的算法。

6.2. 两种算法计算指数矩的耗时比较

表1给出了传统算法与FFT算法计算指数矩的耗时比较。可以看出, 随着阶数的增加, 两种算法耗时差距越来越大, 利用FFT算法计算指数矩的耗时明显优于传统算法。

表1 传统算法与FFT算法计算指数矩耗时比较。

阶数N	5	10	15	20	25	30	35	40
传统算法(s)	42.9	93.9	177.4	294.4	345.5	534.3	696.1	867.5
FFT算法(s)	0.5	0.9	1.8	3.1	4.7	7.2	10.2	14.1

7. 结论

文中从指数矩的定义和计算出发, 研究基于指数矩的图像描述和重构方法, 提出一种利用二维快速傅里叶FFT算法计算的指数矩, 并进行图像重构。在同等硬件平台下, 利用MATLAB软件进行图像重构仿真实验, 结果表明, 基于FFT算法计算指数矩能够较好的重构图像, 与传统算法相比, 利用FFT算法计算指数矩计算速度更快。通过计算重构图像与原始图像之间的误差, 表明FFT算法比传统算法的计算更精确。

致谢

本文受北京市教委“青年英才计划”项目资助。项目编号: YETP1957。

参考文献

- [1] 孟敏, 平子良. 基于指数矩的图像分解和重建[J]. 内蒙古师范大学学报: 自然科学汉文版, 2011, 40(03): 258-260.
- [2] 平子良, 盛云龙. 用切比雪夫图像矩描述图像[J]. 内蒙古师范大学学报: 自然科学汉文版, 2002, 31(3): 216-219.
- [3] 银国瑞. 基于圆谐-傅里叶矩的抗几何攻击数字水印算法研究[D]. 内蒙古师范大学物理与电子信息学院, 2009.
- [4] Jia-li Dong, Guo-rui Yin, Zi-liang Ping. Geometrically robust image watermarking based of Jacobi-Fourier moments[J]. Optoelectronics Letters. 2009(5)
- [5] 姜永静. 指数矩及其在模式识别中的应用[D]. 北京邮电大学2011.
- [6] 孟敏. 基于指数矩的抗几何攻击数字图像水印[D]. 内蒙古师范大学, 2011.
- [7] 孙景峰, 刘慧英, 纪超, 郭慧娟. 高分辨率图像的指数矩描述[J]. 应用科学学报, 2016, 34(2): 127-134.
- [8] PING Z L, JIANG Y J. An algorithm for computation of radial-harmonic-Fourier moments [C]//The Second International Conference on Functional Manufacturing Technologies (ICFMT2010), 2010: 254-288.
- [9] PING Z, JIANG Y, ZHOU S, WU Y. FFT algorithm of complex exponent moments and its application in image recognition [C]//The Sixth International Conference on Digital Image Processing. International Society for Optics and Photonics (ICDIP2014), 2014: 4177-4180.
- [10] 杨帆等. 数字图像处理与分析[M]. 北京: 北京航空航天大学出版社, 2010.