

# 一致性特征点匹配在目标跟踪中的应用

李静宇 姚志军 田睿

(中国科学院 长春光学精密机械与物理研究所 长春 130033)

**摘要:**为了解决运动目标快速跟踪过程的实时性与稳定跟踪问题,提出了一种新的基于局部特征点匹配的KPM(key points matching)算法,对图像的局部多尺度特征提取与匹配进行研究。首先,应用SURF(speeded up robust features)算法在跟踪窗口内提取特征点,生成并匹配特征矢量。然后,结合最近邻提纯法与一致提纯法剔除目标区域以外的特征点对,减少误匹配以提高跟踪精度。最后,生成目标仿射变换矩阵,更新目标运动参数。实验结果表明,本文所提出的KPM算法当目标发生大角度旋转和快速缩放,同时发生光照变化时,仍能够实现稳定的跟踪,且满足运动目标实时跟踪稳定可靠、精确度高、抗干扰能力强等指标要求。

**关键词:**一致性提纯;目标跟踪;特征点匹配

**中图分类号:** TP391.41 **文献标识码:** A **国家标准学科分类代码:** 510.4050

## Target tracking based on consistency feature points matching

Li Jingyu Yao Zhijun Tian Rui

(Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China)

**Abstract:** In order to design a moving target fast tracking system with respect to a time-limited and stable tracking process, especially when the shape of moving objective or its environment condition change, a new approach based on matching local feature points named KPM (key points matching) is first proposed, and local Multi-scale feature extraction and matching technology for images are also been researched. First, based on SURF (speeded up robust features) algorithm, interest points and vectors are presented. Second, combine nearest purifying and consistency purifying to move out features outside the target area, so that we can decrease the failed matching and improve the tracking precision. Finally, generate target affine transform matrix and update the moving parameter of the target. Experimental results indicate that KPM is mostly able to achieve a stable tracking while with the monitored target rotating, scale changing, and also the environment illumination glittering. Moreover, it can satisfy the system requirements of tracking stability, higher precision and anti-jamming.

**Keywords:** consistency purity; target tracking; feature points matching

## 1 引言

目标跟踪是计算机视觉中的一项重要研究课题,随着目标跟踪技术在视频监控、安防、交通管理以及机器人引导等领域的广泛应用,引起了国内外很多学者的关注。近几年,提出了很多跟踪算法。根据目标建模方式的不同可以分为,基于区域、轮廓、局部特征等的跟踪<sup>[1-4]</sup>。

基于区域的目标跟踪算法依靠分割技术获得目标区域作为模板,通过与后继帧的匹配确定目标的实际位置。如相关匹配<sup>[5]</sup>、mean shift<sup>[6]</sup>等,这类方法比较简单,易于实现。但当目标发生部分遮挡,或者当目标颜色与背景十分接近时容易失效。基于轮廓的目标跟踪算法

利用物体的边界轮廓作为模板,目前的流行方法有主动轮廓和水平集方法<sup>[7]</sup>。这类算法一般计算量较大,而且忽略了轮廓内部的信息,当目标发生形变或遮挡时,不能实现稳定跟踪。基于局部特征的目标跟踪算法提取目标区域的特征点,与后继帧做匹配,更新目标位置。目前流行的特征提取方法有SIFT和SURF等<sup>[8-10]</sup>。这类方法当目标发生遮挡时仍然能够提取部分特征点,具有较强的抗遮挡和抗形变能力。

为了提高跟踪算法的鲁棒性,提出了一种基于局部特征的KPM(key points matching)跟踪算法。其基本思想是,利用图像中具有仿射不变性的局部特征,选择其中具有一致性运动的特征点建立目标特征点集,随着跟踪过程的

进行,目标特征点集不断丰富,包含了目标在各种运动姿态时的特征点,因此该算法具有较强的记忆跟踪能力。

## 2 KMP跟踪算法概述

如图1所示KMP跟踪算法。

1)应用SURF(speed up robust features)算法提取跟踪窗内的特征点(block 1),输入 $I_t$ 代表当前帧跟踪窗口内图像数据;

2)与目标特征点集 $F_{t-1}$ 匹配,选择出具有一致性运动的特征点(block 2);

3)利用相邻两帧的一致性特征点对计算目标区域的仿射变换矩阵 $T$ (block 3);

4)得到当前帧图像中的目标位置,最后更新目标特征点集得到 $F_t$ (block 4)。

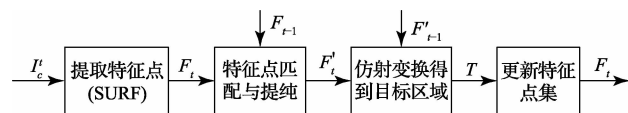


图1 KMP跟踪算法

## 3 SURF特征点提取

### 3.1 SURF特征点检测

SURF算法主要是对DoH(determinant of hessian)斑点检测算法的改进,即将DoH中的高斯二阶微分模板近似为几个矩形区域,同一矩形区域内填充相同的数值,使得模板与图像的卷积运算变成了积分图像的求和运算,从而减少了程序的运行时间,大大提高了算法的实时性。

首先定义图像中某一点 $p(i, j)$ 处的Hessian矩阵 $H(i, j, \sigma)$ :

$$H(i, j, \sigma) = \begin{bmatrix} L_{xx}(i, j, \sigma) & L_{xy}(i, j, \sigma) \\ L_{xy}(i, j, \sigma) & L_{yy}(i, j, \sigma) \end{bmatrix} \quad (1)$$

式中: $L_{xx}(i, j, \sigma)$ 是高斯二阶微分模板 $\frac{\partial^2 g(\sigma)}{\partial x^2}$ 在点 $p(i, j)$ 处与图像的卷积,其数学表达为:

$$L_{xx}(i, j, \sigma) = \sum_{x=-L/2}^{L/2} \sum_{y=-L/2}^{L/2} \frac{\partial^2}{\partial x^2} g(x, y, \sigma) p(i-x, j-y) \quad (2)$$

式中: $g(x, y, \sigma)$ 为标准高斯函数模板元素, $L$ 为模板尺寸。同理, $L_{yy}(i, j, \sigma)$ 和 $L_{xy}(i, j, \sigma)$ 具有相似定义。由于高斯二阶模板与图像的卷积运算量较大,因此SURF算法对其进行了简化。简化后的模板只由几个矩形区域组成,近似过程如图2所示。图中白色区域的值为1,黑色区域为-1,灰色区域为0。白色区域和黑色区域的宽度为高斯滤波器的响应长度,整个图像的宽度为滤波器的尺寸。近似后的DoH可以写成:

$$\det(H_{approx}) = D_{xx}D_{yy} - (\alpha D_{xy})^2 \quad (3)$$

式中: $\alpha$ 为常数,一般取 $\alpha = 0.9$ 。

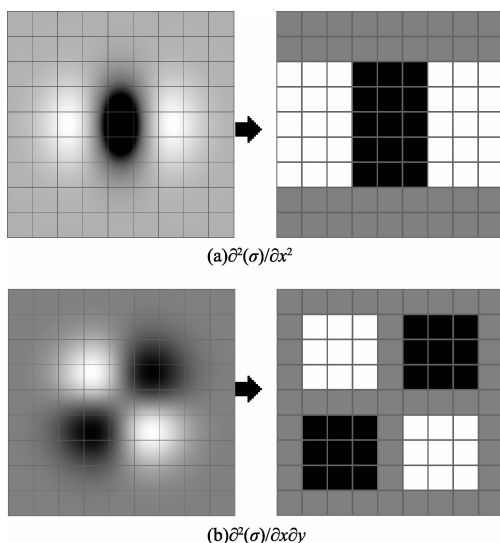


图2 高斯二阶微分模板近似

DoH斑点检测算法通过进一步求解Hessian矩阵行列式的值DoH:

$$\det(H(i, j, \sigma)) = L_{xx}(i, j, \sigma)L_{yy}(i, j, \sigma) - L_{xy}^2(i, j, \sigma) \quad (4)$$

构建所谓的“DoH特征点”检测算子,搜索位置空间和尺度空间的极值,得到DoH特征点。

### 3.2 特征向量的生成

为了保证特征矢量具有旋转不变性,与SIFT一样,需要对每个特征点分配一个主方向。然后将坐标轴旋转为特征点的主方向,以特征点为中心,划分出一个 $20s \times 20s$ 的区域, $s$ 为尺度因子。并将其划分成 $4 \times 4$ 个子块,使用尺寸为 $2s$ 的Haar小波对每一子块进行响应值统计,分别得到沿主方向的 $dy$ 和垂直于主方向的 $dx$ ,然后,以特征点为中心,对 $dy$ 和 $dx$ 进行高斯加权计算。最后,得到每个子块的矢量:

$$V_i = [\sum dx, \sum |dx|, \sum dy, \sum |dy|] \quad (5)$$

由于总共有 $4 \times 4$ 个子块,因此,特征描述子由 $4 \times 4 \times 4 = 64$ 维特征矢量组成。

## 4 特征点匹配与提纯

### 4.1 特征点匹配

特征向量生成以后,采用欧式距离作为当前帧跟踪窗口内的特征向量与目标特征集的匹配度衡量准则,欧式距离函数如下:

$$D(i, j) = \sqrt{\sum_{k=1}^{64} (F_t(i, k) - F_{t-1}^{obj}(j, k))^2} \quad (6)$$

### 4.2 特征点对提纯

虽然特征点对中的两个数据点距离最近,但这并不意味着它们对应相同的图像区域。如果目标集中没有

和  $q$  匹配的特征点,则找到的最近邻  $p$  和  $q$  的距离很大,这种匹配就是错误的。当  $q$  与基准集中若干个点有相近距离时,这时  $q$  与其最近邻构成的特征点对也有可能是错误的。为了降低误匹配的概率,要将匹配后的特征点对进行提纯处理。

1) 比值提纯法

传统的特征点对采用比值提纯法:

$$\frac{D(\mathbf{F}_c(i), \mathbf{F}_p(j))}{D(\mathbf{F}_c(i'), \mathbf{F}_p(j))} < D_{thres} \quad (7)$$

式中:  $\mathbf{F}_c(i)$  为  $\mathbf{F}_p(j)$  的最近邻特征,  $\mathbf{F}_c(i')$  为  $\mathbf{F}_p(j)$  的次近邻特征,文献中取  $D_{thres} = 0.65$ 。保留满足式(7)的特征点,剔除不满足的。

2) 一致提纯法

尽管比值提纯法可以在一定程度上减少误匹配,但仍然有一部分误匹配特征存在,为了解决上述问题,提出了一种新的一致性提纯方法,有效地解决了跟踪过程中的误匹配现象。

首先,生成当前帧与前一帧的距离函数矩阵  $\mathbf{Cur}$  与  $\mathbf{Pre}$ ,如表 1 和 2 所示。

表 1  $\mathbf{Cur}$  矩阵

$\mathbf{Cur}$	$\mathbf{F}_c(1)$	$\mathbf{F}_c(2)$	$\mathbf{F}_c(3)$	...	$\mathbf{F}_c(n)$
$\mathbf{F}_c(1)$	0	$d_{12}$	$d_{13}$	...	$d_{1n}$
$\mathbf{F}_c(2)$	$d_{12}$	0	$d_{23}$	...	$d_{2n}$
$\mathbf{F}_c(3)$	$d_{13}$	$d_{23}$	0	...	$\vdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	0	$d_{n-1n}$
$\mathbf{F}_c(n)$	$d_{1n}$	$d_{2n}$	...	$d_{n-1n}$	0

注:  $d_{ij}$  为当前帧第  $i$  个特征向量  $\mathbf{F}_c(i)$  与第  $j$  个特征向量  $\mathbf{F}_c(j)$  的欧式距离。

表 2  $\mathbf{Pre}$  矩阵

$\mathbf{Pre}$	$\mathbf{F}_p(1)$	$\mathbf{F}_p(2)$	$\mathbf{F}_p(3)$	...	$\mathbf{F}_p(n)$
$\mathbf{F}_p(1)$	0	$d_{12}'$	$d_{13}'$	...	$d_{1n}'$
$\mathbf{F}_p(2)$	$d_{12}'$	0	$d_{23}'$	...	$d_{2n}'$
$\mathbf{F}_p(3)$	$d_{13}'$	$d_{23}'$	0	...	$\vdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	0	$d_{n-1n}'$
$\mathbf{F}_p(n)$	$d_{1n}'$	$d_{2n}'$	...	$d_{n-1n}'$	0

注:  $d_{ij}'$  为与  $\mathbf{F}_c(i)$  和  $\mathbf{F}_c(j)$  相匹配的  $\mathbf{F}_p(j)$  和  $\mathbf{F}_p(j)$  之间的欧式距离。

设相邻两帧的目标发生仿射变换,因此理论上当前帧与前一帧图像中任意两特征向量应具有相同的比例关系即  $d_{ij}/d_{ij}' = k$ 。但由于存在误匹配,使得某些匹配对的  $k$  值偏大,有些偏小,因此,首先将  $n$  个匹配的特征点对的  $k$  值由大到小进行排序,然后选取中间值确定为正确匹配点对的比值  $k^*$ 。

将得到的  $k^*$  与矩阵  $\mathbf{Pre}$  中的  $d_{ij}'$  分别相乘得到  $\mathbf{Pre}'$ ,如表 3 所示。

表 3  $\mathbf{Pre}'$  矩阵

$\mathbf{Pre}'$	$\mathbf{F}_p(1)$	$\mathbf{F}_p(2)$	$\mathbf{F}_p(3)$	...	$\mathbf{F}_p(n)$
$\mathbf{F}_p(1)$	0	$k^* d_{12}'$	$k^* d_{13}'$	...	$k^* d_{1n}'$
$\mathbf{F}_p(2)$	$k^* d_{12}'$	0	$k^* d_{23}'$	...	$k^* d_{2n}'$
$\mathbf{F}_p(3)$	$k^* d_{13}'$	$k^* d_{23}'$	0	...	$\vdots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	0	$k^* d_{n-1n}'$
$\mathbf{F}_p(n)$	$k^* d_{1n}'$	$k^* d_{2n}'$	...	$k^* d_{n-1n}'$	0

通过比较  $\mathbf{Pre}$  与  $\mathbf{Pre}'$  可以判断出  $\mathbf{Cur}$  与  $\mathbf{Pre}$  中的特征点是否具有 consistency 运动。具体方式如下。

1) 利用矩阵  $\mathbf{T}$  记录比较结果:

$$\mathbf{T}(i, j) = \mathbf{T}(j, i) = \begin{cases} 1 & |d_{ij} - k^* d'_{ij}| < T_{thres} \\ 0 & |d_{ij} - k^* d'_{ij}| > T_{thres} \end{cases} \quad (8)$$

式中:  $T_{thres}$  为阈值常数,取  $T_{thres} = 5$ ,例如  $\mathbf{T}$  可以表示成式(9)的形式:

$$\mathbf{T} = \begin{bmatrix} 0 & 1 & 0 & \dots & 1 \\ 1 & 0 & 1 & \dots & 1 \\ 0 & 1 & 0 & \vdots & \vdots \\ \vdots & \vdots & \vdots & 0 & 1 \\ 1 & 1 & \dots & 1 & 0 \end{bmatrix} \quad (9)$$

2) 定义特征点对一致性运动法则

令  $C(i) = \sum_{j=1}^n T(i, j)/n$  代表第  $i$  对特征点的一致性度量,  $C_{thres}$  为阈值常数,一般为特征点对数目的 1/3,即  $C_{thres} = n/3$ 。如果:

$$C(i) < C_{thres} \quad (10)$$

则第  $i$  对特征点符合一致性运动,否则不符合,此特征点对将被剔除。

### 5 更新目标区域

假设相邻两帧的目标只发生旋转和缩放,则相邻两帧目标区域的坐标变换如式(11)所示:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} s_x \cos(\theta) & s_x \sin(\theta) \\ s_y \sin(\theta) & s_y \cos(\theta) \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (11)$$

已知  $(x, y)$  为前一帧图像目标区域内的像素点的坐标,则  $(x', y')$  为所求的当前帧图像目标区域内的像素点的坐标。式中有 5 个未知参数  $(s_x, s_y, \theta, t_x, t_y)$ ,  $s_x$  和  $s_y$  分别为目标横向与纵向的缩放因子,  $\theta$  为目标发生旋转的角度,  $t_x$  和  $t_y$  为目标平移变化量。因此只要已知 3 对特征点,就可以解出方程中的未知参数。当特征点对的数目大于 3 时,可利用最小二乘法求解。令:

$$\mathbf{A} = \begin{bmatrix} -s_x \cos \theta & -s_x \sin \theta & t_x \\ s_y \sin \theta & s_y \cos \theta & t_y \end{bmatrix}, \mathbf{X}' = \begin{bmatrix} x'_1 & x'_2 & \dots & x'_n \\ y'_1 & y'_2 & \dots & y'_n \end{bmatrix}, \mathbf{X} = \begin{bmatrix} x_1 & x_2 & \dots & x_n \\ y_1 & y_2 & \dots & y_n \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

$\mathbf{X}'$  为当前帧图像的特征点坐标矩阵,  $\mathbf{X}$  为前一帧图像的特征点坐标矩阵, 由式(11)可知  $\mathbf{X}' = \mathbf{A}\mathbf{X}$ 。因此, 可得到  $\mathbf{A} = \mathbf{X}'\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}$ , 矩阵  $\mathbf{A}$  可以写成式(12)的形式:

$$\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix} \quad (12)$$

通过求解可以得到:

$$\begin{cases} s_x = \text{sqr}(a_{11}^2 + a_{12}^2) \\ s_y = \text{sqr}(a_{21}^2 + a_{22}^2) \\ \theta = \arctan(a_{21}/a_{22}) \\ t_x = a_{13} \\ t_y = a_{23} \end{cases} \quad (13)$$

## 6 实验结果

应用  $720 \times 576$  的真彩色视频图像并以其中的汽车为目标进行跟踪实验。图3为应用本文提出的 KPM 算法对汽车的跟踪实验效果图。每幅图的左上角均为目标区域放大显示的结果。图中四边形为所定位的目标尺寸与位置, 四边形内的红色圆圈代表 SURF 算法检测出的较亮的特征点, 蓝色圆圈为检测出的较暗的特征点。

从图中看出当目标发生旋转(第 50、130、200 帧)、缩放(第 80、200、240 帧)以及光照变化(第 80、130 帧)时, 本文算法均能实现稳定跟踪。

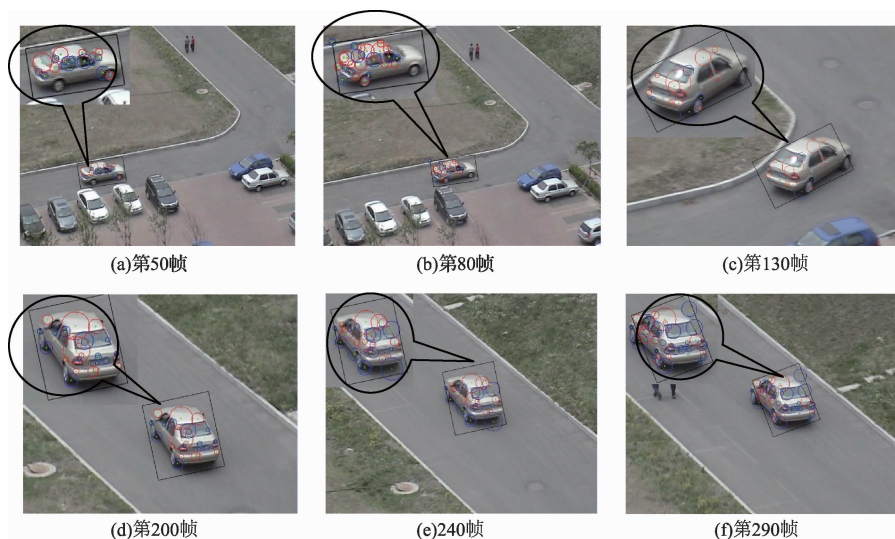


图3 汽车跟踪实验

为了进一步定量分析本文算法的跟踪性能, 如表4所示相应图像中目标跟踪框参数。从表中可以看出目标在整个运动过程中发生了较大的尺度缩放和旋转, 应用本文算法均得到了较好的跟踪效果。

表4 目标跟踪框参数

帧号	$s_x$	$s_y$	$\theta$	$t_x$	$t_y$
50	0.986	1.053	8.531	125	-19
80	0.959	0.997	0.4	125	-5
130	1.963	2.005	17.3	122	-14
200	0.735	1.089	76.2	-111	-57
240	1.010	0.932	-0.5	44	-35
290	0.929	0.950	8.1	22	3

## 7 结论

针对目标跟踪的鲁棒性问题进行研究, 提出了基于局部特征点匹配的 KPM 算法。首先介绍了应用 SURF 算法进行特征提取的步骤; 然后研究了特征点匹配, 并提出了

一种新的特征匹配提纯方法即一致性提纯算法, 降低了误匹配的发生概率, 有效提高了目标的跟踪精度; 最后, 利用匹配特征点对计算出目标旋转和缩放参数, 更新目标区域。实验结果证明应用本文提出的 KPM 算法进行目标跟踪, 当目标发生大角度旋转和缩放时仍能实现稳定跟踪, 具有较强的鲁棒性。

## 参考文献

- [1] 王晓东, 贾继鹏. 一种基于 IMPDA-UKF 的机动目标跟踪算法[J]. 电子测量技术, 2014, 37(10): 5-8.
- [2] 金汉均, 梅洪洋. Sobel 算子在提取视频运动目标轮廓中的应用研究[J]. 电子测量技术, 2014, 37(11): 29-32.
- [3] 吴乐, 凌晨, 周霁婷. 基于改进 Hausdorff 距离的运动目标自动跟踪算法[J]. 电子测量技术, 2014, 37(4): 56-60.
- [4] 韩军, 景彩云, 吴玲玲, 等. 基于 DSP 的图像处理在转角测试中的应用[J]. 国外电子测量技术, 2013, 32(1): 29-32.

(下转第 40 页)