

基于 KCF 跟踪算法的目标轨迹记录系统^①

张乘龙^{1,2}, 夏筱筠², 柏松³, 姚恺丰⁴

¹(中国科学院大学, 北京 100049)

²(中国科学院 沈阳计算技术研究所, 沈阳 110168)

³(中航工业沈阳黎明航空发动机(集团)有限责任公司, 沈阳 110043)

⁴(国家电网公司东北分部 国网东北电力调度中心, 沈阳 110180)

摘要: 为了确保跟踪算法能够实时跟踪上高速移动的目标并且记录目标的三维坐标. 本系统使用了一种基于 KCF(Kernelized Correlation Filters)的高速跟踪算法来保证系统能够跟踪到移动速度较快的目标. 首先, 使用 KCF 跟踪算法来跟踪目标; 然后, 利用 ORB 特征点检测来计算目标特征点从而找到多摄像机中对应的点, 找到对应点之后利用多摄像机的三维重建原理计算出每一帧中目标物体的三维坐标点; 最后, 用多项式对每一帧运动轨迹的离散点进行拟合得到最终的运行轨迹. 实验结果证明该算法能够有效跟踪目标, 整个系统能够满足实际的需求.

关键词: KCF 跟踪算法; ORB 特征检测; PROSAC 特征点匹配; 三维重建; 运动轨迹记录

Target Track Recording System Based on Kernelized Correlation Filters Tracking Algorithm

ZHANG Cheng-Long^{1,2}, XIA Xiao-Jun², BAI Song³, YAO Kai-Feng⁴

¹(University of Chinese Academy of Sciences, Beijing 100049, China)

²(Shenyang Institute of Computing Technology, Shenyang 110168, China)

³(AVIC Shenyang Liming Aero-engine(Group) Co. Ltd., Shenyang 110043, China)

⁴(North-East Branch of State Grid, Shenyang 110180, China)

Abstract: In order to ensure that our tracking algorithm can real-time capture the fast-moving target and record its three-dimensional coordinates, the system uses a high-speed tracking algorithm based on Kernelized Correlation Filters (KCF). First, use KCF tracking algorithm to track the target. Second, use ORB feature point detection algorithm to calculate the target feature point. Then find out the corresponding point in Multi-Camera. After finding the corresponding points, use three-dimensional reconstruction theory of Multi-Camera to calculate the three-dimensional coordinates of the target object in each frame. Finally, using polynomial to fit the discrete points of each frame and then get the final trajectory. The experimental results show that this algorithm can track target efficiently and the whole system can meet actual requirements.

Key words: KCF tracking; ORB feature descriptor; matching with PROSAC; three-dimensional reconstruction; target track recording

目标的跟踪是计算机视觉的一个重要研究领域. 随着科技的发展, 目标跟踪以及目标轨迹记录在交通监控、行人流量、天文观测、导航制导、武器装备的研发等领域有着很实用的价值. 针对目标跟踪, 国内外大量学者做了很多工作: 注重边跟踪边学习目标特性的 TLD 跟踪算法^[1]; 基于压缩感知的跟踪算法 CT

跟踪算法^[2]等. 这些算法几乎已经能够达到实时跟踪的目的. 不过在一些领域, 例如飞行器的研发领域, 或者对目标跟踪实时性要求较高的领域中. 由于目标速度较快, 或者实时性要求较高, 那些传统传统的跟踪方法不能够达到实时跟踪的目的. 针对以往跟踪系统不能跟踪快速移动的目标或者系统本身实时性不

① 收稿时间:2016-08-29;收到修改稿时间:2016-10-17 [doi:10.15888/j.cnki.csa.005780]

够好等问题本系统提出采用 KCF^[3]高速跟踪算法来跟踪目标, 最终达到记录目标轨迹的目的. KCF 跟踪算法是一种新型的高速跟踪算法. 通过构建目标与背景之间的分类器来判别目标, 是一种具有快速训练、快速检测等特性的高速算法. 因此, 在实时性要求较高或目标移动速度较快的跟踪算法应用中, 具有广阔的前景. 本文利用核相关滤波器计算量较小、实时性较好的特性将其应用在实时的目标移动跟踪里, 取得了比较理想的效果.

1 KCF跟踪算法

核相关滤波器跟踪算法的核心思想是将跟踪目标区域进行循环移位, 以此构造大量的样本来训练分类器. 通过分类器来计算候选区域与跟踪目标的相似程度, 选取相似度最大的候选区域为新的跟踪目标, 同时利用离散傅里叶变换降低分类器训练和检测过程中的运算量^[3].

1.1 HOG 特征

HOG(Histogram of Oriented Gradient, HOG)即梯度方向直方图特征, 是用于在计算机视觉和图像处理领域, 目标检测的特征描述子, 该项技术是计算图像局部出现的方向梯度作为图像特征^[4].

HOG 特征计算方法分成以下几步: 1)分割图像, 将目标区域划分成一定大小相连接的小区域, 每个小区域称为细胞(cell); 2)计算每个区块的方向梯度以及梯度的方向, 在图像中像素位置(x, y)的水平与垂直方向梯度定义为:

$$\begin{cases} G_x(x, y) = H(x+1, y) - H(x-1, y) \\ G_y(x, y) = H(x, y+1) - H(x, y-1) \end{cases} \quad (1)$$

像素位置(x, y)的梯度值跟梯度方向的定义为:

$$\begin{cases} G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \\ \alpha(x, y) = \tan^{-1}\left(\frac{G_y(x, y)}{G_x(x, y)}\right) \end{cases} \quad (2)$$

3)根据 cell(细胞)单元中的每一个像素点的梯度值跟梯度方向利用双线性内插法进行加权计算, 得出每个细胞中的梯度方向直方图; 4)把各个直方图组合起来组成特征向量.

在实际过程中由于局部光照的变化, 以及前景和背景对比度的变化, 使得梯度强度的变化范围非常大, 通常需要对梯度强度做归一化.

1.2 正则化最小二乘分类器

由于正则化最小二乘分类器模型具有实现简单训练速度较快等特性, 因此, 正则化最小二乘分类器经常使用在一些实际项目中. 在一些实际问题中此模型分类器效果可以跟 SVM 分类器相同^[5].

正则化最小二乘分类器训练的目标就是用过样本 x 训练得出一个 $f(x) = w^T x + w_0$, 使得正则化风险最小:

$$\min_w \sum_{i=1}^N (y_i - (w_i^T x + w_0))^2 + \lambda \|w\|^2 \quad (3)$$

其中的 λ 为控制过拟合的参数, 由于这个公式是封闭式的, 其解已经被文章^[5]给出:

$$w = (XX^T + \lambda I)^{-1} X^T y \quad (4)$$

1.3 循环矩阵

为了使用位移的样本来训练最小二乘分类器, 我们可以使用循环矩阵. 假设 $C(x)$ 是一个 $n \times n$ 的矩阵, 则它可以通过一个 $1 * n$ 的向量的循环移位获得, 则有:

$$X = C(x) = \begin{bmatrix} x_0 & x_2 & \cdots & x_n \\ x_n & x_1 & \cdots & x_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & \cdots & x_1 \end{bmatrix} \quad (5)$$

所有的循环矩阵都可以对角化, 并且可以由向量 x 的离散傅里叶变换(Discrete Fourier Transform, DFT)转换^[6], 其过程可以写成以下公式:

$$X = F \text{diag}(\hat{x}) F^H \quad (6)$$

其中 F 是不依赖于 x 的常数矩阵, \hat{x} 是向量 x 的离散傅里叶变换. 等式(6)表示了普通循环矩阵的特征分解, 其中的常数矩阵 F 代表离散傅里叶变换的常数矩阵, 并且通过这个矩阵我们能够计算任意输入的向量的离散傅里叶变换, 公式为:

$$F(x) = \sqrt{n} Fz \quad (7)$$

由于循环矩阵有这些性质^[6,7], 我们可以在核相关滤波器中使用它, 从而大大提高算法的速度.

1.4 利用循环矩阵的最小二乘分类器

在跟踪过程中利用循环矩阵的性质, 在对分类器进行训练时, KCF 算法利用目标的基本样本作为正样本, 对基本样本循环位移之后的样本为负样本. 只需对基样本进行计算, 速度比较快. 由等式(6)可以得出:

$$X^H X = F \text{diag}(\hat{x}^*) F^H F \text{diag}(\hat{x}) F^H \quad (8)$$

由于对角矩阵是对称的, 对其进行埃尔米特转换就会剩下复共轭 \hat{x}^* . 另外由于 F 的特性, 我们可以得

到 $F^H F = I$, 这样我们就可以把公式(8)转换为:

$$X^H X = F \text{diag}(\hat{x}^*) \text{diag}(\hat{x}) F^H \quad (9)$$

由于操作在对角矩阵上, 公式(8)可以写成:

$$X^H X = F \text{diag}(\hat{x}^* \cdot \hat{x}) F^H \quad (10)$$

将公式(10)带入公式(4)我们可以得出线性回归的权值 w 的变换形式^[3]:

$$\hat{w} = \frac{\hat{x}^* \cdot \hat{y}}{\hat{x}^* \cdot \hat{y} + \lambda} \quad (11)$$

这样我们可以将最小二乘分类器的训练时间复杂度从原本的矩阵求逆运算转换为矩阵的相对元素相乘与离散傅里叶变换^[3].

1.5 非线性回归的核相关滤波器

我们可以采用核函数将输入的向量 x 映射到特征空间 $\varphi(x)$ 中, 则可以把公式(3)的解表示为输入的线性组合^[8], 系数为 α :

$$w = \sum_i \alpha_i \varphi(x_i) \quad (12)$$

通过文献[5]我们可以得到公式(12)中的 α 变为:

$$\alpha = (K + \lambda I)^{-1} y \quad (13)$$

其中 K 为核矩阵, 元素 $K_{ij} = \kappa(x_i, x_j)$, I 为单位矩阵. 使用核函数的分类器将样本特征映射到高维特征空间, 使得分类更加精确. 但是这样也会使大量样本训练的时间增加. 所以我们通过循环矩阵构建训练样本, 减少计算. 通过文献[9]结合 1.3 节中循环矩阵的性质经推导可得:

$$\hat{\alpha} = \frac{\hat{y}}{\hat{k}_{ij} + \lambda} \quad (14)$$

其中 K_{ij} 是核矩阵 K 的第一行, \hat{k}_{ij} 表示 K_{ij} 的离散傅里叶变换^[3].

1.6 核相关滤波器的响应

当我们训练好分类器之后, 将新的一帧里的图像特征输入分类器, 来判断目标位置. 这样, 分类器的相应 $y = w^T z$ 结合之前的公式, 我们可以得到:

$$y = F^{-1}(\hat{k}_{xz} \cdot \hat{\alpha}) \quad (15)$$

其中, $\hat{k}_{xz} = \kappa(x, z_i)$ 中, z_i 为新一帧中获取的训练样本, x 为上一帧学习获得的目标模型, 使 y 为最大的位置就是新一帧中目标的位置.

2 特征提取与匹配算法

2.1 ORB 特征提取

OBR(Oriented FAST and Rotated BRIEF)特征提取算法是近年提出的一种改进的新型算法. 其特点是使

用 o-FAST 特征点提取算法和 rBRIEF 特征值描述子. ORB 算法运用灰度质心法在 FAST 特征点检测的基础上加上了方向检测, 又通过在 BRIEF 描述子的点对集矩阵上加入旋转矩阵变为 rBRIEF 描述子, 使得 ORB 算法加上了旋转不变性, 使算法具备了平移、旋转、不变性. 该算法的特征提取速度相对于 SIFT^[10]等算法有着一个数量级以上的提高^[11].

2.2.1 O-FAST 特征点检测

O-FAST 相对 FAST 的主要改进之处就在于加入了方向特性, 这一特性会让此特征描述子具备旋转不变性. 使用的是 Rosin 的灰度质心法^[12](intensity centroid). 首先对一个图像区域定义一个矩:

$$m_{pq} = \sum x^p y^q I(x, y) \quad (16)$$

通过这个我们可以找到图像区域的质心.

$$C = \begin{pmatrix} m_{10} & m_{01} \\ m_{00} & m_{00} \end{pmatrix} \quad (17)$$

在 ORB 中, 这个图像区域为 FAST 关键点的邻域, 即以关键点 O 为圆心, 以 3 个像素为半径的圆形区域. 即 x, y 的取值范围是 $[3, 3]$, 且在该圆形区域内. 这样就可以得到一个向量 \overline{OC} , 用向量 \overline{OC} 的方向 θ 表示 FAST 的关键的方向:

$$\theta = \arctan 2(m_{01}, m_{10}) \quad (18)$$

2.2.2 rBRIEF 特征描述

ORB 算子在特征描述部分采用的是基于 BRIEF 算子的改进算法. BRIEF^[13]特征描述子描述简单、占用存储空间小、速度快, 但缺陷也很明显: 不具备旋转不变性, ORB 中的 rBRIEF 特征描述子正是解决了这一点.

在进行特征描述前用积分图像法对图像进行平滑处理. 为了给 BRIEF 加入方向信息, 首先需要将点的集合改写为矩阵形式. 对于一个 n 比特的点的集合 (x_i, y_i) , 定义一个矩阵 S :

$$S = \begin{pmatrix} x_1 & x_2 & \dots & x_n \\ y_1 & y_2 & \dots & y_n \end{pmatrix} \quad (19)$$

根据 FAST 特征点提取算法计算出来的方向角 θ , 求出这个点它对应的旋转矩阵 R_θ , 这样就得到了带有方向特性的点的集合: $S_\theta = R_\theta S$. 这种新型的描述子具有更大的方差, 描述特征的能力更强.

2.2 PROSAC 特征点匹配

PROSAC(Progressive Sample Consensus)^[14]算法是一种全局匹配的算法, 用来匹配不同图片的对应点. 对于 N 个采样点的集合, 可以表示为 U_N , u_i, u_j 是 U_N

中两采样点. 采样点依照他们的相关性降序排列, 其中相关性函数为 $q(u)$:

$$i < j \Rightarrow q(u_i) < q(u_j) \quad (20)$$

算法的采样策略为现存概率更高的点被再次采样的概率更大. 设 u_i 为正确的数据点, $p\{u_i\}$ 代表采样到该点的概率, 可以有如下推断:

$$i < j \Rightarrow P\{u_i\} > P\{u_j\} \quad (21)$$

记 U_n 为有 n 个高质量内点的集合, 记 T_n 为 $\{M_i\}_{i=1}^T$ 中包含 U_n 中的高质量点的平均数则:

$$T_n = T_N \frac{\binom{n}{m}}{\binom{N}{m}} = T_N \prod_{i=0}^{m-1} \frac{n-i}{N-i} \quad (22)$$

这样我们可以得到 T_n 的递推关系式:

$$\frac{T_{n+1}}{T_n} = \frac{n+1}{n+1-m} \quad (23)$$

如果集合中内点数量符合以下两个条件时, 算法退出循环迭代: (1)若某次抽样得到的内点数量超过 95%时; (2)经过 k 次抽样内点数量增长小于 5%时. 在本系统中 $k=5$. 这样递推一直到退出条件为止, T_n 这个集合就只包含 U_n 中高质量的内点.

3 三维坐标记录

3.1 双目视觉测量模型

双目视觉测量模型是两个相机在同一时刻进行拍摄, 这两个相机获得的相同目标的不同角度的成像. 然后通过这种相机成像的区别, 利用双目视觉中的视差原理, 计算该目标的三维空间坐标. 如图 1 所示.

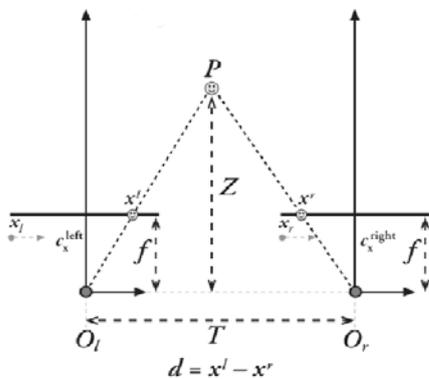


图 1 双目视觉测量视差原理

双目视觉视差深度信息计算公式:

$$\frac{T - (x' - x'')}{Z - f} = \frac{T}{Z} \Rightarrow Z = \frac{fT}{x' - x''} \quad (23)$$

通常情况下双目视觉测量系统可以分为两种形式: 一种是两个相机的光轴平行状态下的比较理想的理想模型; 另外一种则是两个相机光轴不平行放置的测量模型. 大多数情况下, 光轴绝对平行是无法做到的. 所以大多数双目视觉测量的应用中, 用的是不平行放置的模型, 而是利用了相机随意摆放的模型, 例如图 2 所示.

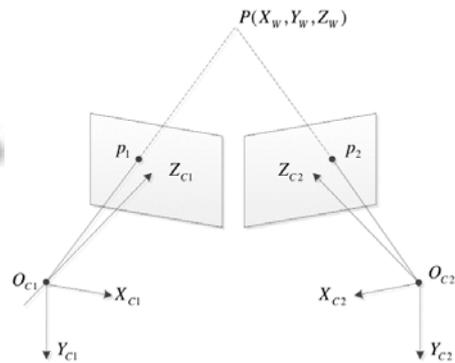


图 2 双目视觉系统

$P(X_w, Y_w, Z_w)$ 点在两个相机的图像坐标系(像素单位)上的投影点为 $P_1(u_1, v_1)$ 、 $P_2(u_2, v_2)$, 假设相机 C_1 和 C_2 已经标定过, M_1 、 M_2 分别为左右相机的投影矩阵, 则根据三维坐标到成像像素坐标的转换有矩阵形式^[15]:

$$r_1 \begin{bmatrix} u_1 \\ v_1 \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11}^1 & m_{12}^1 & m_{13}^1 & m_{14}^1 \\ m_{21}^1 & m_{22}^1 & m_{23}^1 & m_{24}^1 \\ m_{31}^1 & m_{32}^1 & m_{33}^1 & m_{34}^1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (24)$$

其中矩阵 M 为相机标定后的已知数, 这样我们联立公式 (25) 的 r_1 跟 r_2 , 计算出三维空间中的点 $P(X_w, Y_w, Z_w)$ 跟两张图片中投影的点 $P_1(u_1, v_1)$ 、 $P_2(u_2, v_2)$ 的关系:

$$\begin{bmatrix} m_{31}^1 u_1 - m_{11}^1 & m_{32}^1 u_1 - m_{12}^1 & m_{33}^1 u_1 - m_{13}^1 \\ m_{31}^1 v_1 - m_{21}^1 & m_{32}^1 v_1 - m_{22}^1 & m_{33}^1 v_1 - m_{23}^1 \\ m_{31}^2 u_2 - m_{11}^2 & m_{32}^2 u_2 - m_{12}^2 & m_{33}^2 u_2 - m_{13}^2 \\ m_{31}^2 v_2 - m_{21}^2 & m_{32}^2 v_2 - m_{22}^2 & m_{33}^2 v_2 - m_{23}^2 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} = \begin{bmatrix} m_{14}^1 - m_{34}^1 u_1 \\ m_{24}^1 - m_{34}^1 v_1 \\ m_{14}^2 - m_{34}^2 u_2 \\ m_{24}^2 - m_{34}^2 v_2 \end{bmatrix} \quad (25)$$

简化之后为:

$$M_s X = N \quad (26)$$

因为两张图片对应两点在跟其相机焦距的延长线

在空间交于同一点 P , 所以 M_s 是满秩矩阵, 公式(26) 的方程有解. 从而得到 P 的世界坐标.

4 算法实例与结果

程序使用 VS2013 跟 OPENCV2.4.9 开源库编写, 背景复杂而且有扰动. 系统首先由使用了高斯背景模型来检测前景物体. 当物体出现在摄像头中以后, 对目标进行特征的提取. 根据目标的 HOG 特征, 使用 KCF 跟踪算法跟踪进入双目摄像机视野范围的物体. 使用 HOG 特征的 KCF 算法是在 CSK(Circulant Structure with Kernels)算法上扩展出来的. 使用了 HOG 特征并且将 CSK 算法扩展到多通道并行. 这样使得 KCF 算法计算速变快, 适合跟踪快速目标, 其中分类器核函数使用了作者默认的高斯核函数. 表 1 为系统中使用 KCF 跟踪算法平均耗时与其他跟踪算法的比较. 数据表明 KCF 跟踪算法实时性较好能够满足我们的实际需求.

表 1 系统中跟踪算法在速度上的比较

算法	平均帧率	平均耗时
TLD	15.6	64.1ms
CT	34.3	29.1ms
KCF	82.7	12.1ms

由于 KCF 跟踪算法使用了 HOG 特征, 在一些时候 HOG 的块划分会使得跟踪效果有一点不够准确. 为了提高跟踪效果的精确度, 系统使用前 5 帧的位置以及速度跟梯度等信息来修正新一帧中 KCF 算法得到的目标位置, 使得跟踪结果更加精确.

识别并跟踪目标之后使用 ORB 特征提取提取出目标特征点, 使用 PROSAC 来计算对应的匹配点. 并且进行初步的数据处理. SIFT 等算法在尺度不变以及旋转的情况下有很好的效果, 但是由于系统使用双目摄像头, 其成像是平行的而且距离相似目标尺度变化较小. ORB 特征提取算法不但能够满足我们系统的需求, 而且速度很快, 更适合系统使用. 表 2 为系统中使用 ORB 特征提取以及 PROSAC 特征匹配花费的平均时间跟其他算法的比较. 数据表明 ORB+PROSAC 花费的时间最少, 而且效果能够满足我们的需求.

表 2 系统中跟踪算法在速度上的比较

算法	检测时间	匹配时间	总时间
ORB+RANSAC	19.2ms	18.4ms	37.6ms
ORB+PROSAC	19.2ms	6.7ms	25.9ms
SURF+RANSAC	53.3ms	24.5ms	77.8ms

SURF+PROSAC	53.3ms	7.2ms	60.5ms
SIFT+RANSAC	93.7ms	23.2ms	116.9ms
SIFT+RANSAC	93.7ms	7.1ms	100.8ms

然后根据双目视觉系统的三维重建模型利用匹配过的应点的坐标. 得到每帧中离散的三维坐标之后, 系统需要对这些离散的三维坐标进行处理, 去掉每一帧中对应点的离群数据. 最后针对离散的坐标数据进行模糊辨识^[16]. 得到目标距离轨迹的平滑运动曲线. 整个系统的处理流程如图 3 所示.

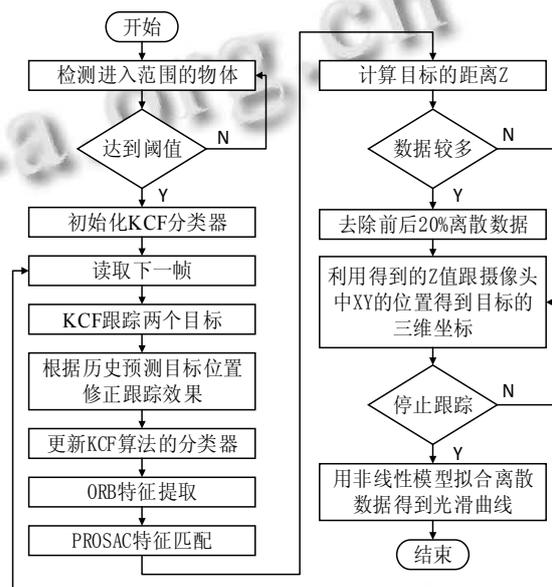


图 3 系统的处理流程

实验结果为跟中系统计观测后计算出来的目标运行轨迹. 坐标系以双目摄像头为原点, 三个坐标轴分别为 X, Y, Z 轴. 单位为 mm. 蓝色为观测数据, 红色为拟合数据. 系统跟踪记录的目标运行轨迹结果为图 4 所示.

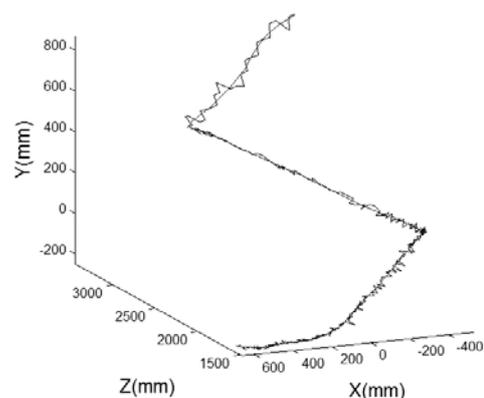


图 4 系统测量的目标运行轨迹

5 结语

本文以为了满足跟踪并记录快速移动目标的轨迹,采用KCF高速跟踪算法对目标进行跟踪;ORB特征检测用来计算对应点的三维坐标,最后使用模糊辨识将离散的三维坐标进行拟合得到了较好目标运行轨迹.实验结果表明,该方法能够在满足速度较快的目标运动条件的跟踪,并由分析录像够得到较理想的运动轨迹,说明此方法有效可行并能满足实际要求的需要.

参考文献

- 1 Kalal Z, Mikolajczyk K, Matas J. Tracking learning detection. *IEEE Trans. on Pattern Analysis & Machine Intelligence*, 2012, 34(7): 1409–1422.
- 2 Zhang K, Zhang L, Yang MH. Real-time compressive tracking. *European Conference on Computer Vision*. Springer-Verlag. 2012. 864–877.
- 3 Henriques JF, Rui C, Martins P, et al. High-speed tracking with kernelized correlation filters. *IEEE Trans. on Pattern Analysis & Machine Intelligence*, 2015, 37(3): 583–596.
- 4 Dalal N, Triggs B. Histograms of oriented gradients for human detection. *IEEE Conference on Computer Vision & Pattern Recognition*. 2013. 886–893.
- 5 Rifkin R, Yeo G, Poggio T. Regularized least-squares classification. *Acta Electronica Sinica*, 2003, (190): 93–104.
- 6 Gray RM. Toeplitz And Circulant matrices: A review (foundations and trends(R) in communications and information theory). Now Publishers Inc, 2006.
- 7 Davis PJ. *Circulant Matrices: Second Edition*. 1994.
- 8 Schölkopf B, Smola A. Learning with Kernels: Support vector machines, regularization, optimization, and beyond. *Journal of the American Statistical Association*, 2011, 16(3): 781.
- 9 Henriques JF, Rui C, Martins P, et al. Exploiting the Circulant Structure of Tracking-by-Detection with Kernels. *European Conference on Computer Vision*. 2012. 702–715.
- 10 Lowe DG. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 2004, 60(2): 91–110.
- 11 Rublee E, Rabaud V, Konolige K, et al. ORB: An efficient alternative to SIFT or SURF. 2011 *IEEE International Conference on Computer Vision (ICCV)*. 2011. 2564–2571.
- 12 Rosin PL. Measuring corner properties. *Computer Vision and Image Understanding*, 1999, 73(2): 291–307.
- 13 Calonder M, Lepetit V, Strecha C, et al. BRIEF: Binary robust independent elementary features. *European Conference on Computer Vision*. 2010. 778–792.
- 14 Chum O, Matas J. Matching with PROSAC – Progressive sample consensus. *IEEE Computer Society Conference on Computer Vision And Pattern Recognition*. 2005,1. 220–226.
- 15 邱茂林,马颂德,李毅.计算机视觉中摄像机定标综述. *自动化学报*,2000,26(1):43–55.
- 16 王辉,肖建,严殊.关于模糊系统辨识近年来的研究与发展. *信息与控制*,2004,33(4):445–450.