

一种基于 HLS 的目标跟踪 IP 核设计方法

冯庭燕¹ 齐宇心¹

FENG Tingyan QI Yuxin

摘要

视频目标跟踪一直是计算机视觉领域非常重要的一个研究分支,多种目标跟踪算法都有着非常亮眼的跟踪效果。但为了适应更多应用场景,目标跟踪算法的结构不断复杂化,计算量的增加使跟踪精度高的算法难以保证实时性的要求。针对这一问题,提出了一种基于 HLS 的目标跟踪 IP 核设计方法,选用目标跟踪领域经典算法——核相关滤波算法(KCF),通过分析算法原理中的可并行性,从循环和数组两个方面实现算法由串到并的转换,利用高级综合工具(HLS)将优化后算法打包为 IP 核,充分结合软硬件设计优势,在保证跟踪精度的基础上提高算法运算效率。经验证,目标跟踪 IP 核的处理速率可达 66.7 帧/s,满足视频实时跟踪的要求。

关键词

视频目标跟踪;核相关滤波;HLS;并行优化;IP 核设计

doi: 10.3969/j.issn.1672-9528.2024.03.021

0 引言

目标跟踪一直是计算机视觉领域的一大研究热点,最早应用于武器制导、雷达定位、视觉导航等军事领域^[1],随着计算机技术的不断发展,图像处理能力不断提高,目标跟踪逐渐应用于日常生活的多个领域,包括智能交通、无人驾驶、视频监控等^[2]。

经过几十年的快速发展,目标跟踪领域提出了多种行之有效的算法,在跟踪效果上有着非常亮眼的表现。相关滤波类算法是其中最常用的跟踪算法之一,在应对强遮挡、外观形变、旋转、强烈光照等背景变化时具有良好的鲁棒性^[3]。随着应用场景的复杂化,研究者们通过优化算法精度实现更为精确的跟踪,但精度的提高往往依赖于更为复杂的算法框架,导致图像数据处理量大幅增加,无法保证跟踪的实时性。故而,如何平衡精度与速度性能一直是视频目标跟踪领域发展的重点与难点^[4]。

随着 FPGA 等嵌入式平台的不断发展,国内外研究者开始重视 FPGA 高并行度在提高算法运算效率上的优势^[5],通过并行设计加速算法计算,以实现不降低精度的同时保证跟踪实时性。但由于跟踪算法涉及大量图像处理过程,仅凭硬件语言难以实现其复杂架构,导致开发难度较大。而高层次综合工具(HLS)允许设计者使用 C、C++ 等高级语言编程算法,以并行方式优化算法流程,再将其转换为寄存器传输级(RTL)电路应用于 FPGA,充分结合软硬件优势,有效缩短开发周期^[6],对基于硬件平台的目标跟踪系统设计具有

非常重要的意义,在视频目标跟踪领域具有良好的应用前景。

本文选用相关滤波类算法中的代表算法——核相关滤波算法(kernel correlation filter, KCF)^[7],提出了一种基于 HLS 平台的目标跟踪算法 IP 核设计方法,深度分析算法流程中的可并行性,实现算法并行计算,在保证跟踪精度的同时大大提高算法计算效率,实现高精度实时跟踪,为目标跟踪算法的硬件系统设计提供基础理论研究。

1 KCF 算法理论基础

2014 年, KCF 算法的提出,充分展示了相关滤波类算法在跟踪性能上的优势,其基于相关滤波算法框架^[8],采用循环矩阵^[9]增加训练样本数量,通过岭回归、核函数等方式优化跟踪检测计算过程,全面提高算法性能。下面对核相关滤波算法的实现原理做简要介绍。

1.1 循环矩阵

KCF 采用循环移位原理,在原图基础上产生多个水平和垂直方向位移图,从而达到增加训练样本、减少冗余样本的目的。

训练样本数量的增加可以有效提高相关滤波器的检测精度,但同时也会增加算法运算量。为了解决这一问题, KCF 巧妙应用循环矩阵傅里叶可对角化的性质,简化复杂的矩阵运算,实现快速训练与检测。

循环矩阵傅里叶可对角化的具体计算公式为:

$$X = Px = F \cdot \text{diag}(\hat{x}) \cdot F^H \quad (1)$$

式中: x 表示一个样本数据矩阵, X 为其循环矩阵, $\hat{x}$ 为 x 的傅里叶变换, F 为离散傅里叶变换的常数矩阵, F^H 为 F 的

共轭矩阵。

可以看出,通过傅里叶变换,原本复杂耗时的矩阵运算转换为在频域内向量的点乘,大大减少了运算量。

1.2 岭回归与核函数

岭回归思想通过构造一个函数 $f(z)=\omega^T z$, 使训练样本 x_i 和回归目标 y_i 有最小的平方误差,其公式为:

$$\min_{\omega} \sum_i (f(x_i) - y_i)^2 + \lambda \|\omega\|^2 \quad (2)$$

式中: λ 为防止过拟合的正则化参数。

利用傅里叶可对角化性质进行化简,可得到频域内岭回归问题的最终解为:

$$\hat{\omega} = \frac{\hat{x}^* \odot \hat{y}}{\hat{x}^* \odot x + \lambda} \quad (3)$$

式中: $\hat{\omega}$ 、 $\hat{x}$ 、 $\hat{y}$ 分别表示 ω 、 x 、 y 的离散傅里叶变换。

岭回归的求解针对的是线性问题,但在实际计算时大多数是非线性回归,非线性函数相比于线性函数更为复杂,需要耗费更多时间。针对这一问题,KCF 使用核函数思想将低维空间的计算映射到高维空间中,使非线性样本变得线性可分。核化的岭回归函数可以表示为:

$$\hat{\alpha} = \frac{\hat{y}}{\hat{k}^x + \lambda} \quad (4)$$

式中: $\hat{k}^x$ 是核函数矩阵 $K=C(k^x)$ 的第一行,表示样本 x 与自身的核相关性。

同时考虑训练样本 x 和候选区域样本 z ,则核函数矩阵 K 可以表示为 $K=C(k^x)$,所有候选样本的回归值计算可以表示为:

$$\hat{f}(z) = k^z \odot \hat{\alpha} \quad (5)$$

向量 $\hat{f}(z)$ 包含所有循环矩阵的输出检测响应值,仅需要一次计算就可以获得所有候选样本的相关滤波响应结果,取最大值对应的候选样本位置即为预测目标位置,从而实现对目标的快速跟踪。

1.3 方向梯度直方图

相关滤波算法通过构建目标的特征模型训练跟踪分类器检测目标响应,所以目标特征描述的精确性直接影响跟踪结果。KCF 采用方向梯度直方图 (histogram of oriented gradient, HOG) [10] 训练相关滤波模板用以检测,通过构建图像的局部梯度方向直方图表征目标纹理,能够很好地描述形状、轮廓等信息,在几何形变、强烈光照等情况下有明显优势。

HOG 特征的提取过程可以总结为三步。

(1) 图像预处理:将彩色 RGB 图像转换为灰度图像,便于后续计算。

(2) 梯度计算:根据式 (6) 和式 (7) 计算每个像素

点的梯度幅值 $G(x, y)$ 和梯度方向 $\alpha(x, y)$ 。

$$G(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2} \quad (6)$$

$$\alpha(x, y) = \tan^{-1} \frac{G_y(x, y)}{G_x(x, y)} \quad (7)$$

(3) 构建梯度直方图: HOG 按照一定像素数量将图像划分为单元 (cell) 和块 (block), 在每个 cell 内根据梯度方向区间对梯度幅值进行加权,形成各区间内幅值加权后的直方图特征,然后以 block 为单位统计每个 cell 的特征,最后串联所有 block 的特征就形成了最终的 HOG 特征。

1.4 KCF 算法框架

KCF 算法基于相关滤波的基本流程,采用循环矩阵增加训练样本数,使用 HOG 特征训练滤波模板,以岭回归和核函数思想简化跟踪计算过程,其流程如图 1 所示。

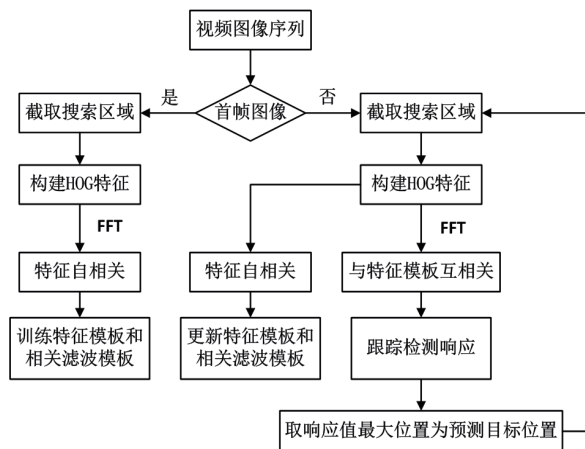


图 1 KCF 算法流程图

当前视频图像帧为首帧时: (1) 根据预置的首帧坐标信息截取搜索区域; (2) 提取搜索区域内的 HOG 特征, 并对其做二维 FFT 转换, 作为初始特征模板; (3) 通过核函数做特征自相关计算, 训练相关滤波模板。

当前视频图像帧为非首帧时: (1) 根据上一帧的预测目标位置截取搜索区域; (2) 提取搜索区域内的 HOG 特征, 并对其做二维 FFT 转换; (3) 将 FFT 结果与特征模板做特征互相关, 以相关结果与相关滤波模板进行跟踪检测响应, 取响应值最大所在位置区域为预测目标位置区域, 完成跟踪检测; (4) 以新的预测目标位置为基准截取新的搜索区域, 提取新区域的 HOG 特征, 更新特征模板和相关滤波模板, 用于新一帧的跟踪检测。

2 基于 KCF 的目标跟踪 IP 核设计

在基于 PC 机的应用中, KCF 算法以串行的方式实现整个功能的计算, 像素点间的遍历皆以行列顺序执行, 算法的计算速度只依赖于算法本身结构的复杂性。但深入分析算法原理, 跟踪结果的计算都是基于对图像数据的处理, 具有局

部性和重复性等特点,满足FPGA并行设计原则。基于此,本文从总体和模块两个方面优化算法构架,使其满足并行计算原则,再结合HLS优化指令实现算法由串到并转换。

2.1 总体设计

为了清晰算法流程,有效减少算法编译中的布局布线和时序控制问题,实现更好的优化效果,本文先从总体角度对算法进行功能模块划分,即区域截取模块、特征提取模块、相关计算模块、跟踪检测模块和模板更新模块。模块实现流程可总结为图2。

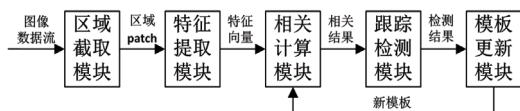


图2 总体设计流程图

优化后模块间算法实现流程为:当接受到新一帧视频图像后,区域截取模块截取出固定大小的搜索区域,特征提取模块提取出HOG特征,并对其做二维FFT转换,生成特征向量,线性相关模块实现相关运算输出相关结果,跟踪检测模块根据相关结果计算滤波响应结果,并判断响应最大值对应的位置,得出跟踪结果。模板更新模块根据相关结果更新特征模板和相关滤波模板,用于后续跟踪计算。

2.2 模块设计

2.2.1 循环优化设计

KCF的跟踪计算过程本质上都是对图像数据的处理,使用C语言编程时,在代码设计上多采用for循环完成对每个像素的遍历操作,以串行方式完成循环计算,会大大降低计算效率。但实际上,循环中的计算步骤并不是完全串行的,像素的处理过程满足局部性和重复性,可做并行流水化设计。本文以特征提取模块中的HOG特征计算过程为例,介绍本文对for循环的优化设计方式。

HOG特征是基于灰度图像的像素值计算得出的,不论是RGB图像的转灰度处理,还是梯度计算和直方图统计都是通过多层或单层for循环实现各像素点的读、写、计算操作,且计算方式统一,行方向与列方向的处理过程相互独立,可做并行优化处理。

本文使用HLS的PIPELINE优化指令对for循环做并行优化,具体原理如图3所示。HLS通过PIPELINE指令将循环内展开为流水线,允许计算步骤中的每个阶段在不同的时钟周期内并行执行。

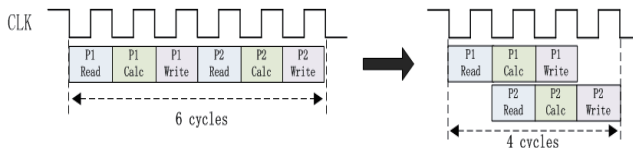


图3 PIPELINE指令优化示意图

如图3所示,在计算HOG特征的计算过程中, P_1 和 P_2

为两个相邻像素点,for循环需要分别实现对像素点的读、计算和写三个步骤。未优化前,函数完成一个像素的运算需要3个时钟周期, P_2 的计算需要等 P_1 计算完成后开始,两个像素的计算需要6个周期。PIPELINE指令允许 P_1 和 P_2 的计算重叠,以流水线的方式完成像素间数据处理,在资源复用的同时有效减少了循环计算所需要的时钟周期数,实现串行到并行的转换,提高计算效率。

PIPELINE的流水线可以有效提高单层循环的计算效率,而对于算法中的多层循环,默认情况下编译器将按照顺序,等上层循环计算完成后再调用下层循环。本文为了更大程度上实现循环的并行化计算,可以在最外层循环中对PIPELINE指令赋予rewind选项,该选项支持以重叠方式执行循环的流水计算,即循环调用的流水化,在执行当前循环的同时执行下一循环,可进一步提高多层循环的并行化程度,进一步缩短循环计算所需要的时间。再结合树流水化思想,将像素处理中的乘加运算拆分为分步形式,加入指令后多步之间即可实现流水化,大大提高循环的并行度的同时也可节省资源使用。

2.2.2 数组优化设计

KCF算法中,搜索区域图像patch、FFT处理后的特征向量、相关结果以及跟踪响应结果皆以多维数组的形式表示,算法计算过程需要遍历每一维数组,数据量较大时对计算效率有明显影响,故而数组的优化设计也尤为重要。深入分析算法对数组的处理方式,不论是图像像素数据还是向量数据,计算方式也都满足重复性原则,且数组间的运算过程相互独立,满足并行设计的基本要求。基于此,本文从向量层面设计数组的并行优化方式。

以特征提取模块为例,HOG特征需要经过二维FFT转换生成用于跟踪检测的特征向量,本质上就是对 $M \times N \times k$ 的三维数组依次进行行和列的二维FFT转换,虽然一组行列向量间的转换过程有数据依赖关系,但 k 个 $M \times N$ 的二维数组间相互独立,故而可以从第三维度将三维数组展开为 k 个二维数组,并行实现数组间的二维FFT。本文通过PARTITION优化指令实现数组的展开操作,设置参数 $dim=3$,可以将第三维度的数组展开,如图4所示。从图4中可以看出,加入PARTITION优化指令后,特征向量以第三维度为基准展开为 k 个二维数组。原本特征向量的二维FFT计算需要依次遍历 k 个二维数组,上一数组计算完成后再开始下一数组计算,优化后, k 个二维数组的二维FFT过程转为并行,理论上需要的时钟周期数减少为原来的 $1/k$ 。再结合2.2.1节中对循环的并行优化方式,就可以完全实现多维数组由串到并的转换,算法整体的计算效率都会有大幅提升。

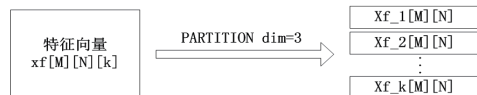


图4 PARTITION指令优化示意图

3 IP 核性能验证

3.1 IP 核功能验证

本文使用 VIVADO HLS 中的 C 仿真台测试算法功能,以目标跟踪经典数据集 OTB100 中的“Blot”为测试样例验证 IP 核的跟踪效果。

图 5 展示了算法在“Bolt”数据集下的跟踪结果。可以看出,在视频的各个图像帧中,算法在目标快速运动、快速形变等情况下都很好地保持了对目标的精确跟踪,说明本文设计的目标跟踪 IP 核在精度上具有明显优势。

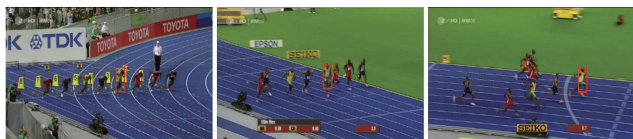


图 5 “Bolt”跟踪结果图

3.2 IP 核性能验证

本文使用 VIVADO HLS 的综合布局布线结果验证 IP 核的处理效率和资源耗费情况,说明本文并行优化设计对算法的加速效果。本文选用的 Xilinx FPGA 型号为 xc7z100ffg900-2,设定时钟周期为 8 ns。

图 6 和图 7 分别展示了优化前后目标跟踪 IP 核的时间和资源使用情况。可以看出,优化后,IP 核计算所需要的时钟周期数由 10 606 373 减少到 1 926 750,运算速度提高了 5.5 倍,实际视频处理速率约为 66.7 FPS,符合实时跟踪的速度要求。同时,优化后的 IP 核 LUT、DSP48E 等资源使用量也有不同程度的增加,这也说明了 FPGA 以资源换速度的特性。

