

由单幅二维灰度图像重构物体表面形状*

高月芳, 罗 飞, 曹建忠

(华南理工大学 自动化科学与工程学院, 广州 510640)

摘 要: 鉴于目前很少有论文讨论完整的由单幅二维灰度图像重构物体表面形状的算法, 包括它的控制参数的估计及算法的实现, 介绍了一种完整的 SFS 算法。它在考虑自遮掩影响的情况下, 有效地估计了 SFS 算法中涉及的各种控制参数, 并引入亮度约束、灰度梯度约束和可积性约束, 计算出表面高度和表面向量, 实现三维重构。与传统的算法相比, 本算法无论是在速度还是在精度方面都达到了比较高的水平, 具有一定的应用前景。最后还指出了在 MATLAB 中实现需要注意的问题。

关键词: 三维重构; 由明暗恢复形状; 参数估计; 变分法; 分层算法

中图分类号: TP319 文献标志码: A 文章编号: 1001-3695(2007) 08-0195-03

3D reconstruction from single 2D image of object

GAO Yue-fang, LUO Fei, CAO Jian-zhong

(Automation College, South China University of Technology, Guangzhou 510640, China)

Abstract: Since there have been few papers on a whole SFS algorithm, this paper introduced a SFS algorithm which includes the estimation of the control parameters and the implementation of the algorithm. Taking into consideration of the effect of self-shading, it can effectively estimate the control parameters used in the SFS algorithms. It implemented the brightness constraint, the intensity gradient constraint and the integrability constraint to compute the surface height and orientation and to reconstruct the surface shape. This algorithm performed better than a great deal of conventional algorithms in both speed and accuracy. This paper finally mentioned some problems that should be paid attention to in MATLAB.

Key words: 3D reconstruction; SFS(shape from shading) ; estimation of parameters; variational calculus; hierarchical algorithm

近几年, 由物体的二维灰度图像重构其表面三维形状 (SFS) 的方法在计算机视觉、图像处理等方面备受关注。国内外理论成果也主要发展为四种方法, 即最小值方法、演化方法、局部分析法和线性化方法^[1, 2]。但是国内很少有论文讨论完整的算法, 包括控制参数的估计以及算法的实现。例如, 文献 [3] 仅介绍了 SFS 算法, 而没有像 Zheng 和 Chellappa 那样讨论、匹配控制参数的估算方法^[4]。然而在各种算法中都需要涉及光源和曲面反射率等控制参数。当实验环境固定时, 这些参数可以是已知的。但为了使 SFS 算法能够更灵活地应用于不同工作、实验环境, 必须能够根据二维灰度图像有效地估计出这些控制参数。而且不同的算法可能需要匹配不同的参数估计方法^[5]。

1 成像模型

在 SFS 研究中, 通常采用图 1 所示的坐标系, 同时将物体表面高度表示为 $z = z(x, y)$ 。在 SFS 算法中, 为简化问题, 通常进行如下假设: 光源为无限远处点光源; 反射模型为朗伯体表面反射模型; 成像几何关系为正交投影^[3]。这样, 成像模型根据二维图像中某像素点的灰度值与物体表面对应点按照反射图函数得到的灰度值相等, 可表示为亮度方: $I(x, y) = R(p,$

$q)$ 。其中: I 为二维图像中像素点 (x, y) 的灰度值; 表面方向 $p = \partial z / \partial x, q = \partial z / \partial y$ 是物体表面点高度 z 关于图像坐标的偏导数; R 是反射图函数, 它表示感受的光亮度随表面方向的不同而发生变化, 具体形式如下:

$$R(p, q) = \eta(n \cdot s) =$$

$$\eta(\cos \gamma - p \cos \tau \sin \gamma - q \sin \tau \sin \gamma) / \sqrt{1 + p^2 + q^2}$$

其中: η 是曲面反射率, 在图像处理过程中设为常数; $n = (-p, q, 1) / \sqrt{1 + p^2 + q^2}$ 为表面点 (x, y, z) 处的法线; $s = (\cos \tau \sin \gamma, \sin \tau \sin \gamma, \cos \gamma)$ 为光源方向; τ 和 γ 分别为光源偏角、倾角。其意义如图 1 所示。

2 完整 SFS 算法

2.1 控制参数估计方法

参数可由式 (1) 估计得到^[4]

$$\tau = \arctan \left(\frac{E\{y_L / \sqrt{x_L^2 + y_L^2}\}}{E\{x_L / \sqrt{x_L^2 + y_L^2}\}} \right) \quad (1)$$

其中: E 为总体均值; $\begin{pmatrix} x_L \\ y_L \end{pmatrix} = (B B)^{-1} B^T d, \vec{d} = \begin{pmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{pmatrix}$;

$$B = \begin{pmatrix} \delta_1 & \delta_1 \\ \delta_2 & \delta_2 \\ \dots & \dots \\ \delta_N & \delta_N \end{pmatrix}, N \text{ 为选取的方向数, } \delta_i \text{ 为 } (\delta_i, \delta_i) \text{ 方向上的}$$

灰度变化值($i = 1, \dots, N$)。在实现过程中,因读入图像为离散数字形式, N 一般取图 2 所示的几个方向。

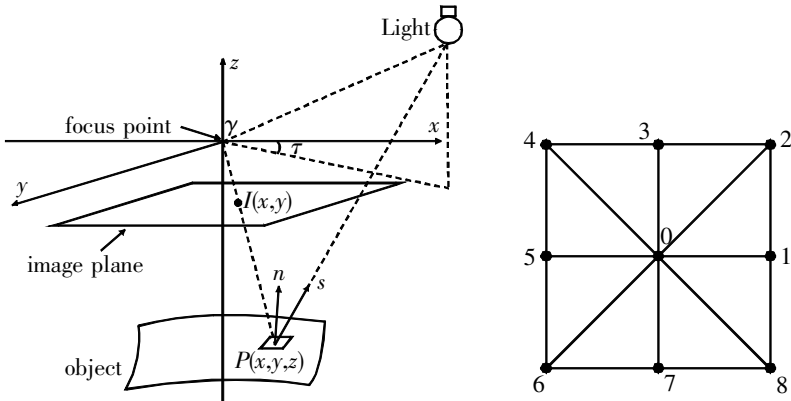


图 1 系统坐标示意图

图 2 方向选取示意图

即 $B' = \begin{pmatrix} 1 & \sqrt{2}/2 & 0 & -\sqrt{2}/2 & -1 & -\sqrt{2}/2 & 0 & \sqrt{2}/2 \\ 0 & \sqrt{2}/2 & 1 & \sqrt{2}/2 & 0 & -\sqrt{2}/2 & -1 & -\sqrt{2}/2 \end{pmatrix}$ 参数 γ 可由式(2)估计得到^[6]:

$$E\{I\} / \sqrt{E\{I^2\}} \approx \sum_{i=0}^7 a_i \cos^i \gamma \tag{2}$$

其中: 系数为 $a_0 = 0.5577$ $a_1 = 0.6240$ $a_2 = 0.1882$
 $a_4 = -0.6514$ $a_4 = -0.5350$ $a_5 = 0.9282$
 $a_6 = 0.3476$ $a_7 = -0.4984$

式(2)右边为 γ 的单调递减函数, 并且 $\gamma \in [0, \pi/2]$, 所以 γ 存在惟一解, 并且 $E\{I\} / \sqrt{E\{I^2\}} : 0.96191$ 时, 令 $\gamma = 0$ 。

参数 η 可由式(3)估计得到^[6]:

$$\eta = 1 / [f_1(\gamma) + f_2(\gamma)] \times [E\{I\} \times f_1(\gamma) + \sqrt{E\{I^2\}} \times f_2(\gamma)] \tag{3}$$

其中: $f_1(\gamma) \approx \sum_{i=0}^7 b_i \cos^i \gamma$, $f_2(\gamma) \approx \sum_{i=0}^7 c_i \cos^i \gamma$
 $b_0 = 0.1615$ $b_1 = 0.3959$ $b_2 = 0.3757$ $b_3 = -0.0392$
 $b_4 = -0.3077$ $b_5 = 0.1174$ $b_6 = 0.1803$ $b_7 = -0.0984$
 $c_0 = 0.0834$ $c_1 = 0.2169$ $c_2 = 0.2487$ $c_3 = 0.1836$
 $c_4 = 0.0048$ $c_5 = -0.1086$ $c_6 = -0.0043$ $c_7 = 0.0424$

2.2 算法的实现

将亮度方程变为误差函数形式, 并引入灰度梯度约束和可积性约束可得: $\int \int F(p, q, z) dx dy = \int \int [(I - R)^2 + (R_x - I_x)^2 + (R_y - I_y)^2] + \mu [(z_x - p)^2 + (z_y - q)^2] dx dy$

利用欧拉方程求泛函极小值可得(p, q, z)对应的迭代增量如下^[3, 4]:

$$\begin{cases} \delta p = (4/\Delta) [(C_1 - 1/4\mu C_3)(5R_q^2 + 4/5\mu) - (C_2 - 1/4\mu C_3)(5R_p R_q + 1/4\mu)] \\ \delta q = (4/\Delta) [(C_2 - 1/4\mu C_3)(5R_p^2 + 4/5\mu) - (C_1 - 1/4\mu C_3)(5R_p R_q + 1/4\mu)] \\ \delta z = (1/4)(C_3 + \delta p + \delta q) \end{cases} \tag{4}$$

其中:

$$C_1 = (-R + I + R_p p_{xx} + R_q q_{xx} - I_{xx} + R_p p_{yy} + R_q q_{yy} - I_{yy}) R_p - \mu(p - z_x)$$

$$C_2 = (-R + I + R_p p_{xx} + R_q q_{xx} - I_{xx} + R_p p_{yy} + R_q q_{yy} - I_{yy}) R_q - \mu(q - z_y)$$

$$C_3 = -p_x + z_{xx} - q_y + z_{yy}$$

$$\Delta = 4[(5R_p^2 + 5/4\mu)(5R_q^2 + 5/4\mu) - 5R_p R_q + 1/4\mu^2]$$

算法流程如图 3 所示。

3 实现中的问题

在算法的实际过程中, 为提高处理速度和精度, 本文特别提出了以下几个注意点。

3.1 分层 SFS 算法

分层算法^[3, 4]能够有效地简化图像的计算程度。在本分层 SFS 算法中, 最低分辨率的图像尺寸是 64 × 64, 相近层间的图像分辨率压缩系数为 2。应用于不同分辨率层的输入图像都是由给定的最高分辨率层图像通过平均每个单元的灰度值计算出的, 此时要注意归一化灰度值。分层算法的重要问题是把产生的结果在层间通信, 并且表面形状的描述在不同层间应一致。

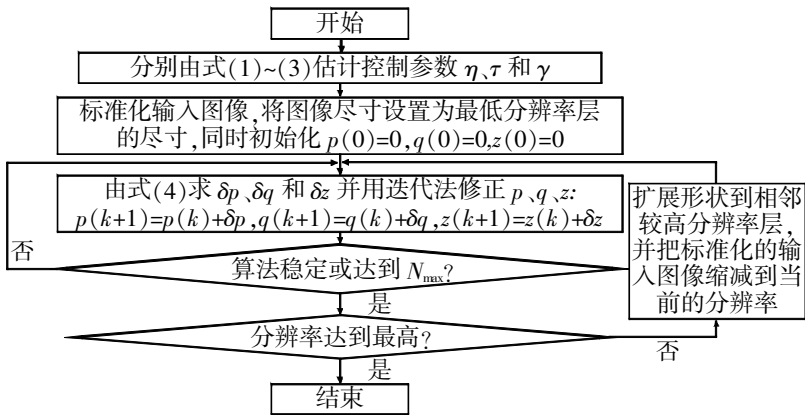


图 3 算法流程图

令标有(')的变量描述较高分辨率层的表面形状, 而没有标(')的变量描述较低分辨率层。较高分辨率层的表面描述由相邻低分辨率层的描述内插。其规则如下:

a) 对于 $i, j \in \{2, \dots, s\}$ (s 为较高分辨率层的图像尺寸)
 $(p', q', z)_{i,j} =$
 $(p, q, 2z)_{i/2, j/2}$ i 和 j 均为偶数
 $(1/2) [(p, q, 2z)_{(i+1)/2, j/2} + (p, q, 2z)_{(i-1)/2, j/2}]$ i 为奇数, j 为偶数
 $(1/2) [(p, q, 2z)_{i/2, (j+1)/2} + (p, q, 2z)_{i/2, (j-1)/2}]$ i 为偶数, j 为奇数
 $(1/4) [(p, q, 2z)_{(i+1)/2, (j+1)/2} + (p, q, 2z)_{(i+1)/2, (j-1)/2} + (p, q, 2z)_{(i-1)/2, (j+1)/2} + (p, q, 2z)_{(i-1)/2, (j-1)/2}]$ i 和 j 均为奇数
b) 对于边界
 $(p', q)_{1,j} = 2(p', q)_{2,j} - (p', q)_{3,j}$
 $z'_{1,j} = z'_{2,j} - p'_{1,j}$ 对于 $j \geq 2$
 $(p', q)_{i,1} = 2(p', q)_{i,2} - (p', q)_{i,3}$
 $z'_{i,1} = z'_{i,2} - p'_{i,1}$ 对于 $i \geq 2$
 $(p', q)_{1,1} = 2(p', q)_{2,2} - (p', q)_{3,3}$
 $z'_{1,1} = z'_{2,2} - p'_{1,1} + q'_{1,1}$

3.2 编程问题

在参数估计和三维重构过程中, 对于各像素的处理, 最简单明了的就是使用 for 语句。但即使是对一幅很低像素的图像, 使用 for 语句对各个像素进行处理, 也需要至少十几个小时的时间, 并且这个处理时间会随着图像像素的增多而快速地增长。在 MATLAB 中, 可以将整个图像作为一个矩阵来处理。特别是在估算参数时, 选用合适的 mask, 再利用 imfilter 函数求取各个 $\delta_i (i = 1, \dots, N)$, 当然此 δ_i 也是以整个图像矩阵为基本形式的。同时注意到系数($B^T B$)⁻¹ B 矩阵与各 δ_i 之间的固定系数匹配关系, 就可以得到(x_L, y_L)的矩阵。在这个矩阵上

求解式(1), 就可以估算出控制参数 τ 。这种方法是以 s 为时间单位的, 图像越大, 就越能体现这种方法的速度优越性。

4 实验结果

SFS 算法的测试图像的选取也是一道复杂的工序: 一幅好的测试图像必须满足算法的所有假设条件。对于合成图像, 满足这些条件并不难, 而实际图像却不可能, 这也是实际图像产生误差的原因之一。另外, 合成图像还有两方面的优势: 对于同一物体表面, 可以任意设定光源方向; 能够得到精确的深度信息, 便于比较和计算误差。它的弊端就是运行结果不可靠, 即对于合成图像运行良好的算法, 对于实际图像却可能有较大的偏差。为此, 本试验选取合成图像和实际图像各一幅。

根据文献[1], 图像坐标 (x, y) 被映射到 $[0, 127]$ 区域, 而高度 z 也相应地扩大了 128 倍。同时, 设定参数 $\mu=1$, 最高分辨率层的 $N_{\max}=500$ 。试验结果如图 4、5 和表 1 所示。由表 1 可以看出, 有的传统算法虽然速度比较快, 但是误差大、精度低。而本文的控制参数估算方法和 SFS 算法部分配合得相当完善, 使得整个算法在速度和精度两方面都达到了比较高的水平。

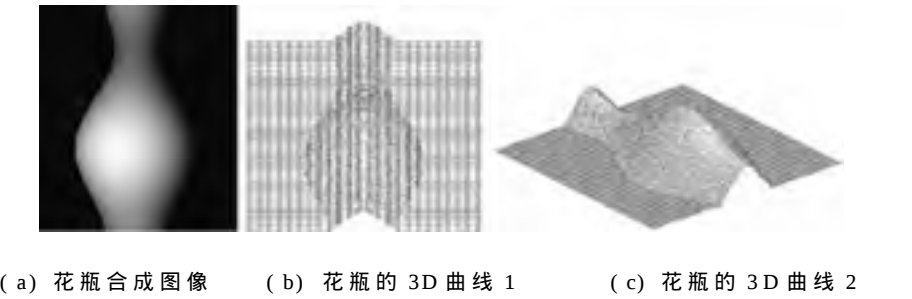


图 4 合成花瓶的试验结果

图 4(a) 参数估计结果为 $\tau=-1.475\ 3$, $\gamma=1.504\ 4$, $\eta=1.119\ 3$; 图 4(b) 和(c) 为利用 SFS 算法得到的、不同视角的花瓶高度 3D 曲线图。

表 1 本算法与传统算法的性能比较^[1, 7]

SFS 算法	时间 / s		高度误差		
	参数估计	重构	平均	最大	标准偏差
完整算法	35.5	67.8	2.85	19.19	4.49
传统算法 1	—	188.4	10.0	37.6	13.2
传统算法 2	—	0.9	8.3	25.6	15.0

(上接第 194 页) 在傅里叶谱图像中心, 设置中心及圆外的频率成分为 0, 全局周期性、重复性的纹理结构可被有效地清除掉, 而局部异常在恢复图像中被明显地保留下来了。用于图像恢复的圆的最优半径由不同半径的平均功率曲线的最大曲率确定。值的微小变化不会影响重构结果。本文提出的方法是无监督的, 它无须预先存储无缺陷图像或参考纹理的任何信息。它使该方法对照明等的环境变化不敏感。实验结果表明提出的方法非常适于检测包含同质结构的统计纹理中的缺陷。提出的方法的性能随着纹理结构改变的不规则而逐渐恶化。

本研究中提出的方法主要集中于纹理图像中的缺陷检测, 而不是缺陷类型的分类。尽管由提出的方法得到的分割可能会模糊或者在形状上不精确, 它仍能可靠检测纹理图像中缺陷的存在。在分类应用中, 提出的方法可用于复杂纹理中缺陷的快速定位, 然后在检测到的缺陷领域应用局部技术提取缺陷的颜色、块等特征。

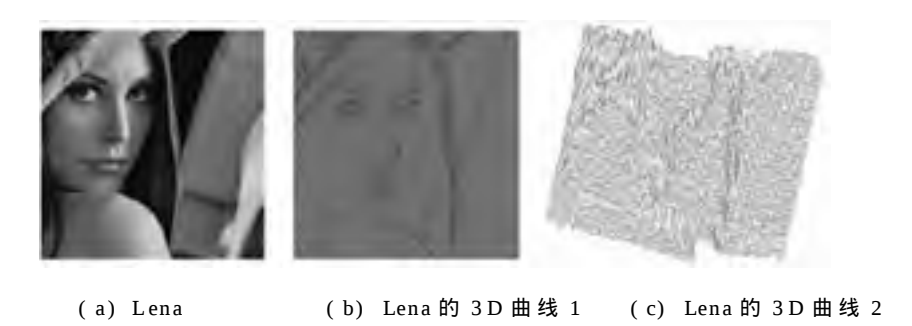


图 5 实际图像 Lena 的试验结果

图 5(a) 参数估计结果为 $\tau=1.075\ 3$, $\gamma=0.897\ 17$, $\eta=0.707\ 8$; 图 5(b) 和(c) 为利用 SFS 算法得到的、不同视角的 Lena 的高度 3D 曲线图。

5 结束语

一个具有实际应用价值的 SFS 算法是必须包括控制参数的估计和算法的实现这两个部分的。本文的试验结果显示: 此控制参数估计方法和三维重构算法能有效结合, 重构出比较逼真的三维效果图, 有一定的应用前景。

参考文献:

[1] HANG R, TSAI P S, CRYER J E, *et al.* Shape from shading: a survey[J]. IEEE Trans Pattern Analysis and Machine Intelligence, 1999, 21(8): 690-706.

[2] 廖熠, 赵荣椿. 从明暗恢复形状(SFS)的几类典型算法分析与评价[J]. 中国图象图形学报, 2001, 6(A)(10): 953-961.

[3] 杨敬安. 基于照明参数与反射系数的分层 SFS 算法[J]. 自动化学报, 1999, 25(6): 735-742.

[4] ZHENG Q, CHELLAPPA R. Estimation of illuminant direction, albedo, and shape from shading[J]. IEEE Trans Pattern Analysis and Machine, 1991, 13(7): 680-702.

[5] 高月芳, 罗飞, 曹建忠. 三维表面重构过程参数的估计研究[J]. 计算机工程与应用, 2005, 41(24): 31-32.

[6] ZHENG Q, CHELLAPPA R. Estimation of lambertian reflectance map[C] //Conference Record the 24th Asilomar Conference on Signals, Systems and Computers. 1990: 805-809.

[7] LEE K M, KUO C C. Shape from shading with a linear triangular element surface model[J]. IEEE Trans Pattern Analysis and Machine Intelligence, 1993, 15(8): 815-822.

致谢: 感谢毕业于中国科学院的姜成山的指导与帮助。

参考文献:

[1] IVARINEN J. Surface defect detection with histogram-based texture features[C] //Proc of SPIE 4197, 2000: 140-145.

[2] CHAN C H, PANG K H. Fabric defect detection by Fourier analysis [J]. IEEE Trans Industry Applications, 2000, 36: 1267-1276.

[3] WILTSCHI K, PINZ A, LINDBERG T. Automatic assessment scheme for steel quality inspection[J]. Machine Vision and Applications, 2000(12): 113-128.

[4] 段峰, 王耀南, 雷晓峰, 等. 机器视觉技术及其应用综述[J]. 自动化博览, 2002(3): 59-62.

[5] 谢勇, 王耀南, 彭涛, 等. 基于机器视觉印品缺陷检测的滤波算法[J]. 湖南大学学报: 自然科学版, 2005, 32(4): 53-57.

[6] 姜涛, 王安麟, 王石刚, 等. 基于机器视觉的印刷电路板误差校正方法[J]. 上海交通大学学报, 2005, 39(6): 945-949.