

Intel音视频团队与阿里巴巴图像处理专家联合撰写，知名专家傅文庆、邹复好、
Vadim Pisarevsky、周强（CV君）联袂推荐
深入解析OpenCV DNN 模块、基于GPU/CPU的加速实现、性能优化技巧与可
视化工具，以及人脸活体检测等应用

OpenCV深度学习应用 与性能优化实践

吴至文 郭叶军 宗炜 李鹏 赵娟 著



OpenCV深度学习应用 与性能优化实践

吴至文 郭叶军 宗炜 李鹏 赵娟 著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

OpenCV 深度学习应用与性能优化实践 / 吴至文等著. —北京: 机械工业出版社, 2020.6
(智能系统与技术丛书)

ISBN 978-7-111-65646-3

I. O… II. 吴… III. 机器学习—算法 IV. TP181

中国版本图书馆 CIP 数据核字 (2020) 第 085938 号

OpenCV 深度学习应用与性能优化实践

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 高婧雅

责任校对: 殷虹

印刷: 北京市荣盛彩色印刷有限公司

版次: 2020 年 6 月第 1 版第 1 次印刷

开本: 186mm×240mm 1/16

印张: 18.75

书号: ISBN 978-7-111-65646-3

定价: 89.00 元

客服电话: (010) 88361066 88379833 68326294

投稿热线: (010) 88379604

华章网站: www.hzbook.com

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东



华章图书

一本打开的书，
一扇开启的门，
通向科学殿堂的阶梯，
托起一流人才的基石。

OpenCV 以其易得、易懂、易用为国内学术界和工业界所熟知。作为经典的图形图像类开源软件，OpenCV 可以说是兼具“大而全”和“小而精”特点。“大而全”是指 OpenCV 全面涵盖了从教科书式的基本图像处理单元算法，到更复杂的高级算法，直至最新的围绕机器学习的算法。“小而精”是指其每个算法的实现被社区反复锤炼、优化，甚至满足一些项目需求的 KPI，可以直接用于产品化。

本书的作者团队长期从事图形图像和视频编解码处理的算法研究、软件的功能实现及性能加速，各自在 OpenCV 项目里都有重要贡献，直至成为一些模块的维护者。本书从 OpenCV 入门，结合行业热点，花大量笔墨介绍机器学习相关实现及平台相关的性能优化，贴近实战，为学生、工程技术人员提供了实用指导。

作者团队所在的 Intel 开源软件中心音视频团队，自 2008 年开始从事 GPU 内核驱动开发，开发项目贯穿整个图形图像软件栈，涵盖开源软件桌面（X.org）、3D 加速协议 OpenGL（Mesa）、通用计算加速协议 OpenCL（Project Beignet）、视频编解码与处理（FFmpeg/GStreamer）及基于软件的全栈式摄像头流水线处理算法实现（libXCam）。很多团队成员如今在流媒体框架、图形图像处理等主要开源软件社区成长为项目维护者。

开源软件开发作为一种开发模式，社区互动是必不可少的组成部分。我们和 OpenCV 社区的互动，最早可以追溯到 2010 年。彼时 OpenCV 已经在业内流行了，但来自国内的贡献很少。最初，我们也经历了彷徨和不安，担心代码会不被接受。经过一些尝试后，我们逐渐与社区、OpenCV 项目的维护者 Vadim 先生建立了良好的合作关系。待后来 Intel 收购了 Itseez 公司之后，大家成为同事，合作就变得更多了。本书的各位作者作为参与 OpenCV 项目的开发者，贡献了重要算法的优化和实现。他们希望通过本书，分享自己的成长经历，携手国内开源社区的程序员和工程师一起参与 OpenCV 项目的开发。

图形图像技术自诞生之日起，就与网络技术的发展相辅相成。随着机器学习的发展，技术热点也从围绕存储和分发，越来越多地转向内容的分析和挖掘。华为的任正非先生曾

经说过，下一个时代是图像时代。智能手机的普及使得拍摄成为最通俗易懂的信息产生方式，由此产生了海量数据。为这些非结构化数据的有效挖掘和利用将催生下一个行业巨人。“把图像处理变得像文字处理一样简单”，一直是本书作者们及其所在团队追求的技术方向。机器学习和图像处理相结合，必将对机器人、工业自动化、医疗和生物技术带来深远的影响。5G的低延时、物对物的连接特性，也将给图形图像技术注入新的催化剂并带来广阔的应用。有幸能为本书作序，借此与读者共勉。

傅文庆

Intel 公司系统软件产品事业部研发总监

于美国俄勒冈州

序二

应 Intel IAGS 的赵娟女士邀请为本书写序，我深感荣幸。本书由 Intel 开源软件中心音视频团队一线工作人员吴至文、郭叶军、宗炜、李鹏（前成员）、赵娟共同撰写，可以说是集体智慧的结晶。该团队成员目前都是资深研发工程师，有的深耕于 OpenCV 加速，有的是视频处理和深度学习领域的专家。在本书中，他们由浅入深地对 OpenCV 在深度学习上的应用和性能优化进行了全面解读。我在阅读本书时获益良多，尤其是加深了对 OpenCV 在深度学习方面的实际应用的理解。我相信本书能够覆盖相当大一部分学习者的需求。

众所周知，OpenCV 是 Intel 公司主导的开源计算机视觉库，它实现了一系列图像处理和计算机视觉算法，目前已经延伸到计算机视觉的各个领域，其功能几乎涵盖了每个研究方向。另外，OpenCV 实现的算法不仅紧跟视觉前沿，而且在性能优化方面做了很多工作，因功能强大而在学术界和工业界得到了非常广泛的应用。深度学习是机器学习中一种基于对数据进行表征学习的算法，近些年来，得益于数据的增多、计算能力的增强、学习算法的成熟及应用场景的丰富，越来越多的学者开始关注并研究深度学习，并掀起了新一轮人工智能的热潮。OpenCV 为了支持基于深度学习的计算机视觉应用加入了新特性，搭建了一个轻量的深度学习框架，添加了支持网络推理的深度学习模块。

本书面向 OpenCV 和深度学习的初学者，按照循序渐进的学习步骤，详细介绍了一些 OpenCV 和深度学习的基本概念与结构，讲解了各种计算架构下深度学习的计算优化和加速，列举了一些精准实用的项目样例，内容不仅涵盖全面详尽的算法原理，还解释分析相关源代码和实践结果。本书对深度学习的最新发展趋势和主要研究方向进行了全面而综合的介绍，从不同的用户场景出发，对算法进行深度的分析和详细的解释，能够满足初学者对各种计算机视觉的应用需求。

鉴于计算机学科是一个高速发展学科，计算机视觉、深度学习必然同样处于高速发展状态。希望 Intel IAGS 团队在出版本书之后，要随着新技术、新方法的涌现而与时俱进，及时更新内容，为广大读者提供持续支撑和服务。

邹复好

华中科技大学计算机学院教授

于华中科技大学

序三

Among many books about OpenCV that have been published already, this one has some interesting properties that make it special. It has been written by the members of Intel China team who optimized OpenCV deep learning module (OpenCV DNN) for GPU. So, the book contains some in-depth first-hand information that would be difficult to find somewhere else. That includes some rarely discussed topics, such as OpenCL, Vulkan and Halide backends of OpenCV DNN. Another expert-level and very useful topic is discussion with practical examples of how to analyse application performance and then tune it using Intel VTune. At the same time, the book is not just for experts. It also provides a comprehensive guide to OpenCV DNN usage, from the simple models, like image classification, to more complex models, such as object detection. Overall, the book is highly recommended for all the software engineers who want to build highly-efficient deep learning applications using OpenCV.

在已经出版的众多关于 OpenCV 的书籍中，本书具有一些有趣的特性，因之与众不同。本书是由 Intel 中国团队的成员编写的，他们优化了 OpenCV 模块在 GPU 上的性能。因此，本书包含了一些深度的第一手信息，这些信息在其他地方很难找到。其中包括一些很少讨论的有用话题，如 OpenCL、Vulkan，以及 OpenCV DNN 的 Halide 后端。另一个专家级且非常有用的主题是讨论如何使用 Intel VTune 分析应用程序性能，然后根据它提供的信息对程序进行优化。本书的读者对象包含但不限于专家。它为 OpenCV DNN 模块的使用提供了全面的指导，从简单的模型（如图像分类）到更复杂的模型（如对象检测）。总体来说，本书值得强烈推荐给所有希望使用 OpenCV 构建高效深度学习应用程序的软件工程师。

Vadim Pisarevsky

OpenCV team lead

Senior Software Engineer, AIRS(Artificial Intelligence for Robotics and Society)

Institute, Shenzhen

In Russia

序四

非常荣幸能为本书作序，这是一本视觉工程师必备的 OpenCV 深度学习开发指南。

深度学习技术的兴起，极大地促进了图像、视频相关应用的发展。现在，类似人脸动画、人体动作识别等高级别视觉任务等需要用到深度学习技术，甚至最近深度学习也在悄悄进入图片、视频压缩、质量增强等低级别视觉任务领域。

一方面，我们使用深度学习技术时需要在性能强大的服务器上训练模型；另一方面，我们需要将模型部署到移动、嵌入式、物联网设备端等。后者往往更具挑战性。

OpenCV 作为计算机视觉领域最具影响力的开发工具，在模型推理部署方面也做了大量工作，其 DNN 模块为此提供了丰富的技术支持。但仅仅有官方文档和少量例程并不够，长久以来，业界迫切需要一本深入解析 DNN 模块相关技术的参考书，而本书恰好满足了广大读者的需求。

作为一直从事计算机视觉算法研究与开发的工程师，我经常关注 OpenCV 的新版本和新特性，在最近几次的版本更新中，深度学习 DNN 模块都是 OpenCV 开发的重点。OpenCV 提供了大量基础且重要的视觉算法的实现，是快速进行产品原型开发验证的首选工具，其深度学习模块也提供了大量的方便实用的功能，是深度学习模型快速部署的重要工具。

图像和视频的应用计算量通常很大，在产品开发中后期，算法工程师更多的工作往往是程序优化，如何最大化利用硬件计算能力、减少计算冗余是颇具挑战的事情。

本书深入解析了 OpenCV DNN 模块，详述了深度学习引擎的性能优化策略，介绍了在 GPU 和 CPU 上进行计算加速的方法，并通过涵盖计算机视觉主流应用的几个案例展示如何在 OpenCV 中使用深度学习，最后则带着大家完整实现一个人脸活体检测与识别的大项目。本书提供的案例，紧跟技术前沿，贴近实际应用场景，相信对参与工程项目开发的读者具有直接的参考价值。

对于想要学习 OpenCV DNN 深度学习模块，并将其应用到工程实践中的读者，这无疑

是一本首选参考书。

技术是不断迭代更新的，但思想可以让我们走得更远。值得一提的是，本书介绍的优化策略和加速方法，对于开发其他视觉算法也大有裨益。

强烈建议大家不要只停留在读这本书，而要动手把书里的程序跑起来，将书中介绍的方法用起来，当你成功让一段程序取得几倍加速后，你一定会获得巨大的成就感！

周强 (CV 君)

“我爱计算机视觉”公众号负责人



前言

为什么要写这本书

图像和视频由最基本的点（像素）组成，众多的像素按照一定规则排列组成了我们所熟知的图像，一系列的图像组成了我们所认识的视频。从 2013 年开始，随着智能手机的普及，全球进入了大数据时代。除了数据本身的多样性，文件数目呈指数级增加，数据的大小也在大规模增长。其中，最典型的是图像和视频数据的爆发式增长，这不仅源于分辨率和帧率的不断提高（分辨率从标清到高清，从 4K 到 8K，帧率从 25FPS、60FPS 到 120FPS），更多是源于采集设备的多样化和各种基于图像视频的社交软件的流行。图像和视频数据的分析与处理已经在目前的大数据处理中占据了举足轻重的地位。如何对图像和视频数据进行高效快捷的分析与处理是摆在软件工程师面前的一个重要问题，而 OpenCV 是我们解决这个问题的好选择。

OpenCV 是一个计算机视觉开源软件库，它提供了大量的图像视频处理的算法工具。在最近几年，随着深度学习的爆发，计算机视觉的算法研究全面转向了深度学习方法，OpenCV 也应需增加了对深度学习的支持。目前，OpenCV 已经广泛应用于基于深度学习的图像分析处理领域，具体的应用场景包括但不限于右图展示的各领域。

我们团队的伙伴们深耕在视频处理和框架领域十几年。近几年，在开源深度学习框架的 GPU 加速方



面相继有了很多积累。与此同时，我们时常会收到来自各方的关于 OpenCV 和深度学习的咨询，内容从初步入门到高级结构和应用，不一而足。于是，我们就萌生了把所积累的知识以中文图书的形式分享给更多人的想法。

作者团队

■ 吴至文曾为 OpenCV 贡献了 Vulkan 加速代码，在 2018 年欧洲嵌入式 Linux 峰会（Embedded Linux Summit Europe 2018）上分享了 OpenCV 中深度学习总体结构、实现细节和硬件加速相关的话题^①（见下图）。演讲材料可以在 Linux 基金会的官网下载^②。



郭叶军是 FFmpeg 深度学习模块的代码维护者，曾参与开源的 OpenCL 驱动（Beignet）的开发，参与向 Khronos OpenCL 提供了两个扩展库：cl_intel_accelerator^③和 cl_intel_motion_estimation^④。他也是本书主要架构的贡献者，在 2019 年 5 月参与本书的整体重构和重写时，对作者团队提出了更高的质量目标要求，几乎重写了之前的 14 万字内容，使书稿达到目前的可以出版的状态。

我是视频处理和框架团队的研究经理，于 2018 年与 2019 年的中国架构师大会（<http://sacc.it168.com/>）上分别介绍了 OpenCV 深度学习和 AI 在 FFmpeg/Gstreamer 中的应用实践。

宗炜是 OpenCV 的上层项目 libxcam（github.com/intel/libxcam）的维护者，为图像处理算

① 参见 <https://osseu18.sched.com/event/FwGX?iframe=no>。

② 参见 <https://events19.linuxfoundation.org/wp-content/uploads/2017/12/Deep-Learning-in-OpenCV-Wu-Zhiwen-Intel.pdf>。

③ 参见 https://www.khronos.org/registry/OpenCL/extensions/intel/cl_intel_accelerator.txt。

④ 参见 https://www.khronos.org/registry/OpenCL/extensions/intel/cl_intel_motion_estimation.txt。

法提供了 CPU 指令集级别的加速，对图形图像处理、摄像头相关软件开发和视频处理有很深的认识。我们参加 CVPR 会议的 Face Antispoof 官方源代码也存放在 libxcam 项目中。

我们团队的前成员李鹏为 OpenCV 深度学习模块提供了 OpenCL FP32 和 FP16 的加速方法。这里面的贡献包含了团队前成员龚志刚在 clCaffe 加速工作中的基本思想。

本书的贡献者还包括来自华中科技大学的张鹏、李雨霏和来自湖南大学的陈祎婧。

我们希望借这次撰写 OpenCV 深度学习书籍的机会和大家分享一下我们的技术积累，打破语言的壁垒，方便更多的国内开发者获利于开源、反馈开源，让开源开发方式可以更好地助力科技发展，创造更大的社会价值。

本书特色

本书由资深架构师利用业余时间倾情奉献，他们很多是 OpenCV 贡献者、算法的开发者。本书还特别结合硬件结构、优化的流程和方法，保证内容深度的同时注重实用性和可操作性，因此本书适合从初学者、高级工程师到架构师的各层面读者。

读者对象

- 图像视频处理架构师；
- 图像视频开发人员；
- 图像视频应用架构师；
- 深度学习应用开发人员；
- 深度学习算法工程师；
- 图像视频相关的管理人员；
- 其他对视频技术感兴趣的人员。

如何阅读本书

我们希望读者可以准备一个运行环境，一边运行代码，一边阅读本书。OpenCV 是一个开源的计算机视觉库，它的开发和维护都遵循开源软件的工作方式，而开源作为一种软件开发方式正在被越来越多的公司和组织所采用。我们希望读者在学习过程中能够积累一些开源软件的使用和调试技巧，迈出探索开源世界的第一步。下面是本书的脉络，供读者参考。

第 1 章

读者可以了解 OpenCV 背景、结构及使用方法。本章将介绍机器学习和深度学习的基础知识。落实到代码层面，读者可以看到具体的编译和运行 OpenCV 的方法，了解深度学习在 OpenCV 中的应用和编译使用方法。

1**第 2 章**

读者可以了解 OpenCV 深度学习模块的整体架构、实现原理、优化策略和加速框架，以及如何通过 Python 接口快速使用 OpenCV 深度学习模块。

2**第 3 ~ 5 章**

读者可以了解 OpenCV 深度学习模块的 GPU 加速原理。首先介绍并行计算的基础知识、Intel GPU 的架构特点，以及在深度学习加速实现中比较常用的 `cl_intel_subgroups` 技术在 Intel GPU 上的参考实现；然后详细讲解深度学习模块的基于 Vulkan 和 OpenCL 的加速实现。

3~5**第 6 章**

读者可以了解深度学习模块的 CPU 加速实现。我们从 CPU 的结构开始，逐渐深入到多线程加速和并行指令集加速。本章将介绍基于跨平台的第三方库的加速实现，如近几年学术界比较流行的 Halide 语言的加速实现及 Intel 的深度学习加速工具包 OpenVINO (Intel 推理引擎) 的加速实现。

6**第 7 章**

读者可以了解深度学习可视化工具 Netscope 和 TensorBoard 的使用方法。本章将详细介绍 Intel 平台性能分析工具 VTune 的使用和相关技巧。本章末尾给出性能调优的一般化建议，供读者参考。

7**第 8 章**

结合我们在 2019 年 CVPR 会议 Face Anti-spoofing 组的比赛中的案例，向读者展示一个完整的、基于深度学习技术项目的细节和开发过程。这个项目综合了人脸检测、活体检测、人脸识别和工程实现。

8**第 9 章**

读者可以了解深度学习方法在典型的计算机视觉任务中（图像分类、目标检测、语义分割和视觉风格变换）的应用。本章将结合深度学习模块的示例程序，从源代码和实际运行两个层面进行应用的讲解。

9

第 1 章首先介绍 OpenCV 的背景和基础，以及机器学习、深度学习的基础知识，在第 1 章结束时从一个简单对象分类的例子开始展开介绍如何使用 OpenCV 深度学习模块。

第 2 章主要介绍 OpenCV 深度学习模块的架构和实现原理。为了使读者更好地理解深度学习模块，第 2 章首先介绍深度学习的数学基础，然后从程序员的角度去解析深度学习的结构。接着从深度学习模块的分层架构展开，结合语言绑定及不同的正确性 / 性能测试

以更进一步地了解 OpenCV 深度学习模块。接下来从使用者的角度，介绍深度学习模块相关的函数接口、Layer 类、Net 类。然后深入地介绍 DNN 引擎的实现，内容包括模型导入、推理引擎内存分配，以及卷积、激活、池化、全连接等典型层类型的原理讲解。第 2 章还将介绍深度学习架构层面的优化方法，如层的融合、内存的复用等。最后介绍深度学习模块支持的各种加速方法和硬件设备，以及如何使用它们。

在理解了深度学习模块的架构和实现原理之后，我们继续从 GPU、CPU 及第三方库的角度深入探讨深度学习模块的加速实现。

第 3~5 章的主题是利用 GPU 的并行计算能力加速深度学习计算。第 3 章讨论并行计算的基础知识和 Intel GPU 的硬件结构。在此基础上，第 4 章和第 5 章将详细讲解深度学习模块的 OpenCL 和 Vulkan 加速。

第 6 章将讨论 CPU 的硬件知识，以及深度学习模块的 CPU 加速方法。读者可以了解到 OpenCV 中使用到的各种 CPU 加速的技巧，如指令集 SIMD 加速的具体方法。近些年，Halide 语言因其跨平台特点备受开发者的关注。OpenCV 也引入了 Halide 的加速方法，所以本书也向读者深入浅出地谈了 Halide，以及基于 Halide 的深度学习模块加速。Intel 的 OpenVINO 软件包提供了各种 Intel 硬件平台（包括 CPU、GPU、FPGA、Movidius）的深度学习加速实现，在 OpenCV 中也有引入，对应的是 Intel 推理引擎后端，该章也将做详细介绍。

第 7 章将介绍常用的深度神经网络可视化工具 TensorBoard（适用于 TensorFlow 网络格式）和 Netscope（适用于 Caffe 网络格式）。本章将详细讲解针对 Intel 硬件平台的性能分析和调优工具 VTune。第 7 章最后给出了高阶程序优化的思路和方法。

本书的另一个重点是应用实践。在第 8 章和第 9 章中，我们讲解基于深度学习的应用实践的具体细节，包括用深度学习方法处理计算机视觉的基本问题及一个完整的、实践性很强的人脸识别项目。

请看到本书的学生关注一下 Google Summer of Code 项目，之前很多年 OpenCV 社区就多次参加过，我所在的 Intel Media and Audio 团队已经作为导师组连续带了三届学生。学生们可领项目为开源社区做一些贡献。开源社区欢迎大家积极贡献代码，不管是提供新的例子，还是修掉某个小问题，或者提交新的功能。

勘误和支持

由于笔者水平有限，编写时间仓促，书中难免会出现一些错误或者不准确的地方，恳请读者批评指正。如果你有更多的宝贵意见，欢迎给我们发电子邮件，作者电子邮箱：

zhaojuanamy@163.com。你也可以通过机械工业出版社联系我们，期待能够得到你们的真挚反馈，在技术之路上互勉共进。本书配套资源可到 www.hzbook.com 本书所在页面下载。

致谢

特别感谢我们的领导傅文庆，一年半以来一直支持、鼓励我们，直到我们将这本书完成。

特别感谢来自湖南大学的陈祎婧、来自华中科技大学的张鹏。他们贡献了第 8 章，为读者展示了一个完整的端到端人脸识别应用的开发细节。

特别感谢来自华中科技大学的李雨霏利用自己的业余时间，贡献了语义分割部分的内容，并提出自己的见解，更好、更完善地向读者展示语义分割。

特别感谢我的同事宋瑞岭对本书的校验及在 OpenCL、GPU/CPU 方面的补充和讨论。

特别感谢我的同事付挺、傅林捷对初稿的仔细审阅。

尽信书不如无书，大家还是需要结合自己的平台来实验。

赵娟