

基于主动漂移矫正的运动目标跟踪算法

李宏友¹ 汪同庆¹ 叶俊勇¹

摘要 针对传统的图像仿射配准 (Affine image alignment, AIA) 算法无法有效兼顾算法效率和鲁棒性的问题, 提出了一种快速鲁棒的新仿射配准算法——主动漂移矫正 (Active drift correction, ADC) 算法用于跟踪视频运动目标。该算法的基本思想是: 通过引入一个漂移矫正项, 来改进传统算法的目标能量函数, 使算法具备抗漂移的能力, 从而提高算法的鲁棒性。改进后的算法不需要传统算法中为增强鲁棒性而采用的许多复杂措施 (如: 被动漂移矫正)。实验结果表明: 本文提出的算法简单、有效, 在不牺牲算法效率和复杂度的条件下, 可以获得比传统算法更好的鲁棒性。

关键词 运动目标跟踪, 图像仿射配准算法, 主动漂移矫正, 鲁棒性
中图分类号 TP391

Tracking Moving Target Using Active Drift Correction Algorithm

LI Hong-You¹ WANG Tong-Qing¹ YE Jun-Yong¹

Abstract For the traditional problem of non-compatibility between the robustness and the efficiency of affine image alignment algorithm, we propose a new fast and robust affine image alignment algorithm called ADC (active drift correction) for tracking the visual moving target. The basic idea of our method is: By incorporating a drift correction term into the traditional goal energy function, the new algorithm has the natural ability of anti-drift, which then, therefore, can boost the performance of robustness. Many extra techniques (typically as passive drift correction) in traditional methods for high robustness are no longer necessary in our algorithm. The experimental results show that our algorithm is simple and efficient. It can attain a higher performance of robustness than the traditional methods without compromise with the complexity or real time performance of the algorithm.

Key words Moving target tracking, affine image alignment algorithm, active drift correction (ADC), robustness

自 Lucas 等^[1]1981 年首次提出以来, 图像仿射配准 (Affine image alignment, AIA) 算法逐渐发展成为计算机视觉领域应用最广的技术之一。其应用包括目标跟踪, 光流场估计, 图像拼接, 图像分层, 参数化运动估计, 医学图像配准, 人脸识别等。为了满足日益提高的应用需求, 研究者们^[2-9]在原始算法思想之上, 提出了许多富有创新的改进。其中, 研究焦点主要集中在算法的鲁棒性和效率两方面。

效率方面, Dellaert 等^[3]提出在仿射配准时将当前帧图像和模板图像两者角色进行互换, 这样可以有效避免每次迭代过程都要重新计算 Hessian 矩阵, 从而可大大提高算法效率。该思想随后被 Baker 等^[4-5]完善推广, 发展出高效率的反向合成 (Inverse compositional, IC) 算法。尽管如此, 由于

噪声 (由光照变化, 目标形变以及遮挡等引起) 的引入, 目标平面与模板在跟踪过程中几乎不可能单纯地呈仿射对应关系, 使得其鲁棒性与原始 AIA 算法一样受到严峻挑战, 具体表现为模板漂移问题。

Hager 等^[6]提出在仿射配准时只计算真实的目标像素, 而阻止噪声、背景等外部像素点 (Outliers) 参与配准, 从而提高了算法鲁棒性。该算法的主要不足在于 Hessian 矩阵的计算与外部像素划分相关, 从而也导致每次迭代均需要重新计算 Hessian 矩阵, 降低了算法效率。为此 Ishikawa 等^[7]提出采用对模板进行“分块”配准的方法, 而每一子块的 Hessian 矩阵均独立采用固定常量值而不必每次迭代都重新计算, 从而提高了算法效率。然而, 分块操作本身也是个复杂问题, 而且在跟踪过程中, 如何自动确定出每帧模板的外部像素点这一问题依然存在。无论是确定外部像素点或者分块, 都将导致算法复杂度升高。

2004 年, 内基梅隆大学的 Matthews 等^[8]提出了一种比较新颖的思想, 即从模板更新的角度来考虑 AIA 算法的鲁棒性问题。他们提出在每一帧内对当前模板跟踪完成后, 再对原始模板作第二次跟踪 (又称矫正跟踪)。由于经过两次跟踪, 算法的漂移抑制能力的确有较明显的改善。但这归根结底是一种类似亡羊补牢的被动措施, 不能最有效地发挥漂移矫正思想的作用。且该算法存在两大问题: 1) “二次跟踪”显然要消耗相当的时间, 大大降低了算法的整体效率; 2) 二次跟踪原始模板, 本质上违背了模板更新的目的, 因此在目标形变突出时的跟踪性能依然不尽人意。习惯上称该算法为被动漂移矫正 (Passive drift correction, PDC) 算法。

Schreiber^[9]在其最新的论文中也采用 PDC 算法来实现目标跟踪。该文算法在 PDC 算法上额外增加了一个类似“阻止外部像素”的过滤函数以及在模板更新时插入一个额外的仿射参数更新步骤。尽管这些措施在一定程度上改善了算法的鲁棒性, 但遗憾的是该算法并没有试图去从根本上解决 PDC 存在的两大固有问题。相反, 其效率较 PDC 算法更低了。受 Matthews^[8]思想的启发, 本文也利用原始模板的漂移矫正能力来增强 AIA 算法的鲁棒性, 但本文采取的是在原有跟踪过程中同步进行漂移矫正, 并不需要额外进行一次矫正跟踪, 从而能够在根本上有效地解决 PDC 算法的两大固有问题。由于本文算法在跟踪漂移尚未产生前或产生时, 就有意识地进行防范和遏制, 因此习惯上称其为主动漂移矫正 (Active drift correction, ADC) 算法。

1 背景知识

1.1 图像仿射配准算法

AIA 算法的根本目的是希望模板像素 $\mathbf{X} = (x, y)^T$ 经过仿射变换 $W(\mathbf{X}; \mathbf{P})$ 后映射在当前图像 I 中的子图像 $I(W(\mathbf{X}; \mathbf{P}))$ 与原始模板间的累计误差最小化, 即要求表达式

$$\sum_{\mathbf{x}} [I(W(\mathbf{X}; \mathbf{P})) - T(\mathbf{X})]^2 \quad (1)$$

取得最小值。上式中 $W(\mathbf{X}; \mathbf{P})$ 为仿射变换函数

$$W(\mathbf{X}; \mathbf{P}) = \begin{bmatrix} 1 + p_1 & p_3 & p_5 \\ p_2 & 1 + p_4 & p_6 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2)$$

$\mathbf{P} = [p_1, p_2, p_3, p_4, p_5, p_6]^T$ 为仿射变换参数。

收稿日期 2007-09-27 收修改稿日期 2007-12-30
Received September 27, 2007; in revised form December 30, 2007
国家科技支撑计划项目 (2007BAG06B06)
National Key Technology R&D Program (2007BAG06B06)
1. 重庆大学光电技术及系统教育部重点实验室 重庆 400030
1. Key Laboratory of Optoelectronic Technique and System of Ministry of Education, Chongqing University, Chongqing 400030
DOI: 10.3724/SP.J.1004.2009.00310

易见, AIA 算法把对目标像素的跟踪问题间接化为: “寻找使表达式 (1) 取得最小值的仿射变换参数 \mathbf{P} 。”这一非线性优化问题. 一般利用最速迭代下降法求解这一优化问题, 即在已知 \mathbf{P} 当前估计值的前提下, 反复迭代累加 $\Delta\mathbf{P}$, 确保每次迭代时关于 $\Delta\mathbf{P}$ 的目标表达式

$$\sum_{\mathbf{X}} [I(W(\mathbf{X}; \mathbf{P} + \Delta\mathbf{P})) - T(\mathbf{X})]^2 \quad (3)$$

最小化, 并不断更新 \mathbf{P} 的当前估计值, 直到迭代收敛为止.

将式 (3) 在 $\Delta\mathbf{P} = 0$ 处泰勒展开, 并求解其极值条件方程, 可得每次迭代 \mathbf{P} 的累加量

$$\Delta\mathbf{P} = H^{-1} \sum_{\mathbf{X}} \left[\nabla I \frac{\partial W}{\partial \mathbf{P}} \right]^T [T(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))] \quad (4)$$

其中 H 为 Hessian 矩阵,

$$H = \sum_{\mathbf{X}} \left[\nabla I \frac{\partial W}{\partial \mathbf{P}} \right]^T \left[\nabla I \frac{\partial W}{\partial \mathbf{P}} \right] \quad (5)$$

1.2 IC 算法和 PDC 算法

从式 (5) 中易见, 原始 AIA 算法的 Hessian 矩阵与 $\nabla I(W(\mathbf{X}; \mathbf{P}))$ 相关, 而仿射参数 \mathbf{P} 是在迭代更新的, 从而导致每次迭代都需要重新计算 Hessian 矩阵, 严重影响了算法的效率. 对此, Baker 等^[4-5] 在求解关于式 (1) 的最小化问题时, 提出了 IC 算法. 改进后的目标表达式变为

$$\sum_{\mathbf{X}} [I(W(\mathbf{X}; \mathbf{P})) - T(W(\mathbf{X}; \Delta\mathbf{P}))]^2 \quad (6)$$

对比式 (3) 和 (6) 可见, IC 算法的基本思想是: 在迭代求解仿射参数 \mathbf{P} 的过程中, 将图像区域与模板进行角色置换, 即利用模板 T 关于 $\Delta\mathbf{P}$ 仿射变换后的图像 $T(W(\mathbf{X}; \Delta\mathbf{P}))$ 去匹配原图像中的当前映射区域内的子图像 $I(W(\mathbf{X}; \mathbf{P}))$, 这与式 (3) 中两图所扮演的角色刚好相反. 与式 (6) 对应的 Hessian 矩阵变为

$$H = \sum_{\mathbf{X}} \left[\nabla T \frac{\partial W}{\partial \mathbf{P}} \right]^T \left[\nabla T \frac{\partial W}{\partial \mathbf{P}} \right] \quad (7)$$

可见, 改进后 IC 算法的 Hessian 矩阵已经不再依赖于仿射变换映射图 $I(W(\mathbf{X}; \mathbf{P}))$, 而反过来只依赖于模板 T , 因此可以在迭代开始前预先计算, 从而极大地提高了算法的效率.

关于 PDC 算法, 由于它是从模板更新的角度利用原始模板 T_0 来实现漂移矫正, 因此本质上它更像一种框架设计算法. 其原理如图 1 所示, 图中 n 表示当前是在第 n 帧图像内跟踪. 该算法在 AIA 算法迭代完成后 (即图中 “第一次

跟踪”结束后), 而在模板更新前, 以对当前模板 T 跟踪到的仿射变换参数 \mathbf{P}_n 作为初始化参数, 再对原始模板 T_0 作二次跟踪 (又称矫正跟踪), 获得矫正后的仿射变换参数 \mathbf{P}^* , 并以此参数来更新模板. 显然, 为了达到漂移矫正的目的, PDC 算法多消耗了一次跟踪所需的运行时间, 极大地损害了整体算法的跟踪效率. 而且对于每一帧, 用于更新模板的图像 $I_n(W(\mathbf{X}; \mathbf{P}^*))$ 始终是跟踪匹配原始模板 T_0 的产物, 这在一定程度上违背了实际应用中需要不断更新模板的初衷.

2 主动漂移矫正算法 (ADC)

2.1 算法思想

研究中我们发现, 传统算法及其改进算法的着眼点基本上都是单纯地希望最小化与当前模板的累计平方误差, 因此, 一旦某一帧的跟踪模板出现噪声则会在后续跟踪中逐渐累加, 形成误差累计效应. 从 PDC 算法中, 我们了解到原始模板 T_0 具有漂移矫正的能力, 但如前所述 PDC 所采用的矫正方法本身有严重的缺陷, 因此, 希望找到一种方法, 使得原始模板 T_0 的漂移矫正功能得到充分的发挥, 但又不至于带来如 PDC 算法那样的副作用.

要实现这一目标, 最佳方法莫过于在 AIA 算法体内直接融入漂移矫正信息, 化被动漂移矫正为主动矫正, 即: 在跟踪漂移产生以前或产生的同时, 便利用 T_0 来防范或遏制. 具体地, 本文将传统已有的跟踪信息和加入的漂移矫正信息看作是不同的能量成分, 由它们构成一个总体能量, 通过最小化总能量来同步实现原有的跟踪和漂移矫正, 这可被称作 “合成目标能量” 思想. 经反复研究和试验, 本文提出如下的合成目标能量函数:

$$E = (1 - \alpha) \cdot E_T + \alpha \cdot E_{DC} \quad (8)$$

$$E_T = \sum_{\mathbf{X}} [T(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))]^2 \quad (9)$$

$$E_{DC} = \sum_{\mathbf{X}} [T_0(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))]^2 \quad (10)$$

式中, $\alpha \in [0, 1]$ 为漂移矫正系数; T 表示当前跟踪模板; T_0 为初始模板; E_T 是关于当前模板 T 的目标能量成分, 它代表着当前仿射映射区域的子图像 $I(W(\mathbf{X}; \mathbf{P}))$ 与当前模板 T 之间的差异程度; E_{DC} 是关于原始模板 T_0 (又称矫正模板) 的目标能量成分, 它反映了 $I(W(\mathbf{X}; \mathbf{P}))$ 与 T_0 之间的差异程度; 而总体能量 E 则有所侧重地反映了 $I(W(\mathbf{X}; \mathbf{P}))$ 与当前模板 T 和矫正模板 T_0 各自差异的总差异.

ADC 算法的目的是希望合成目标能量 E 最小化, 即

$$\mathbf{P} = \arg \min_{\mathbf{P}} E \quad (11)$$

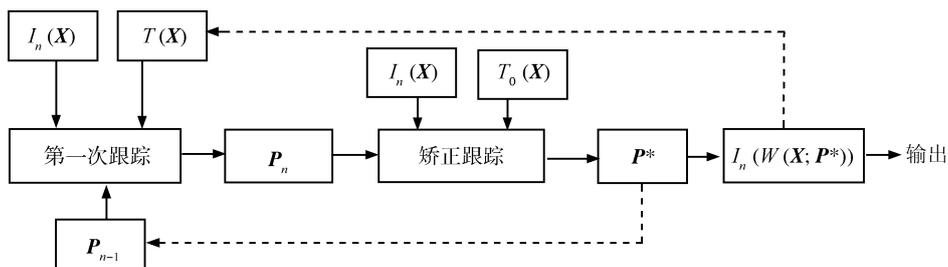


图 1 PDC 算法原理

Fig. 1 Framework of the PDC algorithm

需要特别强调的是, $I(W(\mathbf{X}; \mathbf{P}))$ 具体是指模板像素坐标 $\mathbf{X} = (x, y)^T$ 经仿射变换后的坐标集合 $W(\mathbf{X}; \mathbf{P})$ 映射在当前帧图像 I 上的像素集合(取像素值, 如灰度值) 经过插补运算所得, 且插补均按模板尺寸为淮. 因此, $I(W(\mathbf{X}; \mathbf{P}))$ 实质上是与模板尺寸一致的一幅插补图像. 因此, 无论哪一帧的当前模板尺寸均与原始模板尺寸一样.

2.2 算法推导

为了保证算法的效率, 本文按照反向合成算法的思想来迭代求解式(11)中的非线性优化问题. 假定 \mathbf{P} 的当前估计已知, 迭代累加 $\Delta\mathbf{P}$ 进行求解, 于是式(8)变为如下形式

$$E = (1 - \alpha) \cdot \sum_{\mathbf{x}} [T(W(\mathbf{X}; \Delta\mathbf{P})) - I(W(\mathbf{X}; \mathbf{P}))]^2 + \alpha \cdot \sum_{\mathbf{x}} [T_0(W(\mathbf{X}; \Delta\mathbf{P})) - I(W(\mathbf{X}; \mathbf{P}))]^2 \quad (12)$$

将 $E(\Delta\mathbf{P})$ 在 $\Delta\mathbf{P} = 0$ 处进行泰勒展开, 近似有

$$E = (1 - \alpha) \cdot \sum_{\mathbf{x}} [T(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P})) + \nabla T \frac{\partial W}{\partial \mathbf{P}} \Delta\mathbf{P}]^2 + \alpha \cdot \sum_{\mathbf{x}} [T_0(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P})) + \nabla T_0 \frac{\partial W}{\partial \mathbf{P}} \Delta\mathbf{P}]^2 \quad (13)$$

其中, $\frac{\partial W}{\partial \mathbf{P}}$ 为雅可比矩阵. 再对 $\Delta\mathbf{P}$ 求导可得

$$\frac{\partial E}{\partial \Delta\mathbf{P}} = 2(1 - \alpha) \cdot \sum_{\mathbf{x}} \left[\nabla T \frac{\partial W}{\partial \mathbf{P}} \right]^T \left[T(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P})) + \nabla T \frac{\partial W}{\partial \mathbf{P}} \Delta\mathbf{P} \right] + 2\alpha \cdot \sum_{\mathbf{x}} \left[\nabla T_0 \frac{\partial W}{\partial \mathbf{P}} \right]^T \left[T_0(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P})) + \nabla T_0 \frac{\partial W}{\partial \mathbf{P}} \Delta\mathbf{P} \right] \quad (14)$$

令式(14)中 $\frac{\partial E}{\partial \Delta\mathbf{P}} = 0$ 可得

$$\Delta\mathbf{P} = -H^{-1} \left\{ (1 - \alpha) \cdot \sum_{\mathbf{x}} \left[\nabla T \frac{\partial W}{\partial \mathbf{P}} \right]^T [T(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))] + \alpha \cdot \sum_{\mathbf{x}} \left[\nabla T_0 \frac{\partial W}{\partial \mathbf{P}} \right]^T [T_0(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))] \right\} \quad (15)$$

其中, 合成的 Hessian 矩阵为

$$H = \sum_{\mathbf{x}} \left\{ (1 - \alpha) \cdot \left[\nabla T \frac{\partial W}{\partial \mathbf{P}} \right]^T \left[\nabla T \frac{\partial W}{\partial \mathbf{P}} \right] + \alpha \cdot \left[\nabla T_0 \frac{\partial W}{\partial \mathbf{P}} \right]^T \left[\nabla T_0 \frac{\partial W}{\partial \mathbf{P}} \right] \right\} \quad (16)$$

易见, 上式中 Hessian 矩阵只与当前模板 T 和初始模板 T_0 有关, 因此可以在迭代开始前预先计算, 从而保证了算法的高效率. 最后, 更新仿射变换区域

$$W(\mathbf{X}; \mathbf{P}) = W(\mathbf{X}; \mathbf{P}) \circ W(\mathbf{X}; \Delta\mathbf{P})^{-1} \quad (17)$$

此处, $W(\mathbf{X}; \mathbf{P})$ 是指模板像素坐标 $\mathbf{X} = (x, y)^T$ 经仿射参数 \mathbf{P} 变换后的坐标集合, 其尺寸与原始模板尺寸一致, 即坐标

变量 \mathbf{X} 的取值范围为原始模板像素的坐标范围(尺寸范围). 特别地, 符号“ \circ ”表示将 $W(\mathbf{X}; \mathbf{P})$ 和 $W(\mathbf{X}; \Delta\mathbf{P})^{-1}$ 进行合成运算, 具体就是将仿射参数 \mathbf{P} 和 $\Delta\mathbf{P}$ 分别补充一行元素“[0, 0, 1]”整形为 3 行 3 列, 再对整形后的 $\Delta\mathbf{P}$ 求逆, 并将其 (0, 0), (1, 1) 两个元素分别加 1. 随后对调整后的两个矩阵进行乘运算, 并将结果矩阵的 (0, 0), (1, 1) 两个元素分别减 1. 取出前两行作为更新后的仿射参数, 并按式(2)计算更新后的仿射区域. 关于合成运算, 具体还可以参考文献[5].

2.3 算法实现步骤

给定当前帧图像 $I_n(\mathbf{X})$, 原始模板 T_0 , 当前模板 T , 以及漂移矫正能量权重系数 α , ADC 算法可概括为预先计算和迭代计算两个阶段, 具体步骤如下:

ADC 算法 (Active drift correction)

预先计算:

- 1) 分别计算两模板的梯度: ∇T_0 和 ∇T ;
- 2) 计算 $(\mathbf{X}; \mathbf{0})$ 处的雅可比矩阵: $\frac{\partial W}{\partial \mathbf{P}}$;
- 3) 分别计算两模板的最速下降图像: $\nabla T_0 \frac{\partial W}{\partial \mathbf{P}}$, $\nabla T \frac{\partial W}{\partial \mathbf{P}}$;
- 4) 按式(16)计算 Hessian 矩阵 H .

迭代计算:

- 5) 将当前帧图像 $I_n(\mathbf{X})$, 取 $W(\mathbf{X}; \mathbf{P})$ 位置的像素按模板尺寸插补为 $I(W(\mathbf{X}; \mathbf{P}))$;
- 6) 分别计算误差图像: $T_0(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))$, $T(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))$;
- 7) 分别计算累计误差: $\sum_{\mathbf{x}} \left[\nabla T \frac{\partial W}{\partial \mathbf{P}} \right]^T [T(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))]$ 和 $\sum_{\mathbf{x}} \left[\nabla T_0 \frac{\partial W}{\partial \mathbf{P}} \right]^T [T_0(\mathbf{X}) - I(W(\mathbf{X}; \mathbf{P}))]$;
- 8) 按式(15)计算 $\Delta\mathbf{P}$;
- 9) 按式(17)更新仿射变化区域.

需要指出的是, 出于为第 $n+1$ 帧的跟踪做准备, 当第 n 帧跟踪完毕后, ADC 算法并不需要任何二次跟踪步骤, 仅采用最简单的策略更新模板即可, 即

$$T_{n+1}(\mathbf{X}) = I(W(\mathbf{X}; \mathbf{P})) \quad (18)$$

3 目标跟踪实验及分析

本文实验均在一台 CPU 为奔腾 4 3.0GHz, 内存 512MB 的 PC 机上完成, 开发平台为 Matlab 7.1 版. 实验主要采用 ADC 算法与 Matthews 等^[8]提出的 PDC 算法做性能比较的方式进行. 同时还简单讨论了 ADC 算法中漂移矫正系数的选取问题.

图 2 (见下页) 给出在 PETS2001 视频序列 (第 2001~2500 帧) 上的测试结果. 实验以最前端的车辆尾部 (初始化为矩形模板) 作为目标, 跟踪过程一直持续到目标车辆因超车改道消失为止. 图中左列显示的是被动漂移矫正算法 (漂移矫正更新) 的运行结果, 右列是本文算法的跟踪结果. 被动漂移矫正尽管在前 300 帧基本能够保证跟踪的稳定性, 但是易见, 在第 350 帧时跟踪结果已经出现了明显的漂移, 400 帧时漂移加剧, 450 帧时算法已经发散, 跟踪彻底失败. 而本文提出的 ADC 算法运行结果显示, 直到目标超车被遮挡消失前一刻算法都能持续、准确地跟踪到目标.

值得说明的是, 本文实验只对跟踪精度做了定性比较而没有考虑定量分析. 原因在于实际中几乎不可能获得连续准确的目标像素区域, 即使如文献[9]采用手工确定的方法, 实际上也只能得出一个定性结果.

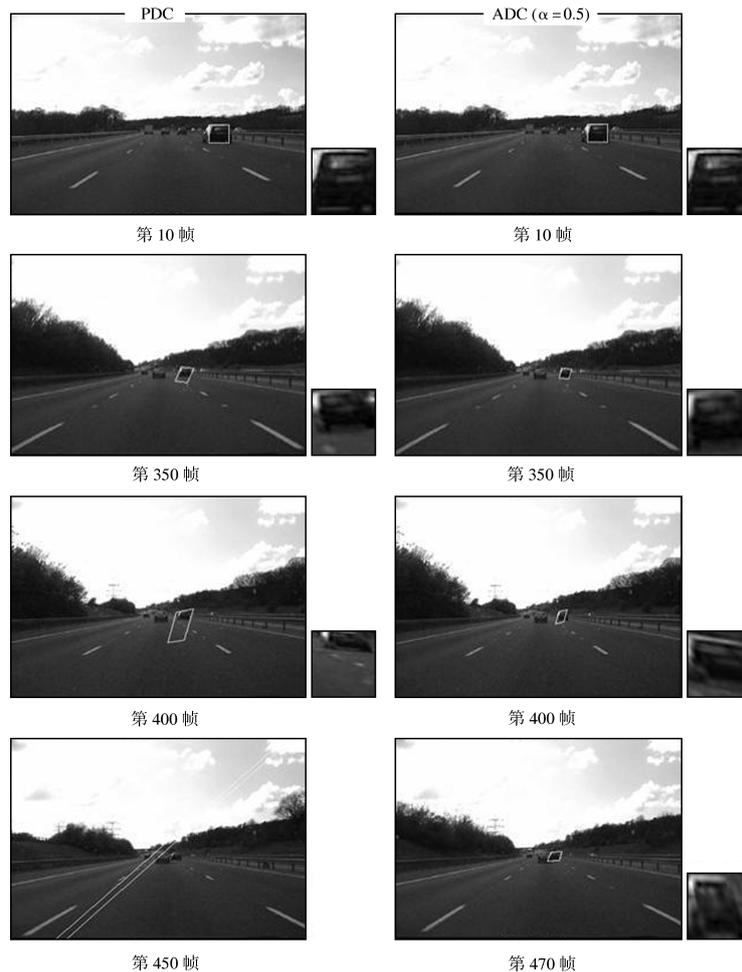


图 2 定性比较主动漂移矫正算法 ADC 与被动漂移矫正算法 PDC

Fig. 2 Qualitative comparison between ADC and PDC

图 3 (见下页) 是在测试序列“PETS 2001-Dataset 5”第 501~750 帧上的实验结果. 图 3 (d) 给出了首帧 (第 501 帧) 图像及跟踪的初始模板. 图 3 (a) 和 (b) 为算法的效率测试结果. 从图 3 (a) 的运行时间上看, 本文提出算法平均所需时间约 0.75 s, 仅为原有算法一半左右. 图 3 (b) 中比较每帧跟踪所需的迭代次数显示, ADC 算法也比原有的被动矫正算法节约了近一半的次数. 这充分表明本文提出的算法不仅有较好的鲁棒性, 而且在算法效率上也有出色的表现.

图 3 (c) 给出了选取不同的 α 值时, ADC 算法的跟踪测试情况. 比较方法选用以初始模板 T_0 为统一的参照模板, 求取每种 α 值下的跟踪结果与 T_0 的均方根误差 (Root of mean square error, RMSE). 测试结果显示, 当 α 取值 0.5, 0.8, 0.2, 0.9 时, 跟踪结果基本一致, 实际观测时它们的跟踪情况也一直比较稳定, 没有出现明显漂移. 而当 α 取 0.1 时, 得出的 RMSE 明显比其他取值下的结果偏高, 实际中也表现出到跟踪后期出现了轻微的漂移. 特别地, 当 α 取 0 (即不加入任何矫正信息) 时, 得出的 RMSE 异常之大, 平均超过 50, 最高达到 80, 远远高于图中显示的所有取值情况, 实际观测到的跟踪结果出现了异常严重的漂移, 在 120 帧以后跟踪已经彻底失败. 实验结果表明: 加入一定量的漂移矫正信息 ($\alpha \neq 0$) 能够很明显起到抑制漂移的作用; 其次, 取值不同时, 跟踪精度并不完全相等, 但一般场合下 α 有较大的选取范围,

对有效跟踪目标而言并不敏感.

然而, 需要指出的是, 在目标主要部分发生改变场合, α 的选取范围会缩小, 其敏感性会酌情上升. 比如本次实验中的车头模板 (见图 3 (c)), 若一个车灯逐渐消失 (脱离视场), 或者因为转向导致车头部分的小部分车身也需要被跟踪时, 在类似这样的目标形变严重的场合, α 的敏感性将会有所上升. 具体上升多少, 要视不同的情况而定. 在目前我们的研究中, 一般而言主要依据如下两条原则来选取 α : 视频序列中噪声越大, 则 α 取值越大; 目标形变 (主要指有目标主要成分发生改变的情况) 越小, 则 α 取值越大, 反之亦然.

4 结论

本文提出了一种快速鲁棒的新图像仿射配准 (AIA) 算法, 即主动漂移矫正 (ADC) 算法. 该算法有两大主要特点: 1) 主动引入原始模板信息来实现漂移矫正, 充分发挥了漂移矫正思想的作用, 提高了算法的鲁棒性; 2) 由于采用反向合成思想, 并且不需要花时间来二次跟踪^[5, 8], 因此算法效率非常高. 本文首次提出并应用了“合成目标能量函数”的思想, 这为解决 AIA 算法的其他问题提供了又一种选择方向.

下一步, 我们将尝试采用动态的漂移矫正系数来完善本文提出的主动漂移矫正算法.

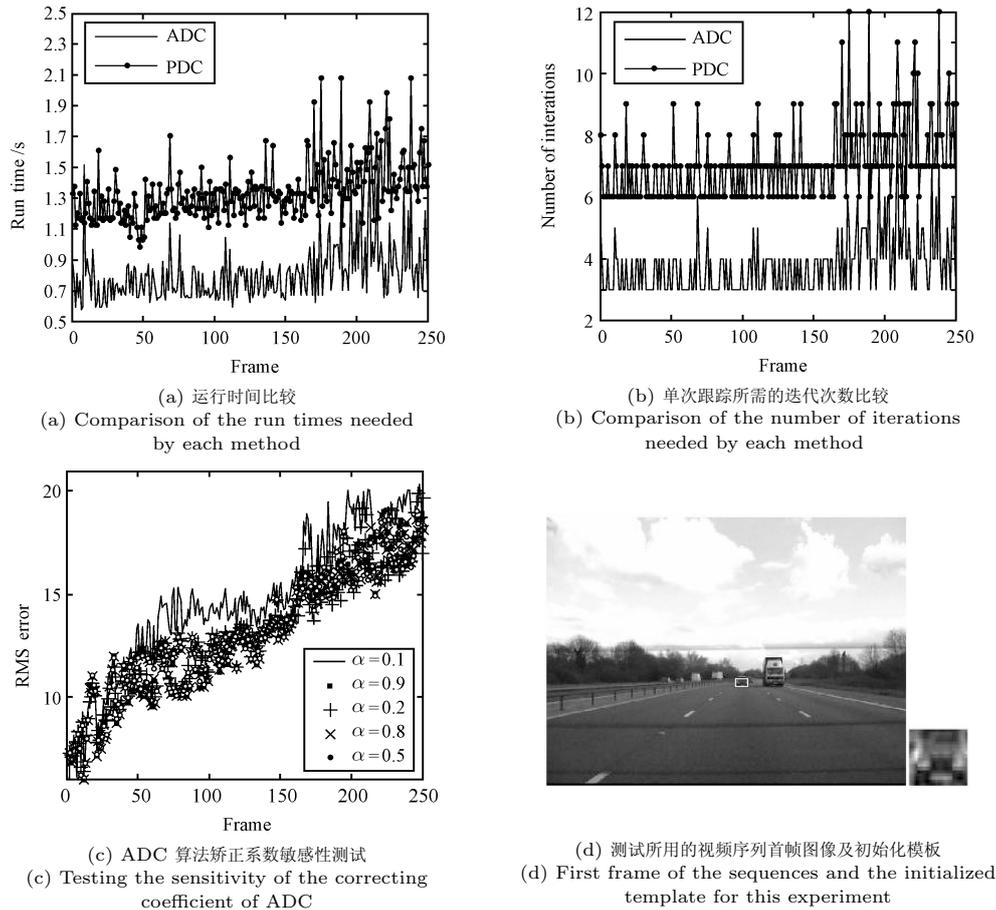


图 3 算法效率比较及 ADC 修正系数的敏感性测试

Fig. 3 Comparing the efficiency of ADC with PDC and testing the sensitivity of the correcting coefficient of ADC

References

- Lucas B D, Kanade T. An iterative image registration technique with an application to stereo vision. In: Proceedings of the 1981 DARPA Image Understanding Workshop. Vancouver, Morgan: Kaufmann Publishers, 1981. 121–130
- Zhang Pei, Wu Ya-Feng. Improved inverse compositional algorithm and operators comparison. *Journal of Computer Applications*, 2007, **27**(3): 669–672
(张培, 吴亚锋. 一种改进的反向合成算法及其算子比较. 计算机应用, 2007, **27**(3): 669–672)
- Dellaert F, Collins R. Fast image-based tracking by selective pixel integration. In: Proceedings of the ICCV Workshop on Frame-rate Vision. Corfu, Greece: IEEE, 1999. 1–22
- Baker S, Matthews I. Equivalence and efficiency of image alignment algorithms. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Kauai: IEEE, 2001. 1090–1097
- Baker S, Matthews I. Lucas-Kanade 20 years on: a unifying framework. *International Journal of Computer Vision*, 2004, **56**(3): 221–255
- Hager G D, Belhumeur P N. Efficient region tracking with parametric models of geometry and illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, **20**(10): 1025–1039
- Ishikawa T, Matthews I, Baker S. Efficient Image Alignment with Outlier Rejection, Technical Report CMU-RI-TR-02-27, Carnegie Mellon University, Robotics Institute, 2002

[Online], available: http://www.ri.cmu.edu/people/ishikawa_takahiro.html, January 1, 2007

- Matthews I, Ishikawa T, Baker S. The template update problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, **26**(6): 810–815
- Schreiber D. Robust template tracking with drift correction. *Pattern Recognition Letters*, 2007, **28**(12): 1483–1491

李宏友 重庆大学光电工程学院博士研究生. 主要研究方向为计算机视觉和智能系统. 本文通信作者. E-mail: hylea@163.com

(LI Hong-You Ph.D. candidate at the College of Optoelectronic Engineering, Chongqing University. His research interest covers computer vision and intelligent system. Corresponding author of this paper.)

汪同庆 重庆大学光电工程学院教授. 主要研究方向为模式识别、计算机视觉、无损检测. E-mail: ocr@cqu.edu.cn

(WANG Tong-Qing Professor at the College of Optoelectronic Engineering, Chongqing University. His research interest covers pattern recognition, computer vision, and nondestructive testing.)

叶俊勇 重庆大学光电工程学院副教授. 主要研究方向为模式识别和人工智能. E-mail: ygyocr@cqu.edu.cn

(YE Jun-Yong Associate professor at the College of Optoelectronic Engineering, Chongqing University. His research interest covers pattern recognition and artificial intelligence.)