

# 基于 DSP 的嵌入式目标跟踪系统<sup>①</sup>



时旭东, 施华君, 陆国强

(中国电子科技集团公司第三十二研究所, 上海 201808)

通讯作者: 时旭东, E-mail: [Creanut@mail.ustc.edu.cn](mailto:Creanut@mail.ustc.edu.cn)

**摘要:** 近年, 由 Henriques 等人提出的核化相关滤波算法 (KCF 算法) 在算法规模、复杂度、性能等方面表现优越. 本文以 KCF 算法为核心, 提出并设计了一种基于 DSP 的目标跟踪系统. 硬件方面, 本文设计实现了一套完整独立的硬件平台; 软件方面, 本文提出一系列针对 DSP 的算法优化方法, 使优化后的 KCF 算法能够满足重要的工程指标要求. 结果表明, 系统在工程环境中表现良好, 跟踪角速度可达 20 度/秒, 平均帧率 25 fps, 跟踪准确率较高, 为计算机视觉领域内的各类算法的嵌入式应用提供参考.

**关键词:** KCF 算法; 目标跟踪系统; 数字信号处理器; 算法优化; 工程环境; 嵌入式应用

引用格式: 时旭东, 施华君, 陆国强. 基于 DSP 的嵌入式目标跟踪系统. 计算机系统应用, 2019, 28(11): 87-95. <http://www.c-s-a.org.cn/1003-3254/7142.html>

## Embedded Target Tracking System Based on DSP

SHI Xu-Dong, SHI Hua-Jun, LU Guo-Qiang

(The 32nd Research Institute of China Electronics Technology Group Corporation, Shanghai 201808, China)

**Abstract:** In recent years, the Kernel Correlation Filtering algorithm (KCF algorithm) proposed by Henriques et al. shows superior performance in terms of algorithm scale, computational complexity, and algorithm performance. Based on KCF algorithm, a target tracking system based on DSP is proposed and designed in this study. In terms of hardware, this study designs and implements a complete and independent hardware platform. In terms of software, this study proposes a series of algorithm optimization methods for DSP to optimize KCF algorithm, in order to meet the requirements of important engineering indicators. The results show that the system performs well in the actual engineering environment, the highest tracking angular velocity can be 20 degrees/s, and the frame rate can be 25 fps on average, and it has high accuracy. The system provides reference for embedded applications of various algorithms in the field of computer vision.

**Key words:** KCF; target tracking system; DSP; algorithm optimization; engineering environment; embedded application

目标跟踪是计算机视觉领域中一个非常重要的研究方向, 其主要工作是通过对图像序列中的指定目标进行特征提取与建模, 而后利用所得模型获取或标注出指定目标在动态图像序列中的位置信息, 从而实现目标跟踪. 近年来, 随着计算机视觉领域的快速发展, 目标跟踪算法和理论成果也层出不穷. 然而, 这些算法虽然种类繁多、效果显著, 但其大多是基于 PC 平台进

行算法实现的, 这些算法的开发普遍需要借助第三方软件或者软件库进行, 很难直接在嵌入式平台上进行算法部署和应用; 此外, 工程环境中获取的图像包含很多不确定因素, 这些因素也会极大影响算法的跟踪性能. 这些缺陷使得大量的目标跟踪算法在实际的国防军事、公共安全、工业生产等众多工程领域中的应用大大受限, 实际效果也大大折扣. 因此, 目标跟踪算法

① 收稿时间: 2019-04-03; 修改时间: 2019-05-08; 采用时间: 2019-05-13; csa 在线出版时间: 2019-11-06

在嵌入式平台上的实现一直是该领域中需要着力攻关的难点,设计一种能够适应现实环境复杂性的、具有一定实时性的、性能具有鲁棒性的目标跟踪系统更是目前亟待解决的课题<sup>[1,2]</sup>。

DSP (数字信号处理器) 作为一种具有强大运算处理能力的嵌入式系统,具备能耗低、规模小、成本低、稳定性高等诸多优点,这使其在实际工程项目中应用广泛,现已成为目标跟踪算法技术应用的关键平台之一。因此,研究基于 DSP 的目标跟踪系统设计及实现方法是十分必要并且具有极高的现实意义的。近年,随着机器学习的迅猛发展,目标跟踪算法已从图像帧差分法<sup>[3]</sup>、图像帧间特征匹配法等传统方法过渡到基于机器学习的目标跟踪算法,在实际应用中,这对算法的嵌入式平台优化与应用提出了更高的要求。

本文介绍了一种基于 TMS320C6414 平台的目标跟踪系统设计及其实现方法,该系统选用了基于机器学习的核化相关滤波算法<sup>[4-8]</sup>(KCF 算法)作为目标跟踪模块的核心算法,在简要介绍系统硬件平台的基础上,详细分析了 KCF 算法的原理和特点,与此同时,针

对本系统的硬件平台,文章着重解读了系统在算法架构、数据处理、程序编译等方面的优化思路和方法,最后展示了本系统在实际工程应用中的实验结果,并得出相关结论,为计算机视觉领域内的各类算法的嵌入式应用提供参考。

## 1 系统硬件设计

作为一个能够独立运行于实际工程环境中的目标跟踪系统,其主要功能包括:图像采集和数据缓存、实现目标跟踪算法、上位机通信与跟踪结果反馈等功能。由于系统的外围通信接口较多,通信任务繁重,同时又要运行目标跟踪算法,运算任务繁重,所以在系统的硬件设计中需要根据不同任务,采用模块化的设计思想。FPGA 作为一种典型的半定制电路,拥有丰富的可并行执行的逻辑运算单元,非常适合进行信号调理和时序控制,因此在系统硬件设计中,本文选择 FPGA 作为协处理器进行数据通信与外围芯片的时序控制,DSP 则作为核心处理器,主要运行需要大量复杂运算的目标跟踪算法。系统硬件平台的基本结构如图 1 所示。

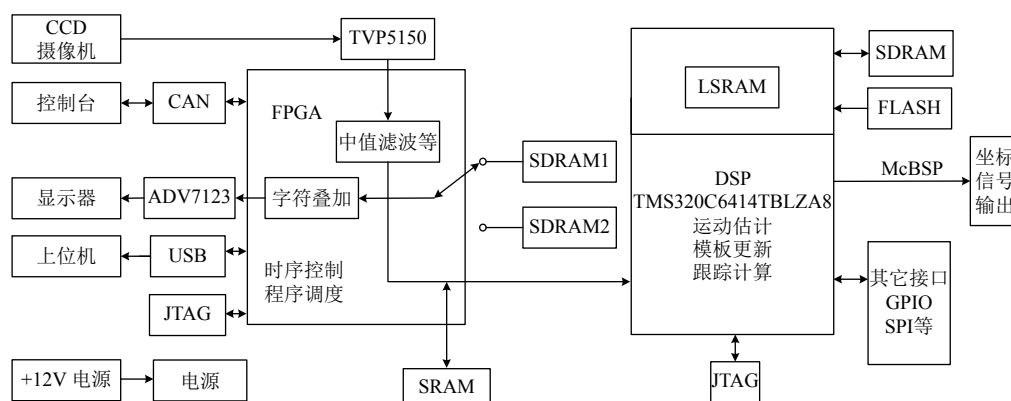


图 1 系统硬件结构

从图中可以看出,硬件平台包括视频采集电路、视频解码电路、视频显示电路、数据缓存电路、视频处理电路、通信电路以及电源电路等,其中视频解码电路、数据缓存电路以及视频处理电路是本系统硬件平台的核心电路。

### 1.1 视频解码电路

视频解码电路由 TVP5150 及其外围电路组成,该电路功能是将 CCD 摄像机采集的 PAL 模拟信号转换为符合 BT.656 标准的 8 位 4:2:2 YCbCr 格式的数字信号<sup>[9]</sup>。本系统中,由协处理器 FPGA 根据 BT.656 帧划分

和数据流格式解析出行同步信号、场信号以及 YCbCr 数据信号,并将解析结果输出至后续电路进行处理。

### 1.2 数据缓存电路

数据缓存电路主要负责图像数据的缓存工作。考虑到 DSP 读取和处理数据需要较快的速度,因此使用工作频率较高的 SRAM 作为 DSP 的数据缓存,而协处理器 FPGA 的数据缓存则采用经典的“乒乓结构缓冲存储”<sup>[10,11]</sup>策略,即采用两片 SDRAM 作为图像数据缓存主体,输入输出数据选择单元切换由 FPGA 控制实

现, 最终完成图像数据的无缝缓冲处理, 提升系统的数据存取速度.

### 1.3 视频处理电路

视频处理电路由协处理器 FPGA、核心处理器 DSP 以及它们的外围电路组成. 为达到系统“小规模、低成本、高效率”的设计需求, FPGA 选用 Altera 公司 Cyclone IV 系列的 EP4CE40F29I7N 芯片, 该产品具备 532 个可用 I/O, 39 600 个基本逻辑单元 (LEs), 1134 Kbit 片上存储空间, 芯片面积仅为  $29 \times 29 \text{ mm}^2$ , 其资源足够满足系统功能需求; DSP 则选用 TI 公司 TMS320C6000 系列最高性能的定点 DSP 产品 TMS320C6414, 该产品具有 2 个 FMIF 接口, 工作频率最高可达 850 MHz, 其运算能力足够满足系统的算法需求.

## 2 核化相关滤波算法 (KCF 算法)

目标跟踪算法在本系统中嵌入在 DSP 的监控程序中, 算法的具体要求是, 在视频中的每一帧图像中均能识别并框选出所指定的跟踪目标, 并将跟踪结果反馈至控制台上位机以供观察. 针对本系统的研制需求, 本文提出了一种核化相关滤波算法 (KCF 算法) 在本系统硬件平台上的优化与实现方法.

KCF 算法是一种典型的判别式目标跟踪算法<sup>[5,6]</sup>, 该算法的核心思想是根据图像中给定的跟踪目标, 在线学习得到一个能够区分目标和背景信息的目标检测器模型, 然后使用该检测器模型在下一帧图像上进行相关滤波计算, 从而得到图像的输出响应并框选出目标区域, 并用得到的目标区域更新目标模板. 算法的优势在于, 在目标检测器模型的训练过程中引入“核技巧”, 该方法可以隐式地使用图像的高维特征, 增强模型的泛化能力和目标识别的准确率<sup>[12]</sup>; 与此同时, 在算法实现过程中巧妙地使用循环矩阵来拓展训练集, 在一定程度上解决了目标跟踪中训练样本匮乏的问题<sup>[13]</sup>, 并且算法利用循环矩阵在傅里叶域内可对角化的性质, 将原本时域中复杂的矩阵求逆运算转化为傅里叶域中的向量 Hadamard 积运算, 大大降低了算法运行的时间复杂度, 极大提高了运算速度<sup>[5,12]</sup>.

### 2.1 循环移位

在样本训练过程中, 将图像中选定的目标区域  $I$  作为基样本, 记其中的一行  $1 \times n$  维向量为  $x$ . 使用置换矩阵  $P$  对该基样本中的行向量  $x$  进行循环移位, 从而得到移位集合, 集合中的每个元素都可以看作是一

个向量的变换, 集合则可以看作是一个矩阵的变换. 对于行向量  $x$ , 其置换矩阵可以表示为:

$$P = \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} \quad (1)$$

置换矩阵与行向量相乘可构成循环矩阵  $X$ :

$$X = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ x_n & x_1 & \cdots & x_{n-1} \\ x_{n-1} & x_n & \cdots & x_1 \\ \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & \cdots & x_1 \end{pmatrix} \quad (2)$$

循环矩阵可通过离散傅里叶变换实现矩阵的对角化:

$$X = G \text{diag}(\hat{x}) G^H \quad (3)$$

上式可以看作是循环矩阵的特征值分解<sup>[12]</sup>, 其中,  $G$  被称作 DFT 矩阵, 是一个与基样本行向量无关的常量矩阵,  $G^H$  表示  $G$  的共轭转置,  $\hat{x}$  表示  $x$  的离散傅里叶变换. 对于二维图像, 可以对图像中的每一个行向量或者列向量进行上述操作, 从而获得循环矩阵. 由于离散傅里叶变换是一种线性变换, 对角化过程等价于对向量做元素级的线性变换, 所以, 其可大大简化后续计算, 提升算法性能.

### 2.2 训练目标检测器

训练目标检测器的本质是一个岭回归问题, 即寻找一个线性函数  $f(z) = w^T z$ , 使得该函数的输出与样本标签值之间的残差最小, 该过程可以表示为:

$$\min_w \sum (f(x_i) - y_i)^2 + \lambda \|w\|^2 \quad (4)$$

上式最优闭式解的复数形式为:

$$w = (X^H X + \lambda I)^{-1} X^H y \quad (5)$$

其中,  $X^H X = G \text{diag}(\hat{x}^* \circ \hat{x}) G^H$ ,  $\circ$  表示 Hadamard 积, 即矩阵的对应元素相乘. 将结果代入式 (5) 可得<sup>[5]</sup>:

$$\hat{w} = \frac{\hat{x}^* \circ \hat{y}}{\hat{x}^* \circ \hat{x} + \lambda} \quad (6)$$

对其进行傅里叶逆变换即可得到  $w$ . 可以看出, 原本  $O(n^3)$  复杂度的矩阵求逆运算被转换成了  $O(n)$  复杂度的元素乘法和  $O(n \log n)$  的傅里叶变换, 计算复杂度减少了几个数量级<sup>[5]</sup>.

### 2.3 核技巧

2.2 节中训练得到的目标检测器是线性的, 为得到具有更优性能的非线性目标检测器, KCF 算法引入“核技巧”来解决非线性目标检测器的在线训练问题. 即首先将问题的输入  $X$  映射到一个非线性空间  $\varphi(X)$  中, 将  $\varphi(X)$  作为岭回归问题的输入, 由算法文献[5]可得, 该问题的最优解为样本的线性组合:

$$w = \sum \alpha_i \varphi(x_i) \quad (7)$$

然后, KCF 算法引入核函数来表示高维空间中映射值的点积, 即:

$$\varphi^T(x) \varphi(x') = \kappa(x, x') \quad (8)$$

其中,  $\kappa$  为核函数, 其计算结果通常存储在核矩阵  $K$  中, 从式 (8) 可以看出, 核函数可以在不实例化任何向量的前提下隐式地使用向量的高维特征<sup>[12]</sup>. 根据算法文献[5]可以得到, 核函数下岭回归的闭式解为:

$$\alpha = (K + \lambda I)^{-1} y \quad (9)$$

文献[5]中证明大部分常用的核函数构成的核矩阵  $K$  是循环矩阵, 利用循环矩阵的性质可以对 (9) 式进行对角化, 结果为:

$$\hat{\alpha} = \frac{\hat{y}}{k^{xx'} + \lambda} \quad (10)$$

其中,  $k^{xx'}$  为核矩阵  $K$  的第一行元素. 从上述推导可知, 该求解过程可以避开式 (9) 中原本复杂的矩阵求逆运算, 将式 (10) 的计算结果进行离散傅里叶逆变换即可得到目标向量  $\alpha$ . 值得注意的是, 当样本规模线性增加时, 式 (9) 求解的复杂度呈指数级提升, 而式 (10) 则是线性级的<sup>[12]</sup>.

### 2.4 快速目标检测

对于输入的待检测图像块  $z$ , 其在训练得到的目标检测器中的输出响应为:

$$f(z) = w^T z = \sum \alpha_i \kappa(x_i, z) \quad (11)$$

其中,  $x$  为目标检测器的训练基本样本, 设  $K^z$  为训练样本和所有候选图像块的核矩阵, 其可表示为:

$$K^z = C(k^{xz}) \quad (12)$$

代入式 (11) 并对角化可得:

$$\widehat{f(z)} = \widehat{k^{xz}} \odot \hat{\alpha} \quad (13)$$

对上式结果进行离散傅里叶逆变换即可得到所有候选图像块的响应值.

以上过程的实质就是计算当前帧中的待检测图像块和训练检测器模型时所使用的目标模板之间的相关性, 并找出计算结果中相关性最高的区域作为目标位置. 结合本工程实际情况, 本文选用高斯核进行算法实现, 高斯核的核相关运算表达式为:

$$\widehat{k^{xz}} = \exp\left(-\frac{1}{\sigma^2} (\|x\|^2 + \|z\|^2 - 2F^{-1}(\widehat{x}^* \odot \widehat{z}))\right) \quad (14)$$

将其代入式 (13) 中即可得到最终结果.

### 2.5 模板更新

在实际的工程应用中, 引入超参数  $\eta$  作为更新因子用来调节目标检测器的更新率<sup>[12,14,15]</sup>, 以保证目标检测器模型的跟踪准确率, 并尽可能减弱噪声、干扰以及目标形变等对检测器性能的影响.

$$\begin{cases} x_t = \eta x_{t-1} + (1 - \eta) x_t \\ \alpha_t = \eta \alpha_{t-1} + (1 - \eta) \alpha_t \end{cases} \quad (15)$$

对于模板更新因子  $\eta$ , 当跟踪目标被遮挡或短暂消失时, 为保证系统的鲁棒性, 应停止对目标检测器的模板更新, 此时更新因子取值为 1; 当目标能够被正常检测时, 为保证目标跟踪的平滑性, 更新因子  $\eta$  应根据实际工程的调试情况, 取一个大于 0 的较小值<sup>[15]</sup>, 以保证目标跟踪的连贯性和平滑性.

## 3 基于硬件平台的算法优化与实现

KCF 算法的作者 Henriques 在其个人主页上公开了算法基于 OpenCV 开发的 C++ 源码以及 Matlab 源码, 相关资料<sup>[5,16]</sup>也给出了该源码在标准测试集上的性能指标. 但是由于本系统硬件不支持 OpenCV 的开发环境, 即源码无法在 DSP 上运行, 且当算法在实际工程环境中所采集的视频数据作为测试数据时, 原算法提供的评测指标也不再适用, 因此, 为满足系统在实际工程中的指标需求, 使其能够适应实际工程环境, 本算法需要基于系统的硬件平台进行多方面的优化和改进.

在本系统中, 算法的优化和改进主要面临以下问题: 其一, 在原代码中使用的大量 OpenCV 库函数无法在 DSP 上直接运行; 其二, KCF 算法中涉及大量的浮点运算, 而本系统中所使用的型号为 TMS320C6414 的 DSP 为定点 DSP, 浮点运算非常慢; 其三, 此算法中的二维傅里叶变换、图像插值、矩阵变换等函数运算量大, 未优化前, 这些函数在本硬件平台上运行时无法满足实时性需求. 鉴于以上问题, 本文针对 TMS320C6414

进行了以下优化<sup>[14,15,17-21]</sup>:

(1) 对原有的 OpenCV 函数进行了重构, 包括图像插值、矩阵变换、HOG 特征提取、矩阵运算、二维傅里叶变换等经典算法模块, 在保留功能的基础上进行适当裁剪与优化, 使其能够适配 DSP 的硬件底层。

(2) 优化算法的内存分配机制, 采用静态内存分配的策略, 消除 DSP 在算法不断申请和释放内存过程中的隐患。

(3) 为提高算法的运行效率, 将算法中的高运算量函数通过 C64X DSP 库函数进行实现, 使得计算过程与底层硬件严格适配。在本系统的算法实现中使用了 DSPLib、ImgLib、FastRTS 3 个由 TI 公司提供的官方库, 主要负责算法中高运算量步骤的计算, 提升计算效率。

(4) 针对算法中大量的浮点运算, 在保证计算精度指标的前提下采用浮点运算转定点运算的策略。

(5) 开通软件流水线操作, 将多层循环转化为多次迭代并行执行。

(6) 将算法中大量的数据存取和常用的数学运算等操作使用 intrinsic 函数进行优化<sup>[22]</sup>, 提升算法的运行速度。

在本系统实现过程中, 由于算法基于硬件平台的适配性优化较多, 在此仅攫取这些适配性优化中的一例以说明优化思路和方法。本算法中, 点积运算的使用频率较高, 对点积函数的优化会极大影响系统的工作效率和性能。首先, 可对代码进行编译器优化, C6000 的 C 优化器是 TI 的一个创新, 它对程序循环的流水处理使得 C 代码的效率得到很大提高, 使用时只需在编译时加入优化选项即可, 在使用中, 具体的优化编译命令可参考 TI 公司提供的使用指南<sup>[22,23]</sup>。采用优化选项进行优化后, 点积函数的执行时间从 1800 周期缩短至 159 周期, 执行效率提升超过 10 倍。除此之外, 本算法在运算过程中, 点积运算所使用的数据大多是 16 位的 short 类型, 而 C6000 的内部数据总线和寄存器都是 32 位的, 因此还可以采用字长优化的方法进行优化<sup>[23]</sup>。具体做法是: 做点积运算时, 每次使用 1 个 LD 指令取 2 个 16 位数据, 并进一步用 C6000 的 2 个 16 位乘法器在 1 个周期内并行完成 2 个 16 位乘法。在 C6000 的 C 编译器中提供了两个特殊的内联函数 `_mpy` 和 `_mpyh`, 该函数可以实现汇编语言中的 MPY/MPYH 功能, 高效率实现 C 语言中的字长优化问题, 使 C 代码

的效率可以达到 90% 以上。经过字长优化以后, 点积算法的核心循环变成了单周期, 而且执行包达到了 8 条指令并行的最佳性能, 点积函数的执行时间也从 159 周期缩短至了 66 周期, 性能提高 1 倍, 由此可见, 采用以上优化方法可以大大提升系统的数据处理性能。

在进行上述诸多硬件适配性优化后, 系统成功实现了基于 DSP 的 KCF 算法, 并且算法在本平台的运行效率也相对较高, 在实际的工程环境中能够实现平均 25 fps 的运行速度, 满足系统实时运行的需求以及其他目标跟踪性能指标要求。最终的算法流程图如图 2 所示。

对于整个目标跟踪系统, 其在实际工程环境中的工作步骤如下:

步骤 1. 系统上电后进入自检, 自检完成后进入工作状态;

步骤 2. 视频采集电路启动拍摄, 并将图片帧流入视频解码电路进行模/数转换, 随后将转换结果输出至 FPGA 进行数据缓存控制;

步骤 3. 视频显示电路将数字视频信号发送至上位机控制台进行显示, 并等待操作人员框选目标区域 (目标框的大小可根据跟踪目标的实际情况进行手动调整);

步骤 4. 目标区域框选完成后, 系统进入目标跟踪状态, 并将含有该目标区域的图片作为跟踪视频的第一帧图片;

步骤 5. DSP 从缓存区域读入帧数据, 并调用目标跟踪算法更新检测器参数, 输出计算得到的目标区域, 同时输出控制量使视频采集电路中的摄像头跟随目标区域进行移动跟踪;

步骤 6. 对后续帧数据重复上述步骤 4 到步骤 5, 若更换跟踪目标, 则重复步骤 1 到步骤 5。

## 4 实验结果与分析

根据本系统的应用场景, 实验测试分为标准数据集测试和实际工程场景测试两部分。

### 4.1 标准数据集测试

标准数据集测试实验中使用的测试视频为 Wu 等人在 2013 年 CVPR 上提出的测试基准<sup>[24]</sup>, 该测试集 Benchmark 中有 50 个测试视频。在本实验中, 由于系统的硬件平台是以 DSP 为核心的嵌入式平台, 其核心处理器的处理性能与通用 PC 机的处理性能不在同一

级别,这就导致平均跟踪帧率这一性能评价指标与原算法差距较大,但由于硬件平台本身的应用背景不同,对比该指标的性能优劣并没有实际意义,因此,在标准数据集测试实验中,本文仅关注跟踪准确率这一指标. Benchmark 中的 50 个测试视频提供了每一帧图像的人工标注值 ground truth, 该值反映的是每一帧图像中目标框的中心坐标值. KCF 算法的作者使用 20 像素精

度作为算法准确率的评判标准,即目标框中心的预测值与 ground truth 之间的欧氏距离在 20 个像素以内则认为该帧跟踪正确,并以正确跟踪帧数的比例作为算法的准确率. 本实验以 KCF 作者公开的算法的 Matlab 代码在 Benchmark 上的测试结果<sup>[5,12]</sup>作为实验基准与本系统在相同测试集上的运行结果进行横向比较 (Gaussian 核+HOG 特征),测试结果如表 1 所示.

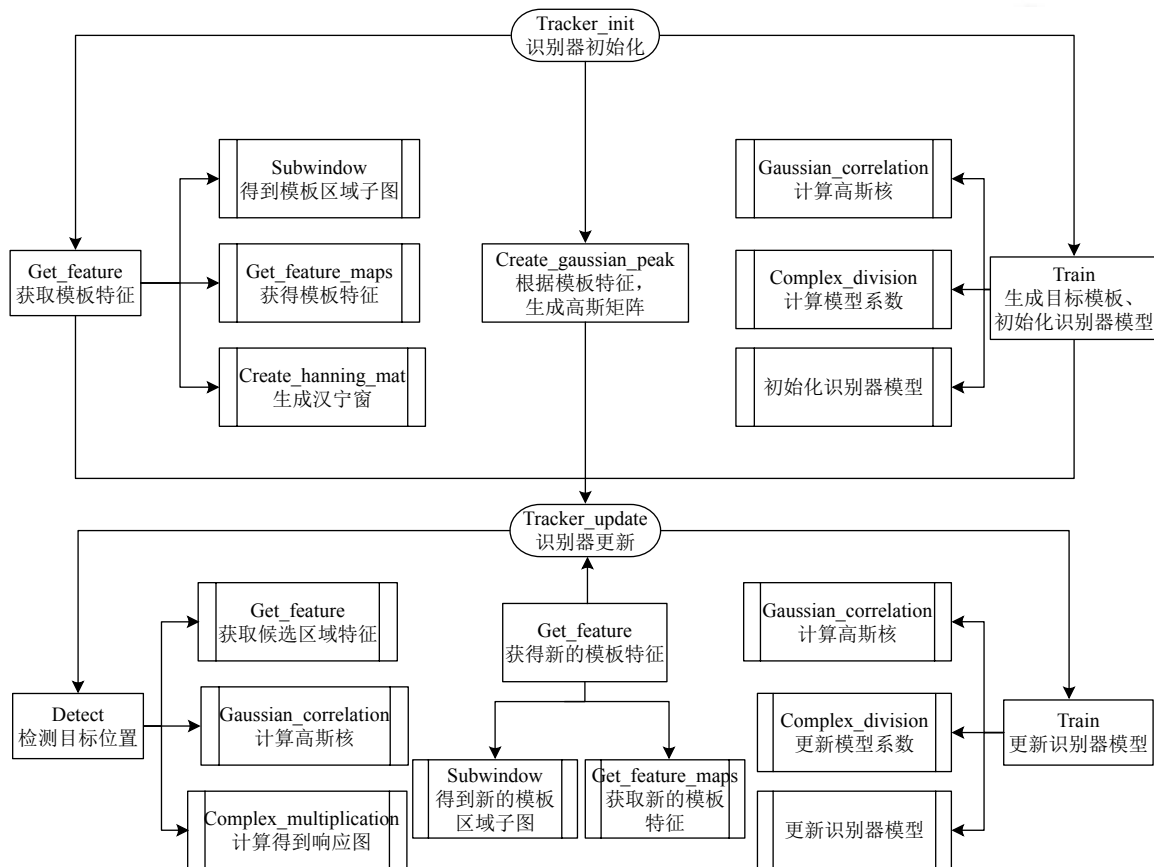


图 2 目标跟踪算法流程图

从测试结果来看,算法在本平台上的准确率与原算法相比下降了 0.7%,可以认为本系统对原算法进行了复现,算法移植效果良好,与原作者算法的准确率比较来看符合实验预期.

#### 4.2 实际工程场景测试

为模拟真实的工程环境,保证测试视频片段的多样性,本实验使用了不同场景、不同复杂度的多段视频作为测试素材,以检测本系统在不同测试环境下的鲁棒性. 测试视频均由本系统的图像采集电路实时采集,视频单帧图像的分辨率为 720×576,通过视频解码

电路、视频显示电路和上位机通信电路将采集到的实时信号投射在上位机界面,操作人员可通过键盘操作选定信号中的待跟踪目标,并调整跟踪框尺寸. 操作完成后系统便进入工作状态,视频采集电路的 CCD 摄像机可自动跟随目标进行二维转动(控制信号由本系统的控制模块产生),以保证跟踪目标在整个实验过程中一直处于视野范围内,并在每一帧图像中框选出目标位置,同时在上位机的视频显示界面实时显示. 其中每一帧的坐标原点默认为图像中心位置,在跟踪过程中,视频下方会实时显示跟踪框的中心坐标值.

表1 标准数据集测试结果

测试视频	作者	本系统
Basketball	0.923	0.891
Biker	0.458	0.550
Bird1	0.069	0.042
BlurBody	0.584	0.580
BlurCar2	0.939	0.944
BlurFace	1.000	0.925
BlurOwl	0.228	0.214
Bolt	0.989	1.000
Box	0.415	0.433
Car1	0.739	0.712
Car4	0.950	0.900
CarDark	1.000	0.862
CarScale	0.806	0.753
ClifBar	0.445	0.500
Couple	0.357	0.139
Crowds	1.000	1.000
David	1.000	1.000
Deer	0.817	0.714
Diving	0.535	0.317
Dog	0.992	0.954
DragonBaby	0.336	0.257
Dudek	0.877	0.591
Football	0.796	0.635
Freeman4	0.530	0.664
Girl	0.864	0.884
Human3	0.536	0.423
Human4	0.252	0.353
Human6	0.290	0.248
Human9	0.725	0.606
Ironman	0.217	0.352
Jump	0.057	0.420
Jumping	0.339	0.246
Liquor	0.977	0.900
Matirx	0.170	0.137
MotorRolling	0.049	0.026
Panda	0.364	0.913
ReadTeam	1.000	1.000
Shaking	0.025	0.300
Singer2	0.945	0.849
Skating1	1.000	0.976
Skating2	0.768	0.657
Skiing	0.074	0.021
Soccer	0.793	0.763
Surfer	0.910	0.862
Sylvester	0.843	0.915
Tiger2	0.356	0.548
Trellis	1.000	1.000
Walking	1.000	0.966
Walking2	0.440	0.387
Woman	0.938	1.000
<b>平均值</b>	<b>0.634</b>	<b>0.627</b>

由于每次实验中所框选出的目标区域大小不同,因此系统在不同实验中的跟踪速度也有差别,实际测

试中,其平均处理帧率为 25 帧/秒,满足工程应用中的实时性指标;在跟踪准确率方面,由于测试数据由系统实时采集得到,无法对视频进行 ground truth 标定,因此跟踪准确率无法精确量化.从视觉上看,系统的跟踪准确率处在较高水平,平均可达 95% 以上,与标准数据集测试实验相比性能有较大提升,这主要是由于本系统应用于实际工程环境中时,视频采集电路由二轴云台进行稳定控制,云台的控制模块可以根据视频处理模块计算输出的目标脱靶量对云台运动进行实时修正,进而可以最大限度地保证 CCD 摄像机的视轴始终处于稳定状态,因此,采集到的视频数据相较于标准测试集中的视频数据更加稳定,抖动较小,且目标运动更连贯,目标区域可以一直处于视野中心附近,这对目标跟踪模块的功能实现具有积极作用.图 3 为实际工程场景测试实验中使用的上位机主界面,界面左上角为视频跟踪显示区域.



图3 上位机主界面

#### 4.2.1 场景一

目标的平移运动是目标跟踪过程中的最简单最基本的运动,本场景目的是用来测试系统的最大跟踪角速度,视频中跟踪目标在测试人员的控制下以一定速度在桌面上沿椭圆轨迹做循环往复运动,测试结果如图 4 所示.在实际的工程环境中,本系统的最大跟踪角速度可达 20 度/秒.

#### 4.2.2 场景二

在实际的工程环境中,待跟踪的目标所处的背景环境中往往会出现与目标相似的物体,这对跟踪结果会产生一定的干扰,同时目标在视野中的远近位置变化还会引起尺度变化,系统能否顺利应对这种干扰和

变化也是系统鲁棒性的重要参考标准. 本场景是实际公路的航拍片段, 以道路上某辆行驶中的小汽车作为跟踪目标, 其余汽车以及背景中的道路标识为干扰项, 同时在汽车由远及近的过程中会发生明显的尺度变化.

测试结果如图5所示, 从测试结果来看, 本系统在该场景下能够顺利避开背景干扰信息, 准确跟踪指定目标, 且目标在由远及近的过程中, 系统均可实现对目标的跟踪.

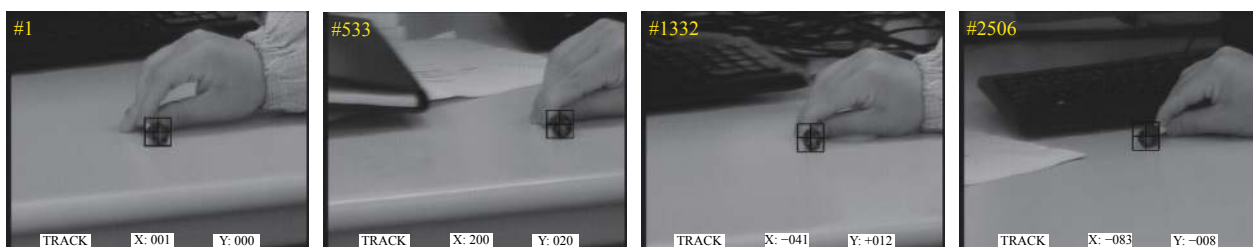


图4 场景一下的跟踪结果



图5 场景二下的跟踪结果

### 4.2.3 场景三

实际应用场景中, 跟踪目标所处环境一般会出现明暗、阴影等变化, 目标有时也会被遮挡, 导致其在视野中发生较为明显的形变等. 跟踪系统能否在这种复杂多变的环境下完成跟踪任务, 亦是衡量系统可靠性和稳定的重要参考标准. 本场景中将对公路上行驶的大巴车进行跟踪, 在其行驶过程中会经过阴影区、树木遮挡区等, 并会发生一定程度的形变. 测试结果如图6所示, 在整个跟踪过程中, 当跟踪目标所处的环境出现阴影、遮挡时, 系统均能正常工作, 完成跟踪

任务.

## 5 结语

为实现目标跟踪算法的工程应用问题, 本文提出一种以 DSP 为核心处理器、FPGA 为协处理器的嵌入式目标跟踪系统设计方案, 并基于此平台对 KCF 算法进行优化与实现, 使之能满足实际工程应用的指标要求. 本系统具有“小规模、低成本、高效率”的特点, 在典型工程环境中能实现良好的目标跟踪效果. 本目标跟踪系统中还配套开发了上位机操作平台和视频采集



电路的运动控制模块,使得本系统可很好兼容其余功能模块,极大提高工程项目的开发效率.本系统的嵌入

式硬件平台和算法优化思想具有通用性,为计算机视觉领域内的各类算法的嵌入式应用实现提供参考.

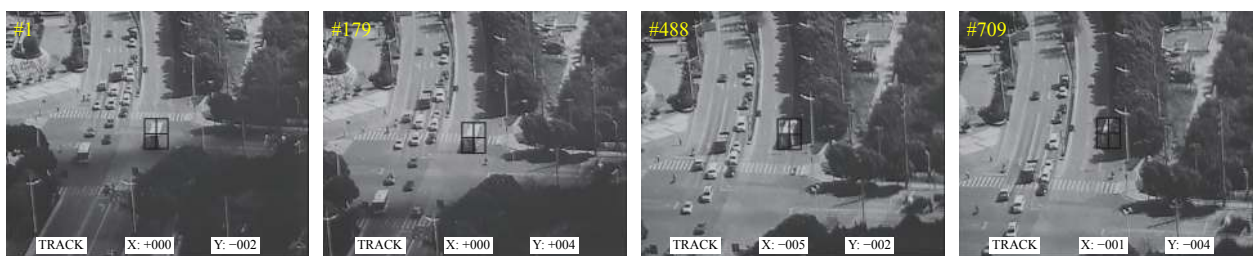


图6 场景三下的跟踪结果

### 参考文献

- 王新洁. 单目标跟踪算法的研究与分析. 计算机系统应用, 2012, 21(5): 245–249.
- 于明. 基于嵌入式系统的运动目标跟踪视频监控系统的的设计. 自动化与仪器仪表, 2015, (12): 17–19.
- 朱建武, 李榕, 凌云. 基于 FPGA 的目标跟踪系统设计与实现. 计算机工程, 2012, 38(5): 243–246. [doi: 10.3969/j.issn.1000-3428.2012.05.076]
- 刘磊, 蔡坚勇, 马正文, 等. 基于遮挡检测的尺度自适应相关滤波跟踪. 计算机系统应用, 2018, 27(10): 285–290. [doi: 10.15888/j.cnki.csa.006589]
- Henriques JF, Caseiro R, Martins P, *et al.* High-speed tracking with kernelized correlation filters. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(3): 583–596. [doi: 10.1109/TPAMI.2014.2345390]
- Henriques JF, Caseiro R, Martins P, *et al.* Exploiting the circulant structure of tracking-by-detection with kernels. In: Fitzgibbon A, Lazebnik S, Perona P, *et al.*, eds. Computer Vision - ECCV 2012. Berlin, Heidelberg: Springer, 2012. 702–715.
- 江维创, 张俊为, 桂江生. 基于改进核相关滤波器的目标跟踪算法. 计算机工程, 2018, 44(11): 222–227.
- 张乘龙, 夏筱筠, 柏松, 等. 基于 KCF 跟踪算法的目标轨迹记录系统. 计算机系统应用, 2017, 26(5): 113–118. [doi: 10.15888/j.cnki.csa.005780]
- International Telecommunication Union, Radio Communication Sector, Broadcasting Service (Television). The digital video standard according to ITU-R BT. 601/656. 2001.
- 康艳霞, 曹剑中, 田雁, 等. 实时视频处理系统中乒乓缓存的设计. 弹箭与制导学报, 2007, 27(4): 218–221. [doi: 10.3969/j.issn.1673-9728.2007.04.068]
- 杜建宝. 基于 FPGA 的视觉目标跟踪算法研究与实现[硕士学位论文]. 天津: 天津理工大学, 2018.
- 刘志强. 基于核相关滤波的高速目标跟踪算法研究与系统实现[硕士学位论文]. 西安: 西安电子科技大学, 2015.
- 柯俊敏, 洪亲, 蔡坚勇, 等. 融合颜色特征的核相关滤波器目标长期跟踪算法. 计算机系统应用, 2018, 27(4): 190–195. [doi: 10.15888/j.cnki.csa.006313]
- 闫钧华, 姜惠华, 杨勇, 等. DSP 实现基于改进压缩跟踪算法的目标实时跟踪. 电子设计工程, 2016, 24(20): 9–12, 16. [doi: 10.3969/j.issn.1674-6236.2016.20.003]
- 杜文彬, 毛征, 梅伟军, 等. 基于 DSP6678 的 KCF 算法实现及优化系统设计. 国外电子测量技术, 2017, 36(7): 62–67. [doi: 10.3969/j.issn.1002-8978.2017.07.016]
- Kristan M, Matas J, Leonardis A, *et al.* The visual object tracking VOT2015 challenge results. Proceedings of the IEEE International Conference on Computer Vision Workshop. Santiago, Chile. 2015. 564–586.
- Hare S, Saffari A, Torr PHS. Struck: Structured output tracking with kernels. Proceedings of International Conference on Computer Vision. Barcelona, Spain. 2011. 263–270.
- 陈仁爱, 凌强, 徐骏, 等. 基于 DSP 的实时圆检测算法的设计实现与优化. 微型机与应用, 2016, 35(11): 93–96, 100.
- 张雷, 王延杰, 孙宏海, 等. 采用核相关滤波器的自适应尺度目标跟踪. 光学精密工程, 2016, 24(2): 448–459.
- 张雷. 复杂场景下实时目标跟踪算法及实现技术研究[博士学位论文]. 长春: 中国科学院研究生院(长春光学精密机械与物理研究所), 2016.
- Ali U, Malik MB. Hardware/software co-design of a real-time kernel based tracking system. Journal of Systems Architecture, 2010, 56(8): 317–326. [doi: 10.1016/j.sysarc.2010.04.008]
- TMS320C6000 Programmer's guide. Texas Instruments Incorporated, 2011.
- 纪铁军, 任雨军, 赵爱明. TMS320C6000 系列 DSP 的 C 代码优化方法. 单片机与嵌入式系统应用, 2003, (7): 76–77. [doi: 10.3969/j.issn.1009-623X.2003.07.024]
- Wu Y, Lim J, Yang MH. Online object tracking: A benchmark. Proceedings of 2013 IEEE Conference on Computer Vision and Pattern Recognition. Portland, OR, USA. 2013. 2411–2418.