

一种改进的光流算法

杨国亮, 王志良, 牟世堂, 解 仑, 刘冀伟

(北京科技大学信息工程学院, 北京 100083)

摘要: 光流法是运动图像序列分析的一种重要方法。该文通过引入前向-后向光流方程, 计算其 Hessian 矩阵, 把 Hessian 矩阵条件数的倒数作为 Lucas-Kanade 光流法的加权阵, 可有效消除局部邻域中不可靠约束点, 同时提高基本约束方程解的稳定性。实验表明该方法相对于其它梯度约束光流法具有更好的可靠性。

关键词: 光流; Hessian 矩阵; 条件数

An Improved Optical Flow Algorithm

YANG Guoliang, WANG Zhiliang, MU Shitang, XIE Lun, LIU Jiwei

(School of Information Engineering, Beijing University of Science & Technology, Beijing 100083)

【Abstract】 Optical flow estimation is an important method to motion image analysis. This paper introduces forward and backward constraint equation and Hessian matrix for the computation of optical flow. It examines well-posedness of each point of local neighbourhood and the weight of Lucas-Kanade's method is defined as the reciprocal of the conditioning number of its Hessian Matrix. This can eliminate those uncertainty constrains and improve the numerical stability of the solution of the gradient constraint equation. Experimental results show that this method is suitable and reliable.

【Key words】 Optical flow; Hessian matrix; Conditioning number

运动目标检测是图像处理技术的一个重要组成部分, 它是计算机视觉、模式识别、图像编码和安全监控等研究领域的重点和难点。现有的运动目标检测方法主要有运动能量法、背景减法、图像差分法和光流法。运动能量法适合于复杂变化场合, 能够粗略地分割出运动目标; 背景减法在背景简单的情况下比较有效; 而图像差分法则直接比较两帧图像对应像素点的灰度值提取运动信息, 计算比较简单。光流法作为一种重要的运动图像分析方法, 在最近 20 年得到了较大发展, 研究者提出了多种不同的改进算法, 主要可分为 4 类^[1]: 时空梯度法, 块匹配方法, 基于能量方法和基于相位分析法, 其中时空梯度法最为常见。

1 Lucas-Kanade 光流法^[3]

光流场的计算最初是由 Horn 和 Schunck 提出^[2]。假定 t 时刻图像上的点 (x, y) 处灰度值为 $I(x, y, t)$, 在时刻 $t + \Delta t$ 时, 这一点运动到 $(x + \Delta x, y + \Delta y, t + \Delta t)$, 对应的灰度值为 $I(x + \Delta x, y + \Delta y, t + \Delta t)$, 假定它与 $I(x, y, t)$ 相等, 即

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) \quad (1)$$

将左边在 (x, y, t) 点用泰勒公式展开, 忽略二阶和二阶以上项可以得到

$$\frac{\partial I}{\partial x} \Delta x + \frac{\partial I}{\partial y} \Delta y + \frac{\partial I}{\partial t} \Delta t = 0$$

记 $u(x, y, t) = \frac{dx}{dt} = \frac{\Delta x}{\Delta t}$; $v(x, y, t) = \frac{dy}{dt} = \frac{\Delta y}{\Delta t}$, 则可得到基本光流约束方程:

$$I_x u + I_y v + I_t = 0 \quad (2)$$

式中 $I_x = \frac{\partial I}{\partial x}$, $I_y = \frac{\partial I}{\partial y}$, $I_t = \frac{\partial I}{\partial t}$, 写成梯度形式为 $(\nabla I)^T U + I_t = 0$ 。

由于光流场 $U = (u, v)^T$ 有 2 个变量, 而基本约束方程只有一个, 只能求出光流场沿梯度方向的值, 因此从基本光流方

程求解光流场是一个不适定问题, 必须引入附加的约束条件。Lucas 和 Kanade 假设在一个小的空间邻域上运动矢量保持恒定, 然后使用加权最小二乘法 (weighed least-squares) 估计光流。在一个小的空间邻域上, 光流估计误差定义为

$$\sum_{(x, y) \in \Omega} W^2(x) (I_x u + I_y v + I_t)^2 \quad (3)$$

其中 $W^2(x)$ 表示窗口权重函数, 它使邻域中心区域对约束产生的影响比外围区域更大, 式(3)解为

$$U = (A^T W^2 A)^{-1} A^T W^2 B \quad (4)$$

其中, 在 t 时刻的 n 个点 $X_i \in \Omega$, $A = [\nabla I(X_1), \dots, \nabla I(X_n)]^T$, $W = \text{diag}[W(X_1), \dots, W(X_n)]$, $B = -[I_t(X_1), \dots, I_t(X_n)]^T$

2 改进的光流计算方法

传统的光流计算方法主要是基于灰度守恒和光流场的平滑性假设, 但这些假设在阴影、边界和遮挡性的地方不再成立, 为此, 本文提出相应的改进算法。

2.1 前向-后向光流方程

考虑方程

$$I(x, y, t) - I(x + \Delta x, y + \Delta y, t + \Delta t) = 0$$

可以得到

$$I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t) \quad (5)$$

$$I(x, y, t + \Delta t) = I(x - \Delta x, y - \Delta y, t) \quad (6)$$

基金项目: 北京市现代信息科学与网络技术重点实验室基金资助项目(TDXX0503); 北京科技大学重点基金资助项目(20040503990); 江西省教育厅科技计划基金资助项目(赣教技字[2005]145)

作者简介: 杨国亮(1973—), 男, 博士生, 主研方向: 图像处理, 面部表情识别, 情感计算; 王志良, 博士、教授、博导; 牟世堂, 高工; 解 仑, 博士、副教授; 刘冀伟, 硕士、副教授

收稿日期: 2005-11-09 **E-mail:** ygliang30@126.com

对式(5)、式(6)分别进行泰勒展开并忽略二阶及二阶以上项：

$$\frac{\partial I(x, y, t)}{\partial x} \Delta x + \frac{\partial I(x, y, t)}{\partial y} \Delta y + \frac{\partial I(x, y, t)}{\partial t} \Delta t = 0$$

$$\frac{\partial I(x, y, t + \Delta t)}{\partial x} \Delta x + \frac{\partial I(x, y, t + \Delta t)}{\partial y} \Delta y + \frac{\partial I(x, y, t + \Delta t)}{\partial t} \Delta t = 0$$

即

$$I'_x u + I'_y v + I'_t = 0 \quad (7)$$

$$I_x^{t+\Delta t} u + I_y^{t+\Delta t} v + I_t^{t+\Delta t} = 0 \quad (8)$$

其中： $I'_x = \frac{\partial I(x, y, t)}{\partial x}$ ， $I'_y = \frac{\partial I(x, y, t)}{\partial y}$ ， $I'_t = \frac{\partial I(x, y, t)}{\partial t}$

$$I_x^{t+\Delta t} = \frac{\partial I(x, y, t + \Delta t)}{\partial x}$$
， $I_y^{t+\Delta t} = \frac{\partial I(x, y, t + \Delta t)}{\partial y}$ ， $I_t^{t+\Delta t} = \frac{\partial I(x, y, t + \Delta t)}{\partial t}$

式(7)、式(8)即为前向 - 后向光流方程，可以合并为一个新的光流方程：

$$I'_x u + I'_y v + I'_t = 0 \quad (9)$$

式中： $I'_x = f_1(I'_x, I_x^{t+\Delta t})$ ， $I'_y = f_2(I'_y, I_y^{t+\Delta t})$ ， $I'_t = f_3(I'_t, I_t^{t+\Delta t})$

文中取 $I'_x = \alpha I'_x + (1-\alpha) I_x^{t+\Delta t}$ ， $I'_y = \alpha I'_y + (1-\alpha) I_y^{t+\Delta t}$ ，

$$I'_t = \alpha I'_t + (1-\alpha) I_t^{t+\Delta t}$$

α 为一常数，根据性能指标式(3)，可求解出光流场 (u, v) 。

2.2 Hessian 矩阵

尽管 Lucas - Kanade 光流法计算简单，光流估计精度较高，但它有一个致命缺点，假定邻域内各像素点光流保持恒定，而且光流计算依赖于窗口权重函数，这意味着如果在邻域内存在严重违反光流约束方程的点或邻域运动不连续，将使得估计的光流可靠性严重降低。为此，本文引入 Hessian 矩阵判断领域内每点对于基本约束方程的“良态性”。

方程(9)分别对 x, y 求偏导可得：

$$I_{xx} u + I_{yx} v = -I_{tx}$$

$$I_{xy} u + I_{yy} v = -I_{ty} \quad (10)$$

即

$$\begin{bmatrix} I_{xx} & I_{yx} \\ I_{xy} & I_{yy} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_{tx} \\ I_{ty} \end{bmatrix} \quad (11)$$

定义 Hessian 矩阵：

$$H = \begin{bmatrix} I_{xx} & I_{yx} \\ I_{xy} & I_{yy} \end{bmatrix} \quad (12)$$

Hessian 矩阵的条件数： $Cond(H) = \|H\| \cdot \|H^{-1}\| = \frac{|\lambda_{max}|}{|\lambda_{min}|}$ ，其

中 λ_{max} 、 λ_{min} 分别为 Hessian 矩阵 H 的最大特征值和最小特征值，可以通过 Hessian 矩阵的条件数大小来判断方程(11)解的稳定性，如果 Hessian 矩阵的条件数很大则方程(11)为病态方程，对应的 Hessian 矩阵秩很小，其解不稳定，计算的光流不可靠；如果 Hessian 矩阵的条件数接近 1，对应的 Hessian 矩阵秩很大，方程(11)为良态，其解鲁棒性较好。由此可以通过计算 Hessian 矩阵的条件数来剔除邻域内不可靠点。

2.3 梯度阈值处理

在复杂场景运动目标检测中，由于场景和运动目标的灰度梯度可能相差不大，使得运动目标轮廓处于模糊状态，计算的轮廓处光流不准确，也即基本光流约束方程在灰度梯度很小时不成立^[4]。为此引入梯度约束条件，设定灰度梯度阈值 T，在灰度梯度大于阈值 T 的像素点计算光流。

2.4 基于 Hessian 矩阵的光流算法

Hessian 矩阵的条件数很好地刻画了线性方程(11)解的稳定性，而且条件数越大，对应的 Hessian 矩阵的秩越小，为此可以先利用 Hessian 矩阵剔除邻域内不可靠点，并把各点对应条件数的倒数作为该点权重，其算法如下：

(1) 计算图像中每点的一阶和二阶梯度；

(2) 分别计算每点对应 Hessian 矩阵的秩 $\det(H)$ 和条件数

$Cond(H)$ ，设定阈值 τ ，则：

$$W(X) = \begin{cases} 0 & f \det(H) < \tau \\ 1/Cond(H) & \text{if } \det(H) \geq \tau \end{cases}$$

并对每个邻域内的 $W(X)$ 进行归一化处理；

(3) 采用加权最小二乘法求解式(9)光流场 (u, v) 。

3 实验结果及分析

为了验证上述算法的有效性，本文采用了 1 组合成图像序列和 2 组真实图像序列进行实验。其中合成图像序列相邻帧间的真实光流已知，通过采用本文算法对其估计光流，然后与真实光流比较，作出定量评估。对于光流场 $V = (u, v)^T$ ，

把它写出一个三维的方向向量 $\vec{V} = \frac{1}{\sqrt{u^2 + v^2 + 1}}(u, v, 1)^T$ ，真实光

流 \vec{V}_c 和估计光流 \vec{V}_e 之间的光流角误差可以表示为 $\psi_E = \arccos(\vec{V}_c \cdot \vec{V}_e)$ 。

$$\text{平均误差为 } AE = \frac{1}{N} \sum_{i=1}^N \psi_E(i)$$

式中，N 为光流场的像素个数。光流场标准角偏差定义为

$$SD = \sqrt{\frac{1}{N} \sum_{i=1}^N (\psi_E(i) - AE)^2}$$

参照 Barron 等人^[1]的做法，本文亦在估计光流之前，采用了标准差为 1.5 像素/帧的时空高斯滤波器平滑图像序列，这有助于削弱时间噪声和输入中的量化效应。

图 1(a)、图 1(b)分别是 Translating Trees 图像序列的第 7、8 帧，该图像序列以 1.73~2.26 的流速向右运动，运动方向平行于水平轴。图 1(c)是利用本文方法计算的第 8、9 帧之间的光流场，可以看出光流方向基本跟真实光流一致，大小稍有差别，表 1 给出了本文和其它方法计算的光流平均角误差和标准角偏差的对照。

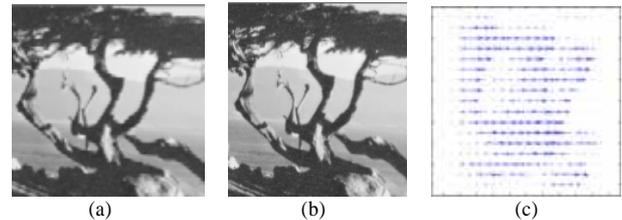


图 1 Translating Trees 序列与光流场

表 1 对 Translating Trees 序列，本文算法与其它算法比较

算法	流速平均角误差	流速标准角偏差	密度
Horn and Schunck(original) ^[1]	38.72°	27.67°	100%
Horn and Schunck(modified) ^[1]	2.02°	2.27°	100%
Lucas and Kanade($\sigma > 1.0$) ^[1]	0.66°	0.67°	100%
本文方法	0.61°	0.62°	100%

图 2 是 Rubic Cube 序列实验结果，其中图 2(a)、图 2(b)为 Rubic Cube 序列中的第 9、10 帧，图 2(c)为采用本文方法得到的第 9、10 帧之间光流场，显然该光流场比较准确地反映了 Rubic Cube 图像序列的运动信息。（下转第 226 页）