

Legendre 矩的两种快速算法

秦 磊¹, 舒华忠¹, 於文雪¹, 金丰华¹, C. Toumoulin², 罗立民¹,
(1. 东南大学生物科学与医学工程系, 南京 210096; 2. 法国雷恩一大信号与图像处理实验室)

摘 要: Legendre 正交矩在模式识别、图像分析等许多领域都有成功的应用, 但是由于其计算的复杂性, 相关的快速算法的研究尚未得到很好的解决. 本文针对一类采用图像块方法描述的图像, 提出两种快速、有效的计算 Legendre 矩的新方法, 它们分别是累加方法和积分方法. 这两种方法都有效地降低了计算复杂度, 缩短了计算时间.

关键词: Legendre 正交矩; 图像块描述方法; 快速算法

中图分类号: TP301 **文献标识码:** A **文章编号:** 0372-2112 (2004) 01-0025-04

Two New Algorithms for Fast Computation of Legendre Moments

QIN Lei¹, SHU Hua zhong¹, JIN Feng-hua¹, TOUMOULIN Christine², LUO Li-min¹
(1. Dept. of Biomedical Engineering, Southeast University, Nanjing, Jiangsu 210096, China;
2. Laboratory of Signal and Image Processing, University of Rennes I, Rennes 35042, France)

Abstract: Orthogonal moments have been successfully used in the field of pattern recognition and image analysis. However, due to its complexity, the problem of fast computation of orthogonal moments has not been well solved. In this paper, on the basis of representing images with blocks, we present two algorithms, cumulative method and integral method, for fast and effective computation of the two dimensional Legendre moments. The numerical results show that the new algorithms can decrease the computational complexity and computation time significantly.

Key words: Legendre moments; image block representation; fast algorithm

1 引言

自从 Hu^[1] 提出矩的不变量以来, 矩在模式识别、图像分析等许多领域都有广泛的应用. 迄今为止, 常见的矩描述子大致可以分为以下几种: 几何矩、正交矩 (Legendre 矩、Zernike 矩)、复数矩和旋转矩. 其中几何矩^[2,3] 提出的时间最早且形式简单, 对它的研究最为充分. 而正交矩描述的图像虽然具有最少的信息冗余度, 但由于它们的复杂性, 有关正交矩的快速算法的研究还很少^[4-6]. Shu 等人^[5] 针对边界为多边形的二值图像采用格林公式将面积分转化为对边界的线积分, 然后采用迭代方法进行计算, 使计算复杂度由 $O(N^2)$ 降低到 $O(N)$; Zhou 等人^[6] 把基于像素点的二维 Legendre 矩转换为线段的形式来计算, 在计算出所有线段的积分后, 使用扩展的 Hatamian 滤波方法来计算一维的 Legendre 矩, 也有效地缩短了计算时间. Spiliotis 等人^[2] 将图像块描述方法应用到几何矩的计算中, 减少了计算复杂度, 缩短了计算时间. 本文旨在把图像块描述方法推广到 Legendre 矩的计算中, 并在此基础上提出两种快速算法.

本文组织如下: 第 2 节介绍图像块描述方法, 第 3 节介绍两种基于图像块描述方法的 Legendre 矩快速算法, 第 4 节分析和讨论方法的有效性 with 精确性.

2 图像块描述方法

考虑如下二值图像

$$f(x, y) = \begin{cases} 1, & (x_i, y_j) \in \Omega \\ 0, & \text{其它} \end{cases}$$

其中 Ω 为目标区域.

图像块描述方法的定义为:

(1) 图像块是图像中的一个矩形区域, 边平行于图像的轴, 并且一个块中的象素具有相同的灰度值.

(2) 如果一幅图像用图像块来表示, 并且每个象素都属于且只属于一个块, 那就说这个图像是用块来描述的.

值得指出的是, 由于块是矩形的, 所以很显然最小的块只包含一个象素.

对图像分块可分为四个步骤:

步骤 1: 对图 f 的每一行 y 进行扫描, 找到第 y 行的目标区间.

步骤 2: 将这些区间和那些含有第 $y-1$ 行象素的块进行比较.

步骤 3: 如果一个区间和任何块都不匹配, 那么这就是一个新块.

步骤 4: 如果该区间和某个块匹配, 那么此块的最后一行

就是第 y 行.

循环这四个步骤直至图像最后一行.

通过上述方法,我们就可以得到一系列的块,用块来描述这幅二值图像就是

$$f(x, y) = \{ b_i; i = 0, 1, \dots, K-1 \} \quad (1)$$

其中 K 是块的个数. 对每个块我们都记录下其左上角坐标和右下角坐标.

3 Legendre 矩的计算

对于一密度函数为 $f(x, y)$ 的二维图像, 其 $(p+q)$ 阶 Legendre 正交矩的定义为:

$$\lambda_{pq} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^1 \int_{-1}^1 P_p(x) P_q(y) f(x, y) dx dy \quad (2)$$

式中 $P_p(x)$ 为 p 阶 Legendre 多项式, 其定义为:

$$P_p(x) = \frac{1}{2^p} \sum_{k=0}^{\lfloor p/2 \rfloor} (-1)^k \frac{(2p-2k)!}{k! (p-k)! (p-2k)!} x^{p-2k}, \quad x \in [-1, 1] \quad (3)$$

对于数字图像, 双重积分可以由双重求和来近似, 通常采用如下形式:

$$\lambda_{pq} = \frac{(2p+1)(2q+1)}{4} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} f(x_i, y_j) P_p(x_i) P_q(y_j) \Delta x \Delta y \quad (4)$$

其中 $\Delta x = x_{i+1} - x_i$, $\Delta y = y_{j+1} - y_j$, 分别代表在 x 和 y 方向上的取样间隔.

对 $N \times N$ 的矩形二值图像来说, λ_{pq} 就可简化为:

$$\lambda_{pq} = \frac{(2p+1)(2q+1)}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} P_p(x_i) P_q(y_j) \quad (5)$$

在计算过程中, 我们使用其迭代性质:

$$P_{n+1}(x) = \frac{2n+1}{n+1} x P_n(x) - \frac{n}{n+1} P_{n-1}(x), \quad n \geq 1 \quad (6)$$

其中 $P_0(x) = 1$, $P_1(x) = x$.

3.1 累加方法

对二值图像进行分块, 如果分成 K 块, 那么 Legendre 矩的值就可以写成式(5)的变化形式

$$\lambda_{pq} = \sum_{i=0}^{K-1} \lambda_{pq}^i = \frac{(2p+1)(2q+1)}{N^2} \sum_{i=0}^{K-1} \sum_{x_1, b_i}^{x_2, b_i} \sum_{y_1, b_i}^{y_2, b_i} P_p(x) P_q(y) \quad (7)$$

其中 (x_1, b_i, y_1, b_i) 和 (x_2, b_i, y_2, b_i) 分别是一个块的左上角和右下角坐标.

由于块为矩形, 故而式(7)中

$$\begin{aligned} \lambda_{pq}^i &= \sum_{x=x_1, b_i}^{x_2, b_i} \sum_{y=y_1, b_i}^{y_2, b_i} P_p(x) P_q(y) \\ &= \sum_{x=x_1, b_i}^{x_2, b_i} P_p(x) [P_q(y_1, b_i) + P_q(y_1, b_i + 1) + \dots + P_q(y_2, b_i)] \\ &= \sum_{x=x_1, b_i}^{x_2, b_i} P_p(x) \sum_{y=y_1, b_i}^{y_2, b_i} P_q(y) = s_p(b_i) s_q(b_i) \end{aligned} \quad (8)$$

于是求 K 阶 Legendre 矩的问题就转化为先求出每个块的式(8)的值, 然后将各块的值相加, 再乘以系数 $\frac{(2p+1)(2q+1)}{N^2}$ 的问题.

对于式(8)的计算可以分成两部分, 即:

$$s_p(b_i) = \sum_{x=x_1, b_i}^{x_2, b_i} P_p(x), \quad s_q(b_i) = \sum_{y=y_1, b_i}^{y_2, b_i} P_q(y) \quad (9)$$

由于式(9)中两者的计算方法一样, 因此我们只考虑其中的一个求和式.

记 $\delta_{b_i} = x_2, b_i - x_1, b_i + 1$ 为块的长度即像素点个数, 则

$$s_p(b_i) = \sum_{j=0}^{\delta_{b_i}-1} P_p(x_1, b_i + j \Delta x) \quad (10)$$

采用 Shu 等人^[5]提出的计算 Legendre 多项式的公式:

$$P_p(x+a) = \sum_{l=0}^p \mathcal{X}_l^+(a) P_{p-l}(x), \quad \mathcal{X}_l^+(\Delta x) = 1, \quad \forall p \geq 0 \quad (11)$$

其中 $\mathcal{X}_l^+(a)$ 的表达式见文^[5], 可以得到:

$$\begin{aligned} &P_{p+1}(x_1, b_i + (j+1)\Delta x) - P_{p+1}(x_1, b_i + j\Delta x) \\ &= P_{p+1}(x_1, b_i + \Delta x + j\Delta x) - P_{p+1}(x_1, b_i + j\Delta x) \\ &= \sum_{l=0}^{p+1} \mathcal{X}_l^{+1}(\Delta x) P_{p+1-l}(x_1, b_i + j\Delta x) - P_{p+1}(x_1, b_i + j\Delta x) \\ &= \sum_{l=1}^{p+1} \mathcal{X}_l^{+1}(\Delta x) P_{p+1-l}(x_1, b_i + j\Delta x) \end{aligned} \quad (12)$$

将式(12)等式两边从 $i=0$ 到 $(\delta_{b_i}-1)$ 求和得到:

$$\begin{aligned} &P_{p+1}(x_1, b_i + \delta_{b_i}\Delta x) - P_{p+1}(x_1, b_i) \\ &= \sum_{j=0}^{\delta_{b_i}-1} \sum_{l=1}^{p+1} \mathcal{X}_l^{+1}(\Delta x) P_{p+1-l}(x_1, b_i + j\Delta x) \end{aligned} \quad (13)$$

引入记号

$$s'_p(b_i) = P_{p+1}(x_1, b_i + \delta_{b_i}\Delta x) - P_{p+1}(x_1, b_i) \quad (14)$$

于是

$$\begin{aligned} s'_p(b_i) &= \sum_{l=1}^{p+1} \mathcal{X}_l^{+1}(\Delta x) \sum_{j=0}^{\delta_{b_i}-1} P_{p+1-l}(x_1, b_i + j\Delta x) \\ &= \sum_{l=1}^{p+1} \mathcal{X}_l^{+1}(\Delta x) s_{p+1-l}(b_i) \end{aligned} \quad (15)$$

记 $S_L(b_i) = (s_0(b_i), s_1(b_i), \dots, s_L(b_i))^T$,

$$S'_L(b_i) = (s'_0(b_i), s'_1(b_i), \dots, s'_L(b_i))^T$$

于是一维向量 $S_L(b_i)$ 和 $S'_L(b_i)$ 的关系为:

$$S'_L(b_i) = A_{L+1} S_L(b_i) \quad (16)$$

其中

$$A_{L+1} = \begin{bmatrix} \mathcal{X}_1^+(\Delta x) & 0 & 0 & 0 \\ \mathcal{X}_2^+(\Delta x) & \mathcal{X}_1^+(\Delta x) & 0 & 0 \\ \mathcal{X}_3^+(\Delta x) & \mathcal{X}_2^+(\Delta x) & \mathcal{X}_1^+(\Delta x) & 0 \\ \dots & \dots & \dots & \dots \\ \mathcal{X}_{L+1}^+(\Delta x) & \mathcal{X}_L^+(\Delta x) & \dots & \mathcal{X}_1^+(\Delta x) \end{bmatrix} \quad (17)$$

由于矩阵所有对角线元素皆为非零值, 因此 A_{L+1} 是非奇异矩阵, 所以有:

$$S_L(b_i) = A_{L+1}^{-1} S'_L(b_i) \quad (18)$$

由此可以看出只需要计算 $S_L(b_i)$ 就可以得到我们所需要的 $S'_L(b_i)$ 了。

3.2 积分方法

一幅具有 K 个图像块的二值图像, 记 x 方向第 b_i 块长度为 $\delta_{b_i} = x_{2, b_i} - x_{1, b_i} + 1$, y 方向块的长度为 $\delta'_{b_i} = y_{2, b_i} - y_{1, b_i} + 1$, 式(2)可写成:

$$\begin{aligned} \lambda_{pq} &= \frac{(2p+1)(2q+1)}{4} \sum_{i=0}^{K-1} \int_{x_{1, b_i} - \frac{1}{2}\Delta x}^{x_{1, b_i} + \delta_{b_i} - \frac{1}{2}\Delta x} \int_{y_{1, b_i} - \frac{1}{2}\Delta y}^{y_{1, b_i} + \delta'_{b_i} - \frac{1}{2}\Delta y} P_p(x) P_q(y) dx dy \\ &= \frac{(2p+1)(2q+1)}{4} \sum_{i=0}^{K-1} \int_{x_{1, b_i} - \frac{1}{2}\Delta x}^{x_{1, b_i} + \delta_{b_i} - \frac{1}{2}\Delta x} P_p(x) dx \int_{y_{1, b_i} - \frac{1}{2}\Delta y}^{y_{1, b_i} + \delta'_{b_i} - \frac{1}{2}\Delta y} P_q(y) dy \end{aligned} \quad (19)$$

根据 Legendre 多项式的积分性质:

$$\int P_p(x) dx = \frac{1}{2p+1} (P_{p+1}(x) - P_{p-1}(x)) \quad (20)$$

容易得到

$$\begin{aligned} &\int_{x_{1, b_i} - \frac{1}{2}\Delta x}^{x_{1, b_i} + \delta_{b_i} - \frac{1}{2}\Delta x} P_p(x) dx \\ &= \frac{1}{2p+1} \left\{ P_{p+1}(x_{1, b_i} + \delta_{b_i} - \frac{1}{2}\Delta x) - P_{p-1}(x_{1, b_i} + \delta_{b_i} - \frac{1}{2}\Delta x) \right. \\ &\quad \left. - P_{p+1}(x_{1, b_i} - \frac{1}{2}\Delta x) + P_{p-1}(x_{1, b_i} - \frac{1}{2}\Delta x) \right\} \end{aligned} \quad (21)$$

记

$$\begin{aligned} S_{x, b_i} &= P_{p+1}(x_{1, b_i} + \delta_{b_i} - \frac{1}{2}\Delta x) - P_{p-1}(x_{1, b_i} + \delta_{b_i} - \frac{1}{2}\Delta x) \\ &\quad - P_{p+1}(x_{1, b_i} - \frac{1}{2}\Delta x) + P_{p-1}(x_{1, b_i} - \frac{1}{2}\Delta x) \\ S_{y, b_i} &= P_{q+1}(y_{1, b_i} + \delta'_{b_i} - \frac{1}{2}\Delta y) - P_{q-1}(y_{1, b_i} + \delta'_{b_i} - \frac{1}{2}\Delta y) \\ &\quad - P_{q+1}(y_{1, b_i} - \frac{1}{2}\Delta y) + P_{q-1}(y_{1, b_i} - \frac{1}{2}\Delta y) \end{aligned} \quad (22)$$

4 讨论与结论

4.1 算法的计算复杂度

直接计算 Legendre 矩的时间开销是巨大的, 假定一个 $N \times N$ 的图像块, 直接计算到 L 阶 Legendre 矩需要 $O(N^2 L^2)$ 次加法和乘法。

分块后的累加方法在用式(18)计算出矢量 $S_L(b_i) = (s_0(b_i), s_1(b_i), \dots, s_L(b_i))^T$ 需 $4L$ 次加法和 $8L$ 次乘法, 每个系数 $\frac{(2p+1)(2q+1)}{N^2}$ 需 2 次加法和 5 次乘法, 因此总共需要:

$$\begin{aligned} \text{加法} & 4L \times 2 + 2 \times \frac{(L+1)(L+2)}{2} = L^2 + 11L + 2 \\ \text{乘法} & 8L \times 2 + 5 \times \frac{(L+1)(L+2)}{2} + \frac{(L+1)(L+2)}{2} \\ & = 3L^2 + 25L + 6 \end{aligned}$$

分块后的积分方法首先利用式(6)计算 Legendre 多项式值, 需要 $3L$ 次加法和 $5L$ 次乘法, 再利用式(22)计算所有矩的值, 共需要:

$$\text{加法} \quad 3L \times 2 + 3L \times 2 = 12L$$

$$\text{乘法} \quad 5L \times 2 + 2 \frac{(L+1)(L+2)}{2} = L^2 + 13L + 2$$

分块后的累加方法和积分方法都大大减小了计算复杂度, 并且它们的计算复杂度都与图像大小无关, 这样在计算比较大的图像的时候就更能显示出这两种方法的优越性了。

4.2 时间比较

我们对直接利用式(4)计算 Legendre 矩、分块后的累加方法以及分块后的积分方法这三种方法计算的时间进行比较, 首先使用 50×50 的汉字“幕”进行计算, 如图 1。



比较结果如表 1 所示:

图 1 50×50 汉字图像

表 1 三种计算方法计算 50×50 汉字“幕”时间表

Legendre 矩的阶数	直接计算方法	分块后的累加方法	分块后的积分方法
20	0.06s	0.01s	0.00s
30	0.13s	0.01s	0.00s
40	0.18s	0.01s	0.00s
50	0.29s	0.01s	0.00s
55	0.34s	0.01s	0.00s
60	0.39s	0.02s	0.00s
65	0.45s	0.02s	0.00s
70	0.52s	0.02s	0.00s

从上面的表中可以看出文中提出的两种方法的确缩短了计算时间, 下面我们将几种方法应用到 256×256 大小的汉字“幕”上, 进行计算时间的比较, 结果如表 2 所示。

表 2 几种计算方法计算 256×256 汉字“幕”时间表

Legendre 矩的阶数	直接计算方法	分块后的累加方法	分块后的积分方法
20	0.86s	0.01s	0.00s
30	1.75s	0.01s	0.00s
40	2.85s	0.01s	0.00s
50	4.24s	0.01s	0.00s
55	4.98s	0.02s	0.00s
60	6.13s	0.02s	0.00s
65	7.18s	0.02s	0.00s
70	7.68s	0.02s	0.00s

从上述结果中可以发现, 直接方法在图像大小增大时计算时间增加了很多, 而分块累加方法和积分方法的计算时间则不随图像大小的变化而变化, 因此这两种算法在计算大图的时候特别有效, 上述实验结果与我们在(4.1)节中的理论分析也是一致的。

4.3 误差分析

为了说明算法的精度, 我们构造一幅在 $[-1, 1] \times [-1, 1]$

1] 的区域内所有像素点值均为 1 的图像(50×50)。我们知道, 对于这一图像其 Legendre 矩值除了零阶矩为 1 外, 其余都为 0。基于这点, 我们对累加方法和积分方法的误差进行比较。定义方差:

$$E_K = \sum_{p=0}^K \sum_{q=0}^{K-p} (L'_{pq} - L_{pq})^2 \quad (23)$$

其中 L'_{pq} 表示由累加方法或积分方法计算得到的矩, 而 L_{pq} 则表示除了 $L_{00}=1$, 其余都为 0 的精确值。

表 3 中显示了两种计算方法的方差:

表 3 两种不同方法计算矩的方差

矩的阶数	分块累加方法	分块积分方法
1	0.000000	0.000000
3	0.000002	0.000000
5	0.000074	0.000000
7	0.000733	0.000000
9	0.004003	0.000000
11	0.015351	0.000000
13	0.046213	0.000000

从上面这张表中我们可以发现, 分块积分方法计算矩得到的结果是非常精确的, 而累加方法则由于采用了近似公式计算, 结果不如积分方法精确。

4.4 总结

通过本文的讨论我们可以看出, 对图像采用合适的描述方法再配合以适当的快速算法会极大地缩短计算时间。文中分别对所提出的两种新的算法的计算复杂度、计算时间和计算误差进行了比较。分析结果表明直接方法计算复杂度最高, 即计算时间最长, 而分块后的积分方法计算复杂度最低, 即计算时间最短, 实验结果也验证了这一结论。分块累加方法计算时间明显小于直接方法, 而分块积分方法计算所需时间更少。另外, 分块积分方法可以精确的计算出 Legendre 矩的值, 这对 Legendre 矩的应用有重要的意义, 因而本文提出的方法是计算 Legendre 矩的快速、有效的方法。此外, 值得指出的是, 本文

采用的图像块描述方法较文献[6]中的 δ 方法是一种更有效的图像表征方法, 因为后者只对图像的 x 方向进行了处理, 而在 y 方向上未作任何处理(这也是我们在文献[6]中提出的两种方法所需计算时间基本一样的原因), 因此, 本文所提出的快速算法也较文献[6]中的快速算法更有效。

图像块描述方法也可以应用到灰度图像的计算中去。这样每个图像块就有 5 个参数: 左上角坐标、右下角坐标和灰度值。在计算每个块的时候由于灰度相同就可以将灰度提出最后计算, 这也将缩短计算时间。

参考文献:

- [1] Hu M K. Visual pattern recognition by moment invariants [J]. IRE Trans on Information Theory, 1962, 10: 179-187.
- [2] Spiliotis I M, Mertzios B G. Real time computation of two dimensional moments on binary images using image block representation [J]. IEEE Trans on Image Processing, 1998, 7(11): 1609-1615.
- [3] Flusser J. Refined moment calculation using image block representation [J]. IEEE Transactions on Image Processing, 2000, 9(11): 1977-1978.
- [4] Shu H Z, et al. An efficient method for computation of legendre moments [J]. Graphical Models, 2000, 62: 237-262.
- [5] Shu H Z, et al. A new fast method for computing Legendre moments [J]. Pattern Recognition, 2000, 33: 341-348.
- [6] Zhou J D, Shu H Z. Two new algorithms for efficient computation of Legendre moments [J]. Pattern Recognition, 2002, 35: 1143-1152.

作者简介:

秦 磊 女, 1979 年生于江苏南通, 东南大学生物医学工程系硕士, 主要研究方向为医学图像处理 and 模式识别。

舒华忠 男, 1965 年生于江西, 现为东南大学生物医学工程系教授, 博士生导师, 1992 年获法国雷恩大学应用数学博士学位, 研究工作主要包括放射治疗计划优化、医学图像处理和模式识别等。

於文雪 男, 1969 年生于山东, 现为东南大学生物医学工程系副教授, 主要研究方向为生物医学工程, 计算机辅助诊断与治疗等。