

基于广义运动模糊模型的前向运动模糊核^{*}

蒋欣兰^{1,2}, 王亮¹, 罗晓月³, 王胜春¹, 罗四维¹

¹(北京交通大学 计算机与信息技术学院, 北京 100044)

²(中国青年政治学院 计算机教学与应用中心, 北京 100089)

³(Department of Mathematics, Linfield College, OR 97128, USA)

通讯作者: 蒋欣兰, E-mail: jxlzgl@sina.com



摘要: 从光流场的角度出发, 建立了一种广义运动模糊模型, 并依据该模型推导出前向运动模糊核, 为高速铁路前向运动视频图像去模糊奠定了理论基础. 给出了理论分析后, 设计了一种快速生成前向运动模糊核的方法, 在这个过程中, 解决了 3 个具体问题: 快速的运动估计方法的解析解、平面场景朝向的快速估计方法的解析解、前向运动模糊核的数值生成方法. 实验结果验证了该算法的正确性.

关键词: 图像模糊; 非一致卷积核; 光流; 无源导航; 模糊核
中图法分类号: TP391

中文引用格式: 蒋欣兰, 王亮, 罗晓月, 王胜春, 罗四维. 基于广义运动模糊模型的前向运动模糊核. 软件学报, 2016, 27(8): 2135-2146. <http://www.jos.org.cn/1000-9825/4864.htm>

英文引用格式: Jiang XL, Wang L, Luo XY, Wang SC, Luo SW. Forward-Motion blurring kernel based on generalized motion blurring model. Ruan Jian Xue Bao/Journal of Software, 2016, 27(8): 2135-2146 (in Chinese). <http://www.jos.org.cn/1000-9825/4864.htm>

Forward-Motion Blurring Kernel Based on Generalized Motion Blurring Model

JIANG Xin-Lan^{1,2}, WANG Liang¹, LUO Xiao-Yue³, WANG Sheng-Chun¹, LUO Si-Wei¹

¹(School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China)

²(Computer Science Teaching and Application Center, China Youth University of Political Studies, Beijing 100089, China)

³(Department of Mathematics, Linfield College, OR 97128, USA)

Abstract: In this paper, a generalized motion blurring model is constructed from the viewpoint of optical flow. Then based on the model, forward motion blurring kernel is deduced. The kernel provides a theoretical foundation for forward motion deblurring of high speed railway from image sequences. A fast method is also designed to estimate forward motion blurring kernel on this theory. Three specific problems are solved in this process. First, the analytical solution under quick motion estimation method is obtained. Next, the analytical solution under quick motion estimation method of planar scene direction is achieved. Lastly, the numerical calculation algorithm of forward motion blurring kernel is developed. Experimental results validate the proposed method.

Key words: image blurring; non-uniformed convolutional kernel; optical flow; passive navigation; blurring kernel

在使用相机进行拍摄的过程中, 由于相机相对于被拍摄对象间存在相对运动, 经常会造成图像模糊. 在实际应用中, 相机的运动形式多种多样, 不同的运动形式产生的图像运动模糊机理也不尽相同, 尽管在视觉上对模糊的感觉是相同的. 例如, 传统的(基于图像平移运动的)卷积模型与以前向运动为主的相机运动模型之间存在很

* 基金项目: 国家自然科学基金(61272354, 61273364, 61300176, 61473031, 61472029); 北京市自然科学基金(4152042); 中央高校基本科研业务费专项资金(2013JBM019)

Foundation item: National Natural Science Foundation of China (61272354, 61273364, 61300176, 61473031, 61472029); Beijing Natural Science Foundation of China (4152042); Fundamental Research Funds for the Central Universities of China (2013JBM019)

收稿时间: 2015-03-18; 修改时间: 2015-05-14; 采用时间: 2015-06-01

大的差别:前者相当于图像中各个像素点以相同的方向、速度进行运动,表现出一致性;而后者相当于图像中各个像素点以不同的方向、速度进行运动,表现出“非一致”性,致使各个像素点的模糊是“非一致”的。

去除图像模糊是一个典型的逆问题,其相应的正问题为图像的模糊过程。对正问题的正确理解,是求解逆问题的关键之一。

为得到有效去除运动模糊的算法,需要在分析其基本原理的基础上建立正确的图像运动模糊模型。

本文焦点放在一个具有典型代表性的运动模糊情况——相机以前向平移方式运动的情况,即,前向运动。该运动导致图像中所有像素点有各自不同的移动方向和速度。对于这种非一致性的图像模糊,称为前向运动模糊。其他运动模糊,如平移、抖动等常见的运动模糊均属一致性运动模糊,是非一致性运动模糊的简化情况。

对于图像的前向运动模糊,目前的方法都不能解决这一问题。我们的研究也表明:本文给出的模型并非对以前研究的简单扩展,而是针对前向运动模糊成像的一种恢复方法。实际上,在计算机视觉、近景摄影测量等领域,也常遇到这种成像环境。

1 相关工作

实际得到的图像可以看作是“理想”图像经过某种退化过程所得到的结果,称为退化图像。图像复原就是要通过退化图像估计出理想图像^[1]。长久以来,不管是静态模糊复原模型(例如 TV(total variation)模型^[2]),还是一些经典的去除运动模糊的模型(例如 Levin^[3]和 Cho 等人^[4]的工作),图像的模糊过程都被描述为一个卷积过程。这样做的原因是模型比较简单,易于分析。事实上,模型本身就是实际情况经过理想化假设而得到的近似结果。对于静态模糊(例如图像失焦、Gauss 模糊等),使用卷积模型所表现出的缺陷并不十分明显,但是对于运动模糊,卷积模型表现出较大的局限性。这是由于相机的运动(例如抖动、前向运动等)所引起的图像模糊并不是一个卷积过程,图像上不同位置的像素点的模糊过程和方式不同。因此在前向运动成像条件下,要取得好的图像复原效果,需要了解图像运动模糊的正过程。

对非卷积模型,也称为非均匀模糊,已经有很多研究成果。如天文图像,由于大气湍流等因素的影响,即使是对于静态模糊,光斑的大小和形状也会随着像素点位置的不同而发生很大的变化。Fusco 等人^[5]对天文图像中的不一致模糊进行了深入研究。Whyte 等人^[6]指出,相机抖动所引起的图像模糊是非均匀的(他们用相机的转动模型来描述相机的抖动)。围绕着图像的非均匀模糊这个热点研究问题,提出了许多不同的方法。例如,Portz 等人^[7]提出了一种有效的光流估计方法,Gupta 等人^[8]提出了“运动密度函数”,并尝试用它来实现图像去模糊问题。Hirsch 等人^[9]提出了一种快速算法,用于去除相机抖动所引起的非均匀模糊。特别值得提出的是,Zheng 等人^[10]针对相机前向运动所引起的运动模糊进行了研究,给出了解决方法。然而,Zheng 处理的目标物体是在一个平面上,因此可以忽略景深的影响,相对于需要考虑景深的情况,在难度上大为减小。

上面提到的这些方法尽管关注重点不同,但其基本思路依然使用卷积模型来估计模糊核,只不过是先将图像划分为一系列局部小区域,对不同的区域使用不同的模糊核。因此,这些方法对模糊核的估计速度较慢。有的方法,如 Zheng 等人^[10]的工作,还需要人来参与。

本文从泛函运动的角度给出广义运动模糊模型。在此基础上,从光流场的角度出发,建立了一种前向运动模糊模型。基本出发点是,对于前向运动运动模糊,模糊核所描述的是像素点的运动轨迹,而光流所描述的是像素点的运动速度^[11,12]。如果曝光时间很短,可以近似认为像素点的运动轨迹等于运动速度乘以曝光时间。因此,可以通过估计光流来计算前向运动模糊核。

光流估计是机器视觉中的一个经典问题,已经有很多成熟算法^[5,11-13]。为提高光流估计的速度,可以先估计出相机的运动速度和景深(即无源导航^[4,15]),然后计算出光流,并最终生成前向运动模糊核。这一过程就是本文提出的“三步法”:无源导航→计算光流→生成前向运动模糊核。

2 广义运动模糊和广义运动模糊核

相机的运动会造成像平面上图像亮度模式的变化,而图像中像素点的灰度值取决于像平面上相应位置的

图像感知器对曝光时间 Δt 内的光子数的积累量的测量结果^[11].因此,这个问题可以分解为两个子问题:

- 1) 图像亮度模式的变化如何造成图像的模糊?
- 2) 相机的前向运动将使得图像亮度模式如何发生变化?

本节主要解决第 1 个问题,下一节解决第 2 个问题,从而建立起“广义运动”所对应的“前向运动”模糊模型.

2.1 广义运动模糊

图像是场景中物体在像平面上的二维投影,图像中的某一个像素点对应于场景中的某一个物体“小块”.由于场景和相机之间的相对运动,在不同时刻,这个“小块”将被投影到像平面上的不同位置,在相机快门曝光的瞬间 $\Delta t=T_0-0$ 内,这个“小块”将被投影到不同细分像平面上的不同位置(所谓细分像平面,是假设的一种没有模糊的理想像平面),从而引起图像亮度模式的变化.也就是说,细分像平面上某一固定位置的亮度将随着时间而发生变化,其中, T_0 为曝光时间.

由于相机的运动,场景中某一个“小块”的像在像平面内形成一条曲线 $l(x_1, y_1, t) = \{x(t), y(t)\}^T$,如图 1 所示,原来的 $x-y$ 平面扩展为 $x-y-t$ 空间.其中, $(x(0), y(0))^T = (x_1, y_1)^T$ 表示曲线的起始点位置.为确定亮度的变化,需要确定场景中某一个“小块”的像在细分像平面上如何运动,即,确定“小块”的像的运动轨迹函数 $l(x_1, y_1, t)$;同时,也要知道场景中某一个“小块”的像在细分像平面上运动过程中其亮度值如何变化?也就是要知道函数 $I(l(x_1, y_1, t))$ 的具体形式.

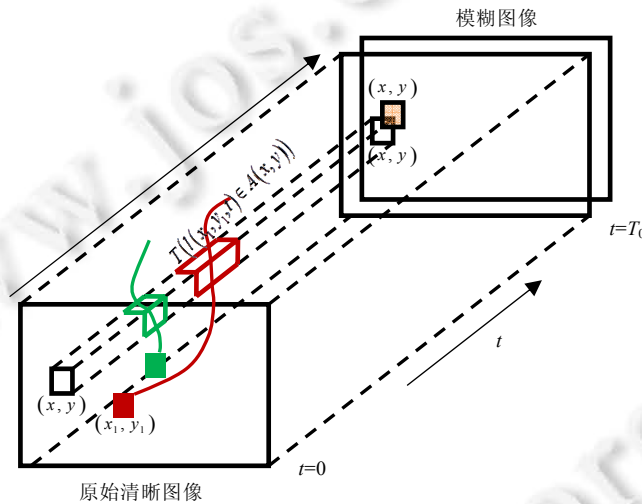


Fig.1 Generation model of motion blurring
图 1 运动模糊的生成模型

2.2 广义运动模糊核

运动模糊图像 $I_b(x, y)$ 是图像序列 $I(x, y, t)$ 的线性叠加结果(如图 1 所示),即

$$I_b(x, y) = \frac{1}{T_0} \int_0^{T_0} I(x, y, t) dt \tag{1}$$

图像序列 $I(x, y, t)$ 是由图像序列中的第一张图像 $I_0(x_1, y_1)$ (通过像素点的运动而)生成的,因此,模糊图像也可以看作是 $I_0(x_1, y_1)$ 中各个像素点的灰度值的线性叠加结果.于是,公式(1)可以被进一步写为

$$I_b(x, y) = \iint w(x, y, x_1, y_1) I_0(x_1, y_1) dx_1 dy_1 \tag{2}$$

本文中,公式(2)被称为广义运动模糊模型,核函数 $w(x, y, x_1, y_1)$ 的具体形式为(证明参见附录)

$$w(x, y, x_1, y_1) = \frac{1}{T_0} \frac{T(l(x_1, y_1, t) \in A(x, y))}{\delta x \delta y} \tag{3}$$

其中, $A(x,y)$ 表示以 $(x,y)^T$ 为中心的像素点区域; 而 δx 和 δy 分别表示该像素区域的长和宽; $T(l(x_1,y_1,t) \in A(x,y))$ 表示曲线 $l(x_1,y_1,t)$ 在像素区域 $A(x,y)$ 中的停留时间.

广义运动模糊模型代表了运动模糊的普遍模式, 可用以描述所有运动模糊. 例如, 当所有的曲线 $l(x_1,y_1,t)$ 具有完全相同的形状时, 如果将曲线 $l(x_1,y_1,t)$ 的起始点位置 $(x_1,y_1)^T$ 和像素区域 $A(x,y)$ 的中心位置 $(x,y)^T$ 同时移动 $(x_0,y_0)^T$, 曲线在像素区域中的停留时间并不发生变化. 不失一般性, 令 $(x_1,y_1)^T = (x_0,y_0)^T$, 于是, 广义模糊核函数为

$$w(x,y,x_1,y_1) = w(x-x_1,y-y_1,0,0) = k(x-x_1,y-y_1) \tag{4}$$

而广义运动模糊模型公式(2)就变为卷积模型为

$$I_b(x,y) = \iint k(x-x_1,y-y_1)I_0(x_1,y_1)dx_1dy_1 \tag{5}$$

对于绝大多数的运动模糊情况, 曲线 $l(x_1,y_1,t)$ 的形状并不相同, 因此, 卷积模型公式(5)不成立. 本文将依据广义运动模糊模型得到前向运动模糊模型.

3 无源导航算法

公式(3)中的曲线 $l(x_1,y_1,t)$ 的具体表达式为

$$l(x_1,y_1,t) : \begin{cases} x(t) = x_1 + \int_0^t u(x,y,t)dt \\ y(t) = y_1 + \int_0^t v(x,y,t)dt \end{cases} \quad (0 \leq t \leq T_0) \tag{6}$$

其中, 像素点的速度 $(u,v)^T$ 又被称为光流^[11,12]. 假设曝光时间 T_0 非常短, 于是, 公式(6)可以近似为

$$l(x_1,y_1,t) : \begin{cases} x(t) = x_1 + u(x,y) \cdot t \\ y(t) = y_1 + v(x,y) \cdot t \end{cases} \quad (0 \leq t \leq T_0) \tag{7}$$

因此, 计算曲线 $l(x_1,y_1,t)$ 等价于估计光流 $(u,v)^T$. 用 $(U,V,W)^T$ 表示相机沿着三维场景空间 X 轴、 Y 轴和 Z 轴的运动速度, 根据透视投影模型^[11], 场景中的一点 $(X,Y,Z)^T$ 被投影成像平面上的一点 $(x,y)^T$, 即

$$x = \frac{f}{Z}X; y = \frac{f}{Z}Y \tag{8}$$

而光流 $(u,v)^T$ 和相机速度 $(U,V,W)^T$ 之间的关系满足^[14]:

$$u = \frac{-Uf + xW}{Z}; v = \frac{-Vf + yW}{Z} \tag{9}$$

若 $W \neq 0$, 那么 $(u,v)^T$ 会随着 $(x,y)^T$ 的不同而发生变化(如图2所示), 因此, 对于图2的模糊并不是一个卷积过程.

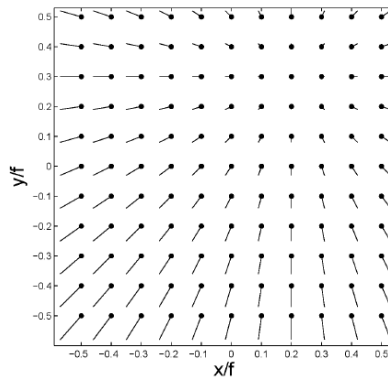


Fig.2 Optical flow field $(u,v)^T$ with $U=0.2W$ and $V=0.3W$ (Z is a constant)

图2 当 Z 为常数, $U=0.2W$ 和 $V=0.3W$ 时的光流场 $(u,v)^T$

直接估计 $(u,v)^T$ 很复杂, 并且精度不高^[12]. 对于相机前向运动的情况, 可以先估计 $(U,V,W)^T$ 和 $Z(x,y)$; 然后, 通过公式(9)计算光流场 $(u,v)^T$. 在机器视觉领域, 估计参数 $(U,V,W)^T$ 和 $Z(x,y)$ 也被称为无源导航^[14,15].

3.1 估计相机运动速度

在亮度不变的假设条件下,光流约束方程为^[12]

$$E_x u + E_y v + E_t = 0 \quad (10)$$

其中, $(E_x, E_y)^T$ 表示图像的亮度梯度,而 E_t 表示像素点亮度的时间变化率.将公式(9)带入公式(10)得到:

$$(-Uf+xW)E_x + (-Vf+yW)E_y + ZE_t = 0 \quad (11)$$

公式(11)中同时包含相机的运动速度 $(U, V, W)^T$ 和场景的深度 $Z(x, y)$, 直接求解较为困难,而无源导航^[15]是求解该问题的一种快速算法,其基本过程为:

- 1) 选择点集 $\Omega = \{(x, y)^T | (E_x, E_y)^T \neq (0, 0)^T \text{ 并且 } E_t = 0\}$.
- 2) 根据点集 Ω , 使用最小二乘法^[16]求出 U 和 V (令 $W=1$).
- 3) 选择点集 $\Phi = \{(x, y)^T | (E_x, E_y)^T \neq (0, 0)^T \text{ 并且 } E_t \neq 0\}$.
- 4) 根据点集 Φ 和步骤 2) 中求出的 U 和 V , 使用最小二乘法^[16]求出 $Z(x, y)$ (令 $W=1$).

我们将详细讨论步骤 2) 和步骤 4) 的具体实现过程.首先,对于点集 Ω , 公式(11)进一步写为

$$(-Uf+x)E_x + (-Vf+y)E_y = 0 \quad (12)$$

最小二乘拟合通过最小化如下目标函数:

$$\Gamma_1(U, V) = \iint_{\Omega} ((-Uf+x)E_x + (-Vf+y)E_y)^2 dx dy \quad (13)$$

来求得相应的 U 和 V . 令公式(13)关于 U 和 V 的偏导数等于 0, 进一步得到:

$$\begin{cases} \left(\iint_{\Omega} E_x^2 dx dy \right) U + \left(\iint_{\Omega} E_x E_y dx dy \right) V = \iint_{\Omega} \frac{E_x}{f} (xE_x + yE_y) dx dy \\ \left(\iint_{\Omega} E_x E_y dx dy \right) U + \left(\iint_{\Omega} E_y^2 dx dy \right) V = \iint_{\Omega} \frac{E_y}{f} (xE_x + yE_y) dx dy \end{cases} \quad (14)$$

求解上式,即可得到关于 U 和 V 的解析解.

3.2 估计场景的深度

在前向运动视频的实际应用中,一个较为合理的假设是将场景视为平面,如路面、护栏、整齐排列的电杆都位于平面范围^[17],即 $Z = -n_1 X - n_2 Y + Z_0$. 根据公式(8)整理得到: $Z = Z_0 / (1 + n_1 X/f + n_2 Y/f)$, 于是,公式(11)整理为

$$g(x, y) f + g(x, y) x n_1 + g(x, y) y n_2 + f E_t Z_0 = 0 \quad (15)$$

其中, $g(x, y) = (-Uf+x)E_x + (-Vf+y)E_y$. 最小二乘拟合通过最小化如下目标函数:

$$\iint_{\Phi} \left(g(x, y) + g(x, y) \frac{x}{f} n_1 + g(x, y) \frac{y}{f} n_2 + E_t Z_0 \right)^2 dx dy \quad (16)$$

来求得相应的 n_1, n_2 和 Z_0 . 令公式(16)关于 n_1, n_2 和 Z_0 的偏导数等于 0, 得到:

$$\begin{cases} a n_1 + b n_2 + c Z_0 = h_1 \\ b n_1 + d n_2 + e Z_0 = h_2 \\ c n_1 + e n_2 + g Z_0 = h_3 \end{cases} \quad (17)$$

其中,公式(17)中的参数为

$$\begin{aligned} a &= \iint_{\Phi} g^2(x, y) x^2 dx dy, & b &= \iint_{\Phi} g^2(x, y) x y dx dy, \\ c &= f \iint_{\Phi} g(x, y) x E_t dx dy, & d &= \iint_{\Phi} g^2(x, y) y^2 dx dy, \\ e &= f \iint_{\Phi} g(x, y) y E_t dx dy, & g &= f^2 \iint_{\Phi} E_t^2 dx dy, \\ h_1 &= f \iint_{\Phi} g^2(x, y) x dx dy, & h_2 &= f \iint_{\Phi} g^2(x, y) y dx dy, \\ h_3 &= f^2 \iint_{\Phi} g(x, y) E_t dx dy. \end{aligned}$$

通过求解公式(16),可以得到关于 n_1, n_2 和 Z_0 的解析解,进而得到平面场景的表达 $Z(x, y)$. 将 U, V 以及 $Z(x, y)$ 带入公式(9) (令 $W=1$), 就得到了光流场 $(u, v)^T$ 的具体形式.

4 生成前向运动模糊核

光流 $(u,v)^T$ 是基于“前向运动”这个假设估计出来的,因此,我们将根据公式(7)计算出的模糊核称为前向运动模糊核,记为 $W=\{w_{i,j,k,l}\}$.具体计算包括两个子过程.

- 1) 确定出从像素点 $A_{k,l}$ 出发的线段(即公式(7))所“扫过”的像素点集合 $R_{k,l}$ (称为离散线段);
- 2) 计算 $R_{k,l}$ 所包含的各个像素点 (i,j) 的权重系数 $w_{i,j,k,l}$.

4.1 生成离散线段

首先,根据公式(9)计算出像素区域 $A_{k,l}$ 的中心点 $(x_{k,l},y_{k,l})^T$ 的运动速度 $(u_{k,l},v_{k,l})^T$.首先考虑 $(u_{k,l},v_{k,l})^T$ 的方向在 0° 和 45° 之间的情况(如图3所示).根据公式(7),可以得到相应的直线段方程:

$$l_{k,l}:(x-x_{k,l})v_{k,l}=(y-y_{k,l})u_{k,l} \quad (x_{k,l} \leq x \leq x_{k,l}+u_{k,l}T_0) \quad (18)$$

将该直线段分别向上和向下平移 $1/2$ 个像素点尺寸,可以得到两条新的直线段:

$$\begin{cases} l_{k,l}^+ : (x-x_{k,l})v_{k,l} = (y-y_{k,l}-1/2)u_{k,l} \\ l_{k,l}^- : (x-x_{k,l})v_{k,l} = (y-y_{k,l}+1/2)u_{k,l} \end{cases}, x_{k,l} \leq x \leq x_{k,l}+u_{k,l}T_0 \quad (19)$$

本文中,我们将这两条直线段之间的区域定义为像素点 $A_{k,l}$ 所“扫过”的区域(如图3所示).进一步,需要确定出哪些像素点包含在该区域之中,并且,将所有这些被检测出的像素点保存成一个数组 $R_{k,l}=\{(i,j)\}$.

对于 $(u_{k,l},v_{k,l})^T$ 的方向在 0° 和 45° 之间的情况,我们考虑像素区域 $A_{k,l}$ 的右上角点(如图3所示).找出 $R_{k,l}$ 中所包含的所有像素点的右上角点是一个迭代过程.如果 $R_{k,l}$ 中所包含的第 m 个右上角点的坐标为 $(x_{i,j}+1/2, y_{i,j}+1/2)^T$,那么第 $m+1$ 个右上角点只有两种可能,即 $(x_{i,j}+3/2, y_{i,j}+1/2)^T$ 或者 $(x_{i,j}+3/2, y_{i,j}+3/2)^T$.如果第 $m+1$ 个右上角点的坐标为 $(x_{i,j}+3/2, y_{i,j}+1/2)^T$,我们将其称为前行点(图4中的 B_5 到 B_6).如果第 $m+1$ 个右上角点的坐标为 $(x_{i,j}+3/2, y_{i,j}+3/2)^T$,我们将其称为上跳点(图4中的 A_3 到 A_4).

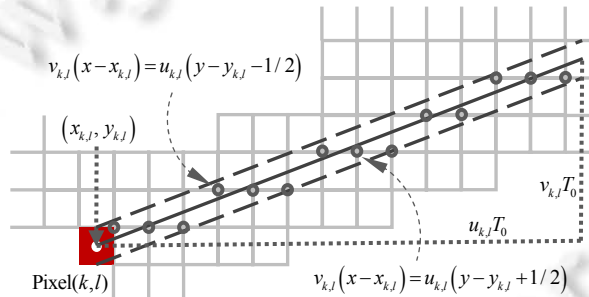


Fig.3 Determining the pixel set $R_{k,l}$ scanning by pixel region $A_{k,l}$

图3 确定像素点区域 $A_{k,l}$ 所“扫过”的像素点集合 $R_{k,l}$

确定第 $m+1$ 个右上角点到底是前行点还是上跳点的关键在于第 m 个右上角点到直线区域下边界的垂直距离 $d_{k,l}^{(m)}$ (即图4中的线段 A_2A_3 或 B_2B_3).令 $r_{k,l}=u_{k,l}/v_{k,l}$,可以得到如下的判断准则(如图4所示):

- 1) 第 $m+1$ 个右上角点为

$$\begin{cases} (x_{k,l}+3/2, y_{k,l}+1/2)^T, & \text{如果 } d_{k,l}^{(m)} - r_{k,l} > 0 \\ (x_{k,l}+3/2, y_{k,l}+3/2)^T, & \text{如果 } d_{k,l}^{(m)} - r_{k,l} \leq 0 \end{cases} \quad (20)$$

- 2) 第 $m+1$ 个右上角点到直线区域下边界的垂直距离为

$$\begin{cases} d_{k,l}^{(m+1)} = d_{k,l}^{(m)} - r_{k,l}, & \text{如果 } d_{k,l}^{(m)} - r_{k,l} > 0 \\ d_{k,l}^{(m+1)} = d_{k,l}^{(m)} - r_{k,l} + 1, & \text{如果 } d_{k,l}^{(m)} - r_{k,l} \leq 0 \end{cases} \quad (21)$$

事实上,公式(20)和公式(21)给出了一个交叉迭代过程,迭代的初始值为:1) 第1个右上角点 $(x_{i,j}+1/2,$

$y_{i,j+1/2}^T$ 以及 2) $d_{k,l}^{(0)} = 1 - 1/2r_{k,l}$. 于是就求出了 $R_{k,l}$ 中所包含的所有的右上角点(即图 4 中的“小圆圈”).

对于 $(u_{k,l}, v_{k,l})^T$ 的方向在其他角度范围内的情况,只需进行微小的调整即可.例如,如果 $(u_{k,l}, v_{k,l})^T$ 的方向在 $45^\circ \sim 90^\circ$ 之间,仍然考虑像素区域的右上角点,只是此时应该不断增加 y 值的大小,然后判断 x 的值是否需要加 1. 该方法最早由 Bresenham 提出^[18],是计算机图形学中经典的离散直线(或曲线)生成算法^[19].

4.2 计算相交面积

在找到所有的右上角点以后(仍然以 $(u_{k,l}, v_{k,l})^T$ 的方向在 0° 和 45° 之间的情况为例),需要进一步计算权重系数,即,相应的像素区域和 $A_{k,l}$ 所“扫过”的区域直线段区域之间的相交面积.仔细观察图 3 不难发现,对于右上角点,存在如下两种不同的情况(如图 4 所示).

- 1) 如果该右上角点是上跳点(例如图 4 中的 A_4),那么在该右上角点左侧,有 3 个像素点属于 $R_{k,l}$;
- 2) 如果该右上角点是前行点(例如图 4 中的 B_6),那么在该右上角点左侧,只有两个像素点属于 $R_{k,l}$.

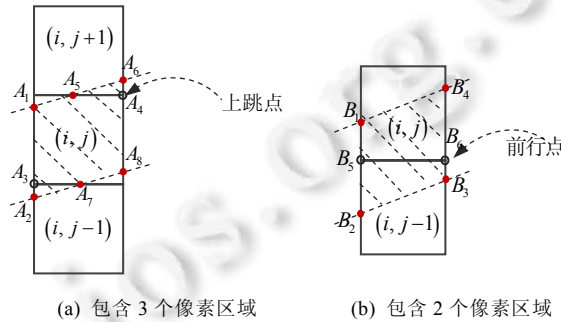


Fig.4 Three pixel points are located on the discrete line segment $R_{k,l}$ at the left side of up-jumpig point; while only two pixel points at the left side of front-jumping point

图 4 对于上跳点,在该上跳点的左侧有 3 个像素点属于离散线段 $R_{k,l}$;而对于前行点,在该前行点的左侧,只有两个像素点属于离散线段 $R_{k,l}$

在由公式(20)和公式(21)所构成的交叉迭代过程中,记录下每一个新得到的右上角点是上跳点还是前行点,于是就得到了 $R_{k,l}$ 中的所有像素点,然后计算 $R_{k,l}$ 所包含的每一个像素点 (i,j) 所对应的权重系数 $w_{i,j,k,l}$,即

$$w_{i,j,k,l} = S_{i,j} / \left(\sum_{(i,j) \in R_{k,l}} S_{i,j} \right), \text{ 对于 } (i,j) \in R_{k,l} \quad (22)$$

其中, $S_{i,j}$ 表示像素点区域 $A_{k,l}$ 与直线段区域(即,图 3 中两条虚线之间的区域)之间的相交面积. 一个重要的观察结果是,图 4 中的四边形 $A_1A_2A_3A_4$ 和 $B_1B_2B_3B_4$ 的面积都是 1 个像素区域.

对于上跳点的情况,三角形 $A_2A_3A_7$ 的面积为

$$S(A_2A_3A_7) = \frac{1}{2} A_2A_3 \times A_3A_7 = \frac{1}{2} (d_{k,l}^{(m)})^2 / r_{k,l} \quad (23)$$

而三角形 $A_5A_4A_6$ 的面积为

$$S(A_5A_4A_6) = \frac{1}{2} A_5A_4 \times A_4A_6 = \frac{1}{2} (1 - d_{k,l}^{(m+1)})^2 / r_{k,l} \quad (24)$$

进一步求得:

$$S(A_1A_3A_7A_8A_4A_5) = 1 - S(A_2A_3A_7) - S(A_5A_4A_6) \quad (25)$$

对于前行点的情况,计算过程更加简单. 首先计算:

$$S(B_5B_2B_3B_6) = \frac{1}{2} (B_2B_5 + B_3B_6) \times 1 = \frac{1}{2} (d_{k,l}^{(m)} + d_{k,l}^{(m+1)}) \quad (26)$$

然后,计算梯形 $B_1B_5B_6B_4$ 的面积:

$$S(B_1B_5B_6B_4) = 1 - S(B_5B_2B_3B_6) \quad (27)$$

5 数值求解

为了进一步验证上述理论分析结果与实际的吻合程度,本文以真实场景为对象设计了实验.高速铁路检测列车前端安装的摄像机获取的图像是典型的前向运动视频.列车的高速前向运动,使得图像中含有很强的运动模糊.如果在去模糊过程中采用前向运动模糊核所获得的去模糊效果优于卷积模糊核的效果,无疑可以进一步表明本文的研究思路以及所提出前向运动模糊核是正确的.为此,本文建立依据卷积模糊核与前向运动模糊核的图像复原模型,并对两者进行比较.

5.1 非线性热扩散方程

我们通过求解一个泛函优化模型^[20]:

$$\min_{I_0} \left\{ \frac{1}{2} \| I_b - \iint w(x, y, x_1, y_1) I_0(x_1, y_1) dx_1 dy_1 \|_2^2 + \lambda \| \nabla I_0 \|_1 \right\} \quad (28)$$

来实现图像复原,其中, $J_0(x, y)$ 的全变分的表达式为

$$\| \nabla I_0 \|_1 = \iint | \nabla I_0 | dx dy = \iint \sqrt{(\partial I_0 / \partial x)^2 + (\partial I_0 / \partial y)^2} dx dy \quad (29)$$

通过变分法^[20],可以得到求解公式(28)的演化方程:

$$\frac{\partial}{\partial t} \nabla I_0 = \text{div}(\nabla I_0 / | \nabla I_0 |) + \lambda \cdot s(x, y) \quad (30)$$

即,公式(28)所对应的变分梯度下降流^[20].其中, ∇ 表示梯度算子, div 表示散度算子^[16],而 $s(x, y)$ 的表达式为

$$s(x, y) = \iint w(x, y, x_1, y_1) (I_b(x_1, y_1) - \iint w(x, y, x_1, y_1) I_0(x, y) dx dy) dx dy \quad (31)$$

公式(30)是一个各向异性的非线性热扩散方程^[20].

5.2 迭代算法

求解公式(30)的离散数值方法(例如有限元和有限差分法^[16])就是相应的广义运动模糊复原算法.本文中,我们采用有限差分的显式迭代格式来进行计算.令 $I^{(n)} = \{I_{i,j}^{(n)}\}$ 表示第 n 步迭代中复原出的图像, $I^{(b)} = \{I_{i,j}^{(b)}\}$ 表示模糊图像, $D_x = \{D_{i,j;x}^{(n)}\}$ 和 $D_y = \{D_{i,j;y}^{(n)}\}$ 表示梯度矩阵,即

$$D_{i,j;x}^{(n)} = I_{i,j}^{(n)} - I_{i,j-1}^{(n)}; D_{i,j;y}^{(n)} = I_{i,j}^{(n)} - I_{i-1,j}^{(n)} \quad (32)$$

首先计算:

$$P_{i,j;x}^{(n)} = \frac{D_{i,j;x}^{(n)}}{\sqrt{(D_{i,j;x}^{(n)})^2 + (D_{i,j;y}^{(n)})^2}}; P_{i,j;y}^{(n)} = \frac{D_{i,j;y}^{(n)}}{\sqrt{(D_{i,j;x}^{(n)})^2 + (D_{i,j;y}^{(n)})^2}} \quad (33)$$

然后,进一步计算求得:

$$q_{i,j}^{(n)} = P_{i,j+1;x}^{(n)} - P_{i,j;x}^{(n)} + P_{i+1,j;y}^{(n)} - P_{i,j;y}^{(n)} \quad (34)$$

$$s_{i,j}^{(n)} = \sum_k \sum_l \left(w_{i,j,k,l} \cdot \left(I_{k,l}^{(b)} - \left(\sum_{i'} \sum_{j'} w_{i',j',k,l} \cdot I_{i',j'}^{(n)} \right) \right) \right) \quad (35)$$

最终,得到了公式(30)所对应的显式迭代格式:

$$I_{i,j}^{(n+1)} = I_{i,j}^{(n)} + \lambda_1 \cdot q_{i,j}^{(n)} + \lambda_2 \cdot s_{i,j}^{(n)} \quad (36)$$

迭代的初始值选为模糊图像(即 $I^{(0)} = I^{(b)}$).

6 实验结果

现在我们来考虑实际情况——高速检测列车所拍摄的图像进行去模糊处理.车头前置摄像机获取图像为前向运动图像,因此,使用传统的反卷积模型无法取得好的图像复原效果.图 5 给出了模糊图像的复原结果.

图 6 是根据图 5(a)~图 5(d)计算出的模糊核结果.图 6(a)给出了一部分前向运动模糊核 $w_{i,j,k,l}$,每一个“小方块”的大小为 39×39 个像素点,对应于一个固定的 (k, l) .图 6(b)中给出了反卷积模型^[4]所估计出的模糊核.

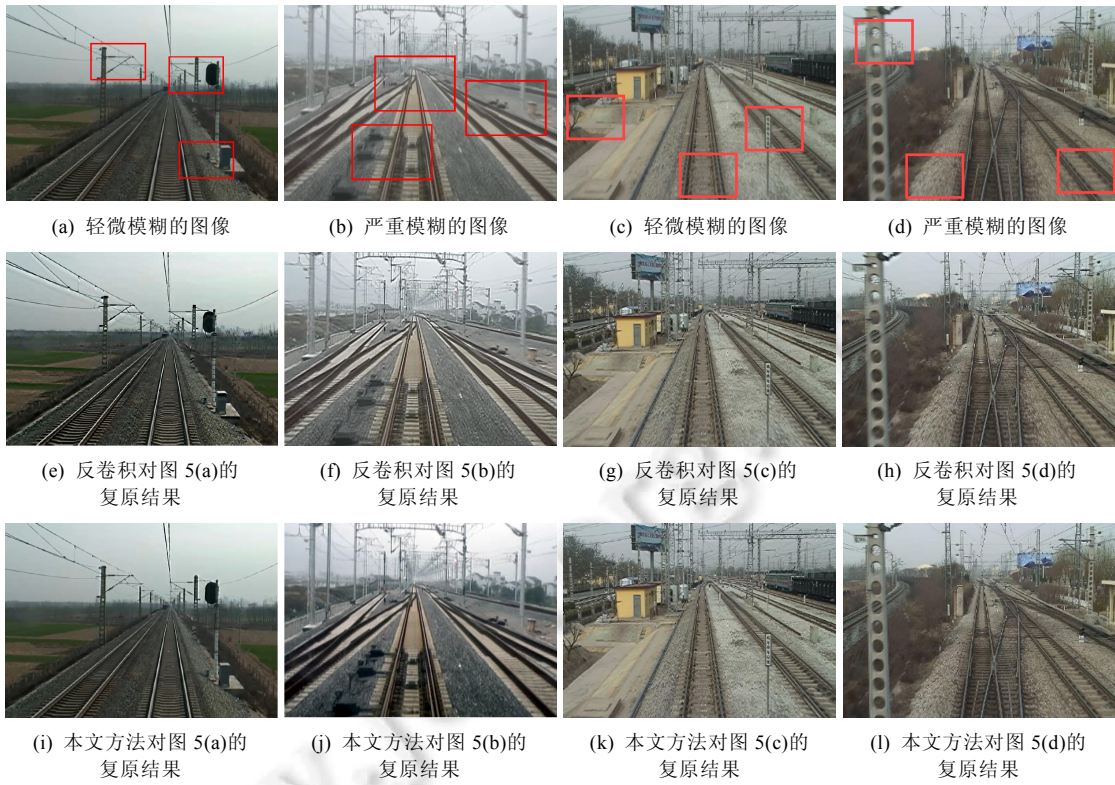


Fig.5 Restoration results of blurring images captured by patrol train-borne camera

图 5 检测列车实拍视频中的模糊图像的复原结果

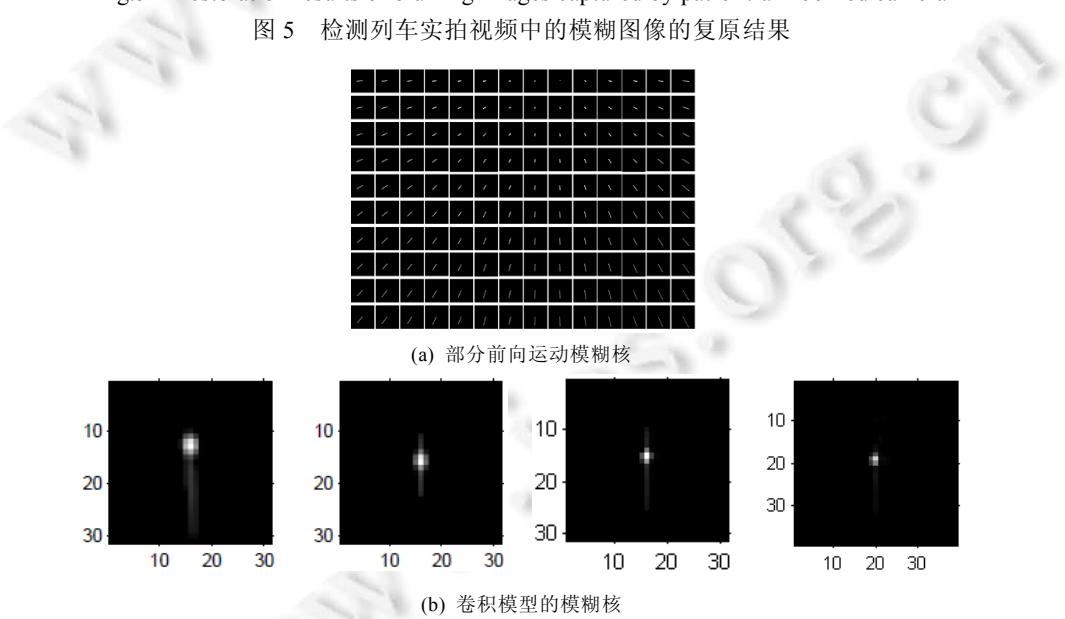


Fig.6 Blurring kernels calculation from Fig.5(a)~Fig.5(d)

图 6 根据图 5(a)~图 5(d)计算的模糊核

从图 6(b)可以看出,反卷积模型“认为”模糊是由图像的“上下晃动”产生的.基于这个错误的图像模糊假设,

复原方法将无法有效地去除图像中的运动模糊,甚至引入了严重的伪影.相比之下,我们的方法取得了较为理想的效果,有效地去除了图像中的模糊,并且在图像的中心区域,即模糊程度很小的区域,并没有产生“伪影”(反卷积模型的复原结果中存在明显的伪影(如图7所示)).

为了便于进一步对比观察,图7给出了图5中的模糊图像、反卷积模型的复原结果,即,图5(a)~图5(d)中方框圈出来的小图像块.

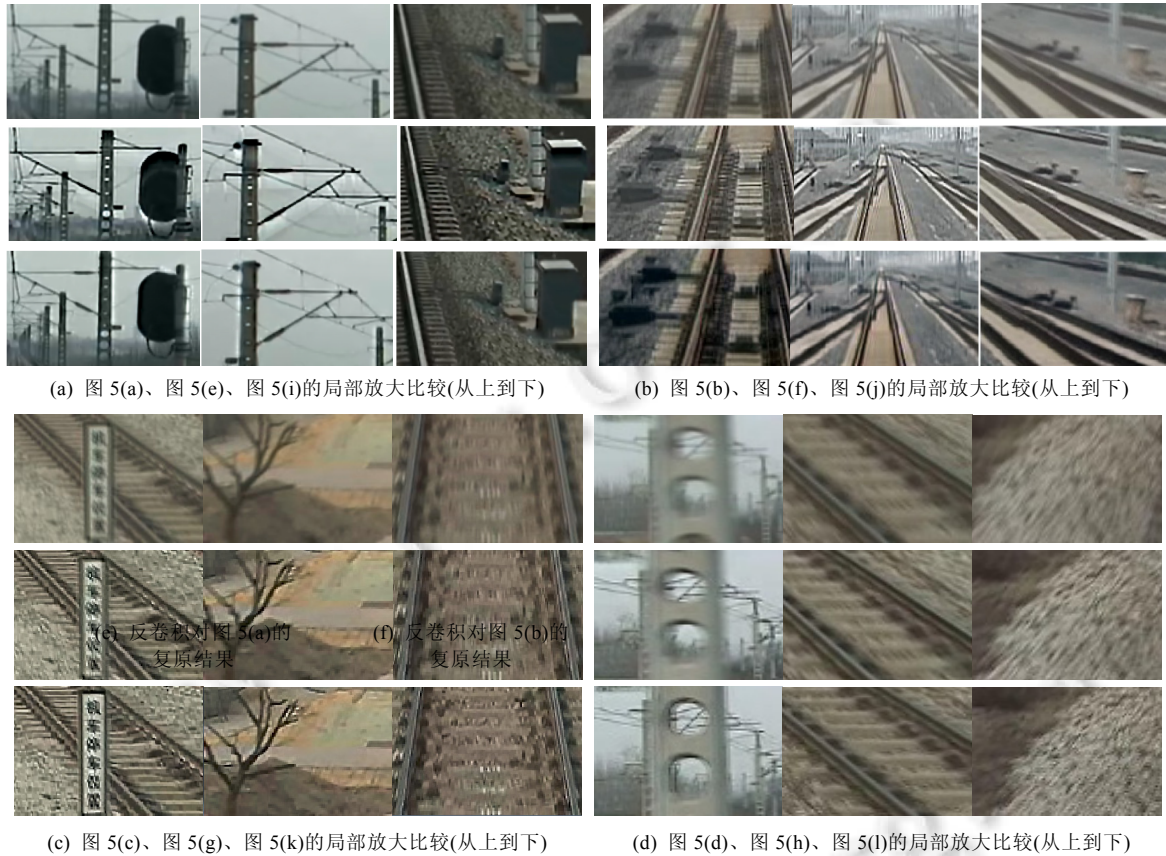


Fig.7 Local zooming comparison between deconvolution model^[4] and proposed model for blurring images

图7 对模糊图像,反卷积模型^[4]和我们的结果进行局部放大比较

7 总结

图像复原模型是一个逆问题,对于求解逆问题来说,对正问题的正确理解是极其重要的.经典的图像复原方法(即反卷积模型)将图像的模糊过程模拟为一个卷积过程.对于前向运动模糊,这个近似假设是不成立的.本文提出了一种广义运动模糊模型,在此基础上,给出了前向运动模糊模型和前向运动模糊核,其目的就是为恢复前向运动模糊找到理论依据.在这里,首先给出了基于光流场的广义运动模糊模型的理论分析;接下来给出了前向运动模糊模型的快速的运动估计方法、平面场景朝向的快速估计方法、前向运动模糊核的数值生成方法;最后,通过对实际的前向运动模糊图像的去模糊实验,验证了本文结论的正确性.

本文研究的前向运动模糊只是广义运动模糊的一种特殊情况,即便是对于相机相对于固定场景运动的特殊情况,相机的运动也可能同时包含平动和转动^[14,15].此时,估计光流和生成广义运动模糊核将变得更加困难.如何结合更加复杂的相机运动假设以及更加复杂的正则化方法^[21]设计出针对其他类型的广义运动模糊的图像复原算法,将是我们进一步研究和探索的问题.

References:

- [1] Bertero M, Boccacci P. Introduction to Inverse Problems in Imaging. London: CRC Press, 1998. 50–74.
- [2] Rudin LI, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 1992, 60(1):259–268. [doi: 10.1016/0167-2789(92)90242-F]
- [3] Levin A, Weiss Y, Durand F, Freeman WT. Understanding and evaluating blind de-convolution algorithms. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2009). Miami: IEEE, 2009. 1964–1971. [doi: 10.1109/CVPR.2009.5206815]
- [4] Cho S, Lee S. Fast motion deblurring. *ACM Trans. on Graphics*, 2009,28(5):145. [doi: 10.1145/1618452.1618491]
- [5] Fusco T, Conan JM, Mugnier LM, Michau V, Rousset G. Characterization of adaptive optics point spread function for anisoplanatic imaging. Application to Stellar Field Deconvolution, *Astronomy and Astrophysics Supplement Series*, 2000,142(1):149–156. [doi: 10.1051/aas:2000145]
- [6] Whyte O, Sivic J, Zisserman A, Ponce J. Non-Uniform deblurring for shaken images. *Int'l Journal of Computer Vision*, 2012,98(2): 168–186. [doi: 10.1007/s11263-011-0502-7]
- [7] Portz T, Zhang L, Jiang H. Optical flow in the presence of spatially-varying motion blur. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR 2012). Providence: IEEE, 2012. 1752–1759. [doi: 10.1109/CVPR.2012.6247871]
- [8] Gupta A, Joshi N, Zitnick L, Cohen M, Curless B. Single image deblurring using motion density functions. In: Proc. of the European Conf. on Computer Vision (ECCV). Heidelberg: Springer-Verlag, 2010. 171–184. [doi: 10.1007/978-3-642-15549-9_13]
- [9] Hirsch M, Schuler CJ, Harmeling S, Scholkopf B. Fast removal of non-uniform camera shake. In: Proc. of the IEEE Int'l Conf. on Computer Vision (ICCV). Barcelona: IEEE, 2011. 6–13. [doi: 10.1109/ICCV.2011.6126276]
- [10] Zheng S, Xu L, Jia J. Forward motion deblurring. In: Proc. of the IEEE Int'l Conf. on Computer Vision (ICCV). Sydney: IEEE, 2013. 1465–1472. [doi: 10.1109/ICCV.2013.185]
- [11] Horn BKP. Robot Vision. Cambridge: MIT Press, 1986. 30–448.
- [12] Horn BK, Schunck BG. Determining optical flow. *Artificial Intelligence*, 1981,17(1-3):185–203. [doi: 10.1016/0004-3702(81)90024-2]
- [13] Szeliski R. Computer Vision: Algorithms and Applications. Heidelberg: Springer-Verlag, 2010. 405–410.
- [14] Bruss AR, Horn BKP. Passive navigation. *Computer Vision, Graphics, and Image Processing*, 1983,21(1):3–20. [doi: 10.1016/S0734-189X(83)80026-7]
- [15] Negahdaripour S, Horn BKP. Direct passive navigation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1987,9(1): 168–176. [doi: 10.1109/TPAMI.1987.4767884]
- [16] Strang G. Computational Science and Engineering. Wellesley: Wellesley Cambridge Press, 2007. 128–471.
- [17] Wang SC, Zheng JY, Luo SW, Luo XY, Huang YP, Gao DL. Route panorama acquisition and rendering for high-speed railway monitoring. In: Proc. of the 2013 IEEE Int'l Conf. on Multimedia and Expo (ICME). San Jose: IEEE, 2013. 1–6. [doi: 10.1109/ICME.2013.6607592]
- [18] Bresenham JE. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 1965,4(1):25–30. [doi: 10.1147/sj.41.0025]
- [19] Horn BKP. Circle generators for display devices. *Computer Graphics and Image Processing*, 1976,5(2):280–288. [doi: 10.1016/0146-664X(76)90036-8]
- [20] Aubert G, Kornprobst P. Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations. Heidelberg: Springer Science & Business Media, 2006. 67–136. [doi: 10.1007/978-0-387-44588-5]
- [21] Wang L, Huang Y, Luo X, Wang Z, Luo S. Image deblurring with filters learned by extreme learning machine. *Neurocomputing*, 2011,74(16):2464–2474. [doi: 10.1016/j.neucom.2010.12.035]

附录:关于公式(2)和公式(3)的证明

考虑到计算机的数值计算,现写出公式(1)的离散形式.设 (x,y) 为一个像素点区域的中心, $\delta x \delta y$ 为其面积, $0 \sim T_0$ 时间段内,图像辐照强度累积量灰度值为 $I(x,y)$.若将 $0 \sim T_0$ 时间段分为 N 个间隔,那么每个小时时间段 $\Delta t = \frac{T_0}{N}$ 内图像灰度值强度为 $\frac{I(x,y)}{N}$,则公式(1)的离散形式为

$$I_b(x,y) = \frac{1}{N} \sum_{k=1}^N I(x,y,k\Delta t) \quad (37)$$

若在 $k\Delta t$ 时刻,点 $(x,y)^T$ 位于曲线 $I(x^{(k)},y^{(k)},t)$ 上,那么 $I(x,y,k\Delta t) = I_0(x^{(k)},y^{(k)})$ (依据亮度不变假设),于是有:

$$I_b(x, y) = \frac{1}{\delta x \delta y} \frac{1}{N} \sum_{k=1}^N I_0(x^{(k)}, y^{(k)}) \delta x \delta y \quad (38)$$

具体计算时,应对每个像素点进行遍历:

$$I(x, y, k\Delta t) = \sum_i \sum_j I_0(x_i, y_j) \Gamma(l(x_i, y_j, k\Delta t) = (x, y)^T) \quad (39)$$

其中定义了一个二值表征函数 $\Gamma(F)$ 来判断点 $(x, y)^T$ 是否位于曲线 $l(x_i, y_j, k\Delta t)$ 上,即

$$\Gamma(F) = \begin{cases} 1, & \text{如果 } F \text{ 成立} \\ 0, & \text{否则} \end{cases} \quad (40)$$

将公式(39)代入公式(37),有:

$$\begin{aligned} I_b(x, y) &= \frac{1}{\delta x \delta y} \frac{1}{N} \sum_{k=1}^N \left(\sum_i \sum_j I_0(x_i, y_j) \Gamma(l(x_i, y_j, k\Delta t) = (x, y)^T) \right) \delta x \delta y \\ &= \sum_i \sum_j \left(\frac{1}{\delta x \delta y} \frac{1}{N} \sum_{k=1}^N \Gamma(l(x_i, y_j, k\Delta t) = (x, y)^T) \right) I_0(x_i, y_j) \delta x \delta y \end{aligned} \quad (41)$$

而

$$\lim_{N \rightarrow \infty} \sum_{k=1}^N \Gamma(l(x_i, y_j, k\Delta t) = (x, y)^T) \Delta t = T(l(x_i, y_j, t) \in A(x, y)) \quad (42)$$

公式(41)可改写为

$$I_b(x, y) = \sum_i \sum_j \left(\frac{1}{\delta x \delta y} \frac{1}{T_0} T(l(x_i, y_j, t) \in A(x, y)) \right) I_0(x_i, y_j) \delta x \delta y \quad (43)$$

令 $\delta x \rightarrow 0, \delta y \rightarrow 0$, 公式(43)可写为积分形式:

$$I_b(x, y) = \iint \left(\frac{1}{\delta x \delta y} \frac{T(l(x_1, y_1, t) \in A(x, y))}{T_0} \right) I_0(x_1, y_1) dx_1 dy_1 \quad (44)$$

其中,

$$\frac{1}{\delta x \delta y} \frac{T(l(x_1, y_1, t) \in A(x, y))}{T_0} = w(x, y, x_1, y_1) \quad (45)$$

证毕. □



蒋欣兰(1976—),女,江苏常州人,博士生,讲师,主要研究领域为机器视觉,图像处理.



王胜春(1985—),男,博士,主要研究领域为机器视觉,图像处理.



王亮(1983—),男,博士,讲师,主要研究领域为机器视觉,成像中的逆问题.



罗四维(1943—),男,博士,教授,博士生导师,主要研究领域为计算机视觉.



罗晓月(1977—),女,博士,副教授,主要研究领域为逆问题,逆问题应用.