

面向边缘GPU设备的快速光流估计算法

石珂 聂苏珍 李东兴 曹杰 盛云龙 姚斌 陈泓霖

Fast optical flow estimation algorithm for edge GPU devices

SHI Ke, NIE Suzhen, LI Dongxing, CAO Jie, SHENG Yunlong, YAO Bin, CHEN Honglin

引用本文:

石珂, 聂苏珍, 李东兴, 等. 面向边缘GPU设备的快速光流估计算法[J]. 应用光学, 2025, 46(2): 355–363. DOI: 10.5768/JAO202546.0202008

SHI Ke, NIE Suzhen, LI Dongxing, et al. Fast optical flow estimation algorithm for edge GPU devices[J]. *Journal of Applied Optics*, 2025, 46(2): 355–363. DOI: 10.5768/JAO202546.0202008

在线阅读 View online: <https://doi.org/10.5768/JAO202546.0202008>

您可能感兴趣的其他文章

Articles you may be interested in

基于嵌入式GPU的特征畸变单目视觉定位系统的研究

Monocular visual positioning system with characteristic distortion based on embedded GPU

应用光学. 2023, 44(3): 469–475 <https://doi.org/10.5768/JAO202344.0301001>

融合RANSAC光流跟踪法和特征点匹配法的视觉里程计

Design of visual odometer based on RANSAC optical flow method and feature point matching method

应用光学. 2020, 41(6): 1214–1221 <https://doi.org/10.5768/JAO202041.0602005>

基于嵌入式GPU的红外弱小目标检测算法

Infrared weak small target detection algorithm based on embedded GPU

应用光学. 2020, 41(5): 1089–1095 <https://doi.org/10.5768/JAO202041.0506004>

面向边缘智能光学感知的航空紧固件旋转检测

Aviation fastener rotation detection for intelligent optical perception with edge computing

应用光学. 2022, 43(3): 472–480 <https://doi.org/10.5768/JAO202243.0303002>

一种基于嵌入式全景视频流影像数据实时拼接方法

Real-time splicing method of image data based on embedded panoramic video stream

应用光学. 2023, 44(2): 354–364 <https://doi.org/10.5768/JAO202344.0202008>

液体颗粒测量技术鞘流系统的设计与仿真

Design and simulation of sheath flow system in liquid particle measurement

应用光学. 2024, 45(6): 1132–1137 <https://doi.org/10.5768/JAO202445.0601004>



关注微信公众号，获得更多资讯信息

文章编号: 1002-2082 (2025) 02-0355-09

引用格式: 石珂, 聂苏珍, 李东兴, 等. 面向边缘 GPU 设备的快速光流估计算法 [J]. 应用光学, 2025, 46(2): 355-363.
SHI Ke, NIE Suzhen, LI Dongxing, et al. Fast optical flow estimation algorithm for edge GPU devices[J]. Journal of Applied Optics, 2025, 46(2): 355-363.



在线阅读

面向边缘 GPU 设备的快速光流估计算法

石 珂¹, 聂苏珍¹, 李东兴¹, 曹 杰², 盛云龙¹, 姚 斌¹, 陈泓霖¹

(1. 山东理工大学 机械工程学院, 山东 淄博 255000; 2. 北京理工大学 光电学院, 北京 100081)

摘要: 提出了一种适用于边缘 GPU 设备的光流估计网络, 旨在解决稠密光流估计由于巨大计算量而难以在嵌入式系统上部署的问题。首先, 针对充分发挥 GPU 资源的需求, 设计了一个高效的特征提取网络, 以降低内存访问成本; 其次, 通过采用扁平形结构的迭代更新模块来估计光流, 进一步缩小了模型的尺寸, 并提升了 GPU 带宽的利用率。在不同数据集上的实验结果表明, 本文模型具备高效的推理能力和出色的光流估计能力。特别地, 与目前先进的轻量级模型相比, 所提出的模型在仅使用 0.54 Mb 参数的情况下, 误差减少了 12.8%, 推理速度提升了 22.2%, 在嵌入式开发板上展现出了令人满意的性能。

关键词: 光流估计; 嵌入式系统; 边缘 GPU 设备; 推理速度

中图分类号: TN202; TP391.41

文献标志码: A

DOI: [10.5768/JAO202546.0202008](https://doi.org/10.5768/JAO202546.0202008)

Fast optical flow estimation algorithm for edge GPU devices

SHI Ke¹, NIE Suzhen¹, LI Dongxing¹, CAO Jie², SHENG Yunlong¹, YAO Bin¹, CHEN Honglin¹

(1. School of Mechanical Engineering, Shandong University of Technology, Zibo 255000, China;

2. School of Optics and Photonics, Beijing Institute of Technology, Beijing 100081, China)

Abstract: An optical flow estimation network suitable for edge GPU devices was proposed, aiming to solve the problem that dense optical flow estimation was difficult to deploy on embedded systems due to huge quantity of computation. Firstly, to fully exploit the GPU resources, an efficient feature extraction network was designed to reduce memory access costs. Secondly, by adopting a flat-shaped iterative update module to estimate the optical flow, the size of the model was further reduced, and the utilization of GPU bandwidth was improved. Experimental results on different datasets show that the proposed model has efficient inference capability and excellent flow estimation performance. In particular, compared with the advanced lightweight models, the proposed model reduces the error by 12.8% with only 0.54 Mb parameters, and improves the inference speed by 22.2%, demonstrating the satisfactory performance on embedded development boards.

Key words: optical flow estimation; embedded systems; edge GPU devices; inference speed

引言

光流估计是计算机视觉领域中一个重要的研究问题, 在目标跟踪^[1]、视觉同步定位和映射

(SLAM)^[2]、语义分割^[3]等领域具有重要应用。随着深度学习的持续发展, 光流估计的准确性得到进一步的提升, 在精度和速度上都超过了传统方

收稿日期: 2023-11-23; 修回日期: 2024-01-30

基金项目: 国家自然科学基金 (62076152); 山东省自然科学基金青年基金 (ZR2022QF084); 山东省高等学校青年创新团队发展计划 (2022KJ234)

作者简介: 石珂 (1997—), 男, 硕士, 主要从事机器视觉研究。E-mail: 21501040040@stumail.sdu.edu.cn

通信作者: 李东兴 (1962—), 男, 博士, 教授, 主要从事成像探测与导引、智能仪器研究。E-mail: lidongxing@sdu.edu.cn

法,具有十分广阔的应用前景。DOSOVITSKIY A 等人^[4]提出的 FlowNet,通过使用简单的 U-Net^[5]架构,以细化的方式逐层学习光流,迈出了首个使用深度学习解决光流估计问题的重要的一步。为了解决 FlowNet 小位移预测不准确的问题,FlowNet2.0^[6]通过将 FlowNet 中的几个子模型(FlowNetS 和 FlowNetC)堆叠的办法来解决这一问题,进一步提高了光流估计精度。RANJAN A 等人^[7]提出了 SpyNet 网络,将经典空间金字塔和卷积神经网络相结合,使模型参数大大减少,运算速度得到提高,但是由于网络结构不够复杂,导致其估计精度较低。SUN D 等人^[8]提出了 PWC-Net 网络结构,其输入采用金字塔结构,利用多尺度特征来替换子网络串联,不仅减小了模型的大小,而且提高了模型的性能,小位移的光流也能有较好的估计精度。但是这些模型在粗略估计阶段时容易丢失快速运动的小物体,并且在细化阶段难以从错误估计中恢复。为了解决这一问题,TEED Z 等人^[9]提出仅用单一的高分辨率特征图,通过大量的迭代细化来逐步改善初始预测,从而在标准基准测试中展现了显著的性能。近年来,Transformer^[10]被证明可以应用于计算机视觉任务^[11]中,并且在大多数任务中取得了显著的效果。最近的一些工作^[12-14]侧重于利用注意力机制的方法进一步提高光流估计的精度,但是 Transformer 的二次计算复杂度严重制约了其推理速度,不利于实际应用。

将高性能的神经网络应用于边缘计算平台,例如无人机(UAV)和微型智能机器人,并且能够在实际应用环境下实时地运行,是人工智能的未来发展方向。目前已经提出了许多光流估计算法,但它们的目标大多都是在追求更高的精度,对于模型实际部署的研究相对较少。这是因为稠密光流估计需要大量且密集的计算,而移动端或者嵌入式设备这些平台的特点是内存资源少,处理器性能不高,功耗受限,这使得目前精度较高的模型根本无法在这些平台进行部署,并且难以达到实时运行的效果。

据调查所知,FastFlowNet^[15]是目前唯一一个专为应用于边缘设备而设计的模型,它在 PWC-Net^[8]框架内进行了轻量级改进。该模型可随时部署在嵌入式设备上,本文将其视为轻量级 SOTA(state of the art)模型,并进行了性能比较。自 RAFT(recurrent all-pairs field transforms)^[9]推出以来,几乎所

有的模型都将其作为创新改进的基准,间接证明了其结构的稳健性。按照相同的结构,本文开发了更高效的特征提取网络和轻量级的循环更新模块,由此组成的模型可以在保证准确性的同时大幅减少计算量,更好地满足低计算能力设备的要求。具体如下文。

1) 本文设计了一种低内存访问成本的特征提取网络。通过使用一种新颖的通道重排方法,将通道重新排序,解决了降低内存访问成本与扩张特征数量之间的矛盾。这大大降低了网络的内存访问成本,从而有利于实现高效的特征提取。

2) 本文开发了一个扁平形循环更新模块。通过对分支结构的优化,改变了特征提取的深度和广度,有效利用了 GPU 资源,加快了模型的推理速度。此外,该模块可以在较少的更新次数下,得到精确的光流估计结果。

3) 与同类模型相比,所提出的模型只需 0.54 Mb 参数,具有更小的终点误差 EPE (end-point-error),并在嵌入式设备上展现出了具有压倒性优势的推理速度。本文所提出的工作将缩短某些基于光流的视觉任务的推理时间。

1 方法原理

本文提出的光流估计模型 MiniFlow 如图 1 所示。网络分为 3 个模块:特征提取模块、视觉相似性计算模块、迭代更新模块。为了保证模型的准确性和推理速度,使用更加高效且轻量化的方法来搭建每一个模块,最终组成轻量级的光流估计网络 MiniFlow。

1.1 低内存访问成本网络

高质量的特征提取是得到精确光流估计的关键。为了能够高效地对两帧图像进行特征提取,设计了一个新颖的特征提取网络 LMAC-Net (low memory access cost net)。详细结构如图 2 所示。

通过实验发现,特征提取模块在模型的整个推理过程中占用了较多时间,因此本文希望重新设计一个特征提取网络,能够减少这部分的推理时间。ShuffleNetV2^[16]考虑了 GPU 访存带宽即内存访问成本 MAC(memory access cost)对于模型推理时间的影响。通过减少访存代价,网络能够更有效地利用 GPU 资源,从而提高推理速度。论文证明了当卷积的输入通道等于输出通道时,MAC 最小。本文考虑让所有卷积层都严格实现 MAC 最

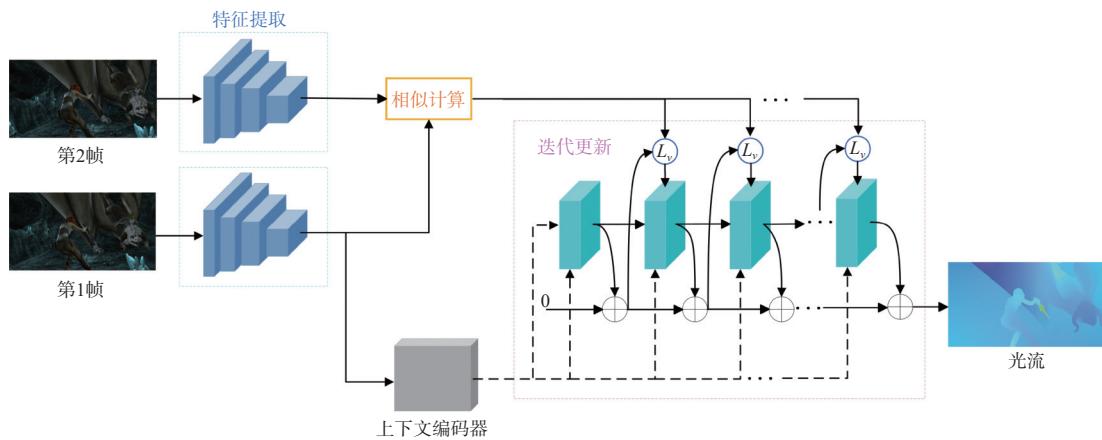


图 1 MiniFlow 光流估计神经网络模型结构

Fig. 1 Structure diagram of MiniFlow optical flow estimation neural network model

小, 即所有卷积层的输入通道数目等于输出通道数目。然而这存在一个明显的问题, 即输入的图片为 RGB(red-green-blue)三通道, 特征提取的目的是基于 RGB 这 3 个特征, 利用卷积层不断地学习到新的更多、更深层次的特征。在这个过程中, 通道数目不断增加, 即卷积层的输出通道数目是大于输入通道数目的。如何让卷积层既能够学习到更多的特征, 又能保持输入通道数目等于输出通道, 是搭建模型需要面对的难题。为了解决这个问题, 本文采用沿通道维度拼接的方式来扩充通道数目, 如图 2 所示。通过结合通道复制和通道拼接的方法, 可以实现所有卷积层的输出通道数目等于输入通道数目。

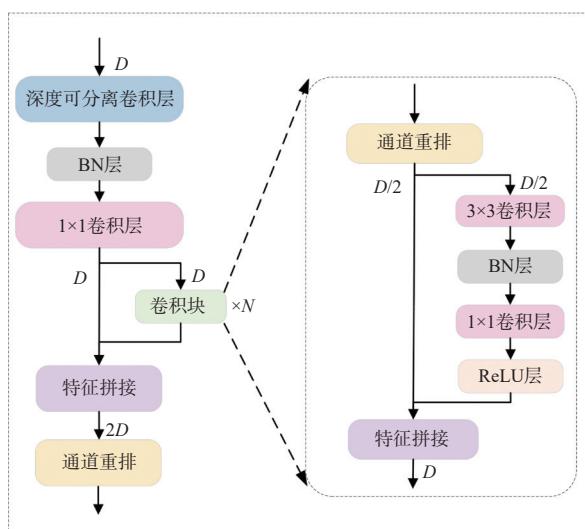


图 2 LMAC-Net 特征提取网络结构

Fig. 2 Structure of LMAC-Net feature extraction network

为提高特征的多样性和独立性, 更好地捕获输入数据中的各种特征, 本文提出了通道重排操作,

不同于 ShuffleNet 提出的 Shuffle 操作, 本文将相邻通道逐一分离, 并分到两组, 可以有效地降低近邻特征之间的相关性, 避免特征之间的冗余和混淆, 从而有助于提高网络的鲁棒性和泛化能力。本文使用分类数据集 Mini-Imagenet 单独训练特征提取网络。该网络利用了仅经过一次整体数据集训练的权重, 并对第 1 阶段的卷积块中激活函数的输出进行了可视化。为了清晰展示, 选取了前 16 个通道的输出进行可视化, 如图 3 所示。观察发现, 在训练的初期阶段, 左图邻近的特征图表现出了高度的相似性, 经过通道重排后, 可视化效果如右图所示。值得注意的是, 通道重排本身并不直接减少通道间的相似性。要实现降低特征图间形似性的效果, 需要结合网络的整体训练过程来实现。

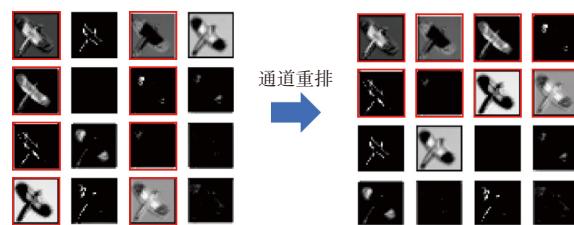


图 3 特征图经通道重排层后的可视化效果

Fig. 3 Visualization effects of feature maps after channel rearrangement layer

本文所提出的特征提取网络有以下特点: 首先, 步幅为 2 时, LMAC-Net 利用了跨阶段的短路连接, 可以融合深层和浅层的特征, 使得提取的特征更加丰富; 其次, 所有卷积的输入通道数目都等于输出通道数目, 这是为了使得内存访问成本(MAC)最低, 由此可使网络拥有更快的推理速度;

最后,所提出的通道重排的方法将近邻特征逐一分离,由此可以提高特征的表达能力,增强网络的表征学习能力。

通过上述模块,可以构建出特征提取网络 LMAC-Net。给定两帧连续的 RGB 图像 $I_1, I_2 \in \mathbb{R}^{H \times W \times D}$, 其中 H, W 和 D 表示高度、宽度和通道数目。通过特征提取器 LMAC-Net 后,得到 1/16 分辨率的稠密特征图 $f_\theta \in \mathbb{R}^{H/16 \times W/16 \times D}$, 其中 $D = 128$ 。

1.2 视觉相似性计算模块

在光流估计模型中,一般都是利用基于内积运算的相关层计算特征对应关系,其匹配过程如式(1)所示。

$$V_{ijmn} = \sum_h f_\theta(I_1)_{ijh} \cdot f_\theta(I_2)_{mnh}, V(f_\theta(I_1), f_\theta(I_2)) \in \mathbb{R}^{H \times W \times H \times W} \quad (1)$$

式中: $f_\theta(I_1)$ 与 $f_\theta(I_2)$ 分别为第 1 帧图像特征与第 2 帧图像特征; i, j, m, n 分别为第 1 帧与第 2 帧图像特征的高、宽索引; h 为图像特征的通道维度索引,由此构建出两帧图像间的全像素的匹配关系,即相关体积 V 。

为了恢复与大小运动位移相关的信息,创建了一个 3 层金字塔。这是通过一个核大小为 2×2 、步幅为 2 的池化层对相关体积的最后两个维度进行

三次迭代池化实现的。之后通过一个相关查找算子 L_v 在相关体积 V_l 中索引来生成特征映射,其中 l 代表金字塔索引。给定当前光流的估计值,可以预测出 I_1 中的每个像素 p 在 I_2 中的位置 p' , 定义 L_v 为在 p' 的 r 个单位半径内的整数偏移量的集合,即:

$$L_v = \sum_l \{p' + d | d \in \mathbb{Z}^2, \|d\|_\infty \leq r\} \quad (2)$$

为减轻计算量,本文将搜索半径 r 设置为固定值 3。

1.3 扁平形迭代更新模块

通过将新的特征重复送入更新模块进行迭代,可使最终得到的光流信息更加准确。在 RAFT-small 中,需要将这个模块循环 10 次以上,这会导致网络更深,严重影响模型的推理时间。因此需要设计一个轻量级的更新模块,能够在较低的循环次数下达到一个较好的光流估计结果。对于相同的计算资源和模型规模,宽而浅的网络通常具有更快的推理速度。这是因为较宽的网络可以同时处理多个特征通道,利用了 GPU 并行处理的优势,从而提高了推理效率。因此本文设计了一个扁平形的迭代更新模块 FIUM(flat iterative update module),其结构有着更少的分支、更低的计算复杂度和更少的参数,具体细节如图 4 所示。

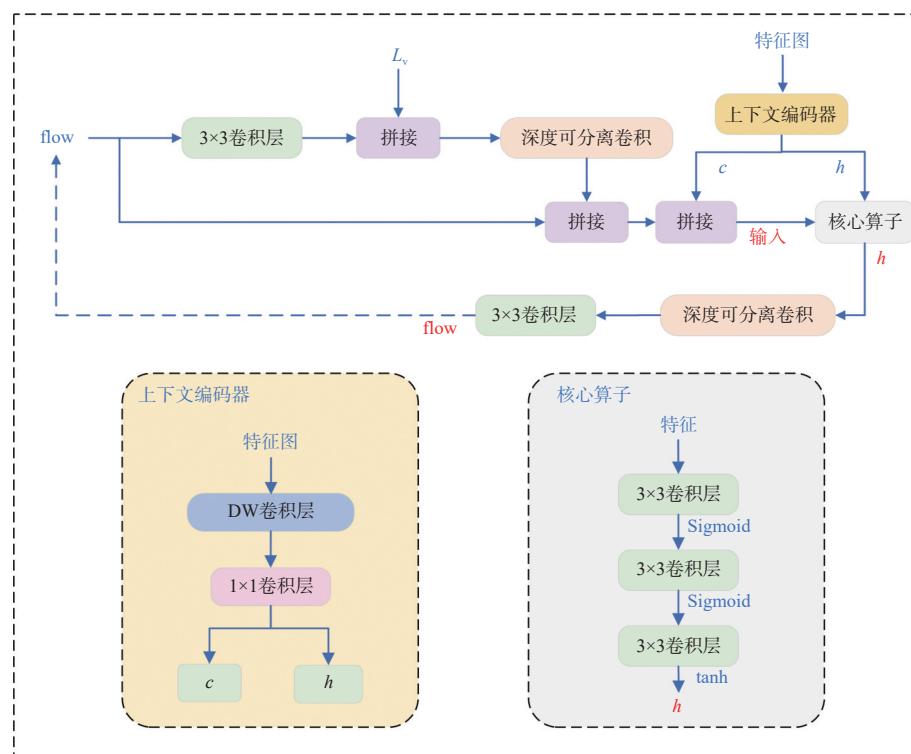


图 4 扁平形迭代更新模块结构图

Fig. 4 Structure diagram of flat-shaped iterative update module

上下文信息包含了原有图像的结构信息,对于光流估计有着十分重要的作用。与其他网络需要重新使用上下文编码器对第1帧图像提取上下文信息不同,本文通过上下文编码器提取第1帧图像的特征图来获得上下文信息,这可以缩短计算路径,提高推理速度。然后将获得的上下文信息沿通道维度均分为两份,一份作为常量 c 与相关查找算子、上一次的光流估计结果融合,另一份作为变量 h 在循环迭代中不断更新。

当循环次数降低时,基线模型 RAFT-small 的门控递归单元^[17](GRU)不再具有优势,并且其中的内积运算也会严重增加计算负担。本文通过串联3个卷积核大小为 3×3 的普通卷积作为核心算子(Core),整合相关查找算子、上一次的光流估计值、对第1帧图像提取特征得到的上下文信息,在每次迭代时都产生一个更新方向 Δf ,该方向应用于当前的流量估计 $f_{t+1} = f_t + \Delta f$ 。经过 n 次循环迭代输出预测的光流 $\{f_1, \dots, f_n\}$,此时估计的光流为 $1/16$ 全分辨率的光流,需要将其上采样至全分辨率才可以计算训练损失。本文通过权衡所预测光流的边缘平滑度和计算量,在每次迭代中,采用先用凸上采样^[9]将分辨率提升至 $1/2$,再用双线性插值上采样至全分辨率的方法,由此得到上采样后的预测结果 $\{\tilde{f}_1, \dots, \tilde{f}_n\}$ 。

1.4 损失函数

本文根据预测的光流 $\{\tilde{f}_1, \dots, \tilde{f}_n\}$ 和给定的真值 f_{gt} 之间的 L_1 距离来监督光流估计结果,损失函数为

$$\mathcal{L}_{epe} = \sum_{t=1}^n \alpha_{n-i} \|f_{gt} - \tilde{f}\|_1 \quad (3)$$

在训练阶段,设置 $\alpha = 0.8$,在微调阶段设置 $\alpha = 0.85$ 。

2 实验结果与分析

2.1 实验细节

本文使用 PyTorch 实现了所提模型,训练模型为两块 3090 GPU,优化器为 AdamW^[18]。根据之前相关工作的经验,模型依次在 FlyingChairs^[4] 和 FlyingThings^[19] 两个合成数据集上进行了训练,然后在 Sintel^[20] 和 KITTI-2015^[21] 训练集上对其进行了评估。首先在 FlyingChairs 上预训练 20 000 次,批量大小为 16,学习率为 $4e-4$,然后在 FlyingThings 上训练 400 000 次,批量大小为 8,学习率为 $1.2e-4$ 。通过结合 Sintel、KITTI-2015 和 HD1K^[22] 的数据,对 Sintel 进行另外 800 000 次微调,批量大小为 8,学习率为 $1.2e-4$ 。最后,模型在 KITTI-2015 上进行 100 000 次微调,批量大小为 6,学习率为 $1e-4$ 。

2.2 Sintel 和 KITTI 数据集上的性能测试

实验结果如表 1 所示。其中,速度根据 Sintel 分辨率图像(1024×436 像素)确定;括号中的数值代表网络其论文中所采用的微调策略实验的结果;FastFlowNet_v2 是 FastFlowNet 的等效版本,支持更高版本的 CUDA 和 PyTorch;“-ft”表示在 Sintel 或 KITTI 数据集上进行微调;“-N”表示递归更新模块更新 N 次;“*”为使用设备 CPU i5-8500、GPU GTX 1080 Ti 进行推理速度测量的结果,其他测试结果为原论文中同样采用 GTX 1080 Ti 推理测试的结果。

表 1 Sintel 和 KITTI 数据集上的性能比较

Table 1 Performance comparison on Sintel and KITTI datasets

方法	Sintel(train)		KITTI-15(train)		参数/Mb	时间/s
	Clean	Final	F1-epc	F1-all/%		
RAFT	1.43	2.71	5.04	17.4	4.80	0.327*
RAFT-ft	(0.76)	(1.22)	(0.63)	(1.5)	4.80	0.327*
FlowNet2	2.02	3.14	10.06	30.37	162.52	0.116
FlowNet2-ft	(1.45)	(2.01)	(2.30)	(8.61)	162.52	0.116
LiteFlowNet	2.48	4.04	10.39	28.50	5.37	0.055
LiteFlowNet-ft	(1.35)	(1.78)	(1.62)	(5.58)	5.37	0.055
SPyNet	4.12	5.57	—	—	1.20	0.050
SPyNet-ft	(3.17)	(4.32)	—	—	1.20	0.050
RAFT-small-10	2.21	3.38	9.55	28.67	0.99	0.041*
PWC-Net	2.55	3.93	10.35	33.67	8.75	0.034

表 1 (续)

方法	Sintel(train)		KITTI-15(train)		参数/Mb	时间/s
	Clean	Final	F1-epe	F1-all/%		
PWC-Net-ft	(2.02)	(2.08)	(2.16)	(9.80)	8.75	0.034
LiteFlowNetX	3.58	4.79	15.81	34.90	0.90	0.030
RAFT-small-4	2.89	4.22	11.77	34.23	0.99	0.027*
PWC-Net-small	2.83	4.08	—	—	4.08	0.024
FastFlowNet	2.89	4.14	12.24	33.10	1.37	0.011
FastFlowNet_v2	2.89	4.14	12.24	33.10	1.37	0.012*
FastFlowNet-ft	(2.08)	(2.71)	(2.13)	(8.21)	1.37	0.011
MiniFlow	2.52	3.77	12.98	39.73	0.54	0.009*
MiniFlow-ft	(1.46)	(1.88)	(1.47)	(6.09)	0.54	0.009*

在表 1 中, 未经微调的 MiniFlow 在 Sintel(Clean) 上的平均 EPE 为 2.52。图 5 展示了 MiniFlow 与其他模型的可视化对比结果, 可以观察到, 本文模型的可视化结果在边缘部分更加平滑。与在 GTX 1080 Ti 上推理速度低于 30 ms 的模型相比, MiniFlow 的误差与 FastFlowNet 和 PWC-Net-small^[8] 相比分别减少 12.8% 和 10.9%。与更新数

同为 4 的基线模型 RAFT-small 相比, MiniFlow 误差减少了 12.8%。在推理速度上, 该模型比 FastFlowNet 快 22.2%, 并且本文输出的是全分辨率的光流, 比 PWC-Net-small 快 166.7%, 相对于同样更新次数为 4 次的 RAFT-small, MiniFlow 推理速度提高了 200%, 并且精度表现同样优于上述模型。

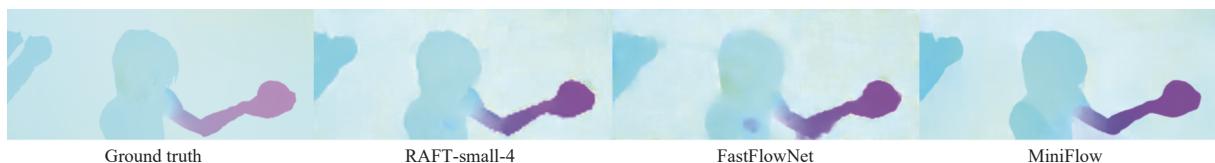


图 5 模型在 Sintel 数据集上的可视化结果

Fig. 5 Visualization results of model on Sintel dataset

在 FlyingChairs 和 FlyingThings 数据集上进行训练后, 同样在 KITTI 上对模型进行了评估。对模型微调使用了在 Sintel 上微调后的权重。与 Sintel 数据集相比, 模型在 KITTI 上的泛化效果稍差, 但经过微调后, F1-epe 和 F1-all 的值都显著降低。这可能是由于不同数据集之间存在差异。Flying-Chairs、FlyingThings 和 Sintel 都是虚拟场景, 因此训练和测试结果比较接近。KITTI 数据集是真实世界的场景, 其中的真值是通过人工进行了稀疏的标注, 可能会影响模型的泛化性能。不过, 该问题可以通过微调来改善。

2.3 嵌入式设备上的性能测试

在现有嵌入式设备发展方面, 市场上主流的嵌入式设备型号包括 Jetson 系列、Raspberry Pi 等。本文选择 Jetson Xavier NX 作为测试设备。首先, Jetson Xavier NX 拥有强大的计算能力和丰富的神经网络加速器支持, 能够满足深度学习模型对

于复杂计算的需求; 其次, Jetson Xavier NX 在嵌入式深度学习领域具有较高的知名度和广泛的应用^[23-24]。

本研究使用 PyTorch 框架对各个模型的推理速度进行了测试。测试所采用的代码基于 FastFlowNet 提供实现。首先, 为不同的模型在开发板上配置了适当的运行环境, 确保 PyTorch 版本和 CUDA 版本的兼容性; 然后, 将模型代码移植到开发板, 并加载相应权重; 最后, 使用 Sintel 数据集的图像作为输入, 在进行了 300 次推理实验后, 计算得出平均推理时间, 并将其转换为帧率。如表 2 所示, 在嵌入式设备 Jetson Xavier NX 上比较了 3 个模型的推理速度和误差结果, Jetson Xavier NX 采用了 10 W 的预设功耗模式。从表 2 可以看出 MiniFlow 推理速度可达到 12.57 frame/s, 比 RAFT-small-4 和 FastFlowNet_v2 分别快了近 5 倍和 3.5 倍。同时, MiniFlow 参数为 0.54 Mb, 比

RAFT-small-4 的参数小了近一半, 且不到 FastFlowNet_v2 的五分之二。可以看出, 与其他两个模型

相比, MiniFlow 在嵌入式开发板上具有明显的速度优势, 而且该模型的误差也比其他模型小。

表 2 RAFT-small、FastFlowNet_v2 和 MiniFlow 在 NX 上性能比较

Table 2 Performance comparison of RAFT-small, FastFlowNet_v2, and MiniFlow on NX

方法	Sintel (train)		速度/frame/s	参数/Mb
	Clean	Final		
RAFT-small-4	2.89	4.22	2.1	0.99
FastFlowNet_v2	2.89	4.14	2.77	1.37
MiniFlow	2.52	3.77	12.57	0.54

为了进一步验证 MiniFlow 的性能, 本文对比了 3 种模型在 DAVIS 数据集^[25]上的可视化结果, 如图 6 所示, 使用 1 080×1 920 像素的图像作为输入。

可以看出, MiniFlow 与其他两个模型相比, 在细节还原上表现得更好, 可视化结果最好, 这也证明了本文的模型具有良好的泛化能力。

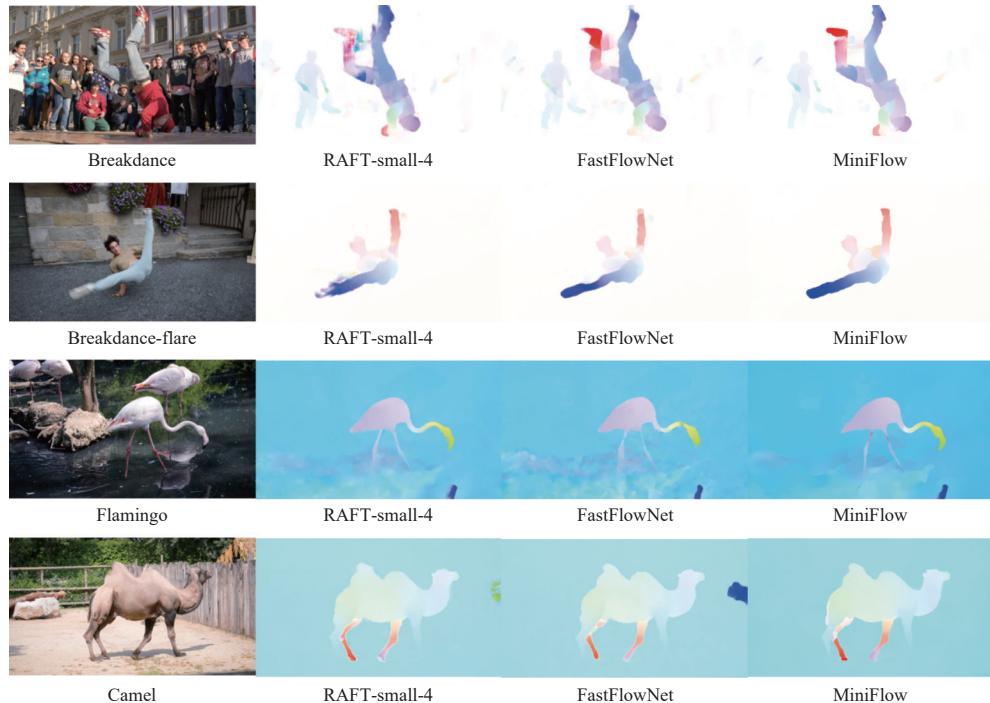


图 6 模型在 DAVIS 的 1 080×1 920 像素视频的可视化结果

Fig. 6 Visualization results of models on DAVIS 1 080×1 920 pixel video

2.4 消融实验

本文进行了两组消融实验, 以验证所提的特征提取和更新算子对整体结果的影响。所有模型先在

FlyingChairs 上训练 100 000 次, 然后在 FlyingThings 上训练 60 000 次, 最后在 Sintel 训练集上进行评估, 默认更新次数都为 4 次, 实验结果如表 3 所示。

表 3 不同模型在 Sintel 上的性能比较

Table 3 Performance comparison of different models on Sintel dataset

方法	Sintel (train)		时间/ms	参数/Mb
	Clean	Final		
<u>LMAC-Net</u>	3.10	4.18	8.7	0.54
ResNet	3.32	4.46	10	0.56
<u>FIUM</u>	3.10	4.18	8.7	0.54
ConvGRU	3.08	4.39	8.9	0.82

1) 特征提取。能够对图像特征进行有效提取是提高光流估计模型精度的关键之一。为了验证本文提出的特征提取网络 LMAC-Net 的有效性, 将模型中特征提取网络 LMAC-Net 替换为 ResNet 网络进行实验, 结果如表 3 所示。可以看到模型在误差、推理速度和模型大小方面都具有优势。

2) 循环更新算子。同样, ConvGRU 表示用卷积 GRU 替换扁平形迭代更新模块(FIUM)。下划线表示本文模型中使用的设置。从表 3 可以看出, 本文的方法在更具挑战性的 Sintel(Final) 上效果更好, 在 Sintel(Clean) 上的误差非常接近, 并且所提出的模型参数更少, 速度更快。

3 结论

本文提出了一种轻量级的光流估计网络 MiniFlow, 能够在低算力平台上实现快速、准确的光流估计。首先, 提出了一种低内存访问代价的特征提取网络, 能够高效地提取出高质量的特征; 其次, 设计了一种扁平形迭代更新模块, 能够合理利用计算资源, 可在较低的更新次数下有效进行光流估计。通过两种改进方法的结合, 大幅降低了模型参数, 提高了模型的推理速度。所提模型在不同基准数据集上均取得了令人满意的成绩, 特别在嵌入式设备上, 展现出了比目前的先进模型更出色的性能。在未来, 本文的工作有望使得一些基于光流的视觉任务实现更高的推理速度、更少的计算资源消耗。

参考文献:

- [1] MARVASTI-ZADEH S M, CHENG L, GHANEI-YAKHDAN H, et al. Deep learning for visual tracking: a comprehensive survey[J]. *IEEE Transactions on Intelligent Transportation Systems*, 2022, 23(5): 3943-3968.
- [2] TEED Z, DENG J. DROID-SLAM: deep visual slam for monocular, stereo, and RGB-D cameras[J]. *Advances in Neural Information Processing Systems*, 2021, 34: 16558-16569.
- [3] HU Y, HE Y, LI Y, et al. Efficient semantic segmentation by altering resolutions for compressed videos[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2023: 22627-22637.
- [4] DOSOVITSKIY A, FISCHER P, ILG E, et al. FlowNet: learning optical flow with convolutional networks[C]//Proceedings of the IEEE International Conference on Computer Vision. New York: IEEE, 2015: 2758-2766.
- [5] RONNEBERGER O, FISCHER P, BROX T. U-Net: convolutional networks for biomedical image segmentation[C]//Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Cham: Springer International Publishing, 2015: 234-241.
- [6] ILG E, MAYER N, SAIKIA T, et al. FlowNet 2.0: evolution of optical flow estimation with deep networks[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2017: 2462-2470.
- [7] RANJAN A, BLACK M J. Optical flow estimation using a spatial pyramid network[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2017: 4161-4170.
- [8] SUN D, YANG X, LIU M Y, et al. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2018: 8934-8943.
- [9] TEED Z, DENG J. RAFT: recurrent all-pairs field transforms for optical flow[C]//Computer Vision – ECCV 2020. Cham: Springer International Publishing, 2020: 402-419.
- [10] VASWANI A, SHAZER N, PARMAR N, et al. Attention is all you need[J]. *Advances in Neural Information Processing Systems*, 2017, 30: 5998-6008.
- [11] BROWN T, MANN B, RYDER N, et al. Language models are few-shot learners[J]. *Advances in Neural Information Processing Systems*, 2020, 33: 1877-1901.
- [12] JIANG S, CAMPBELL D, LU Y, et al. Learning to estimate hidden motions with global motion aggregation[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. New York: IEEE, 2021: 9772-9781.
- [13] XU H, ZHANG J, CAI J, et al. GMFlow: learning optical flow via global matching[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2022: 8121-8130.
- [14] HUANG Z, SHI X, ZHANG C, et al. FlowFormer: a transformer architecture for optical flow[C]// Computer Vision – ECCV 2022. Cham: Springer Nature Switzerland, 2022: 668-685.
- [15] KONG L, SHEN C, YANG J. FastFlowNet: a lightweight

- network for fast optical flow estimation[C]//2021 IEEE International Conference on Robotics and Automation (ICRA). New York: IEEE, 2021: 10310-10316.
- [16] MA N, ZHANG X, ZHENG H T, et al. ShuffleNet V2: practical guidelines for efficient CNN architecture design[C]//Proceedings of the European Conference on Computer Vision (ECCV). New York: IEEE, 2018: 116-131.
- [17] CHO K, VAN MERRIENBOER B, BAHDANAU D, et al. On the properties of neural machine translation: encoder-decoder approaches[EB/OL]. [2024-10-07]. <https://arxiv.org/pdf/1409.1259.pdf>.
- [18] LOSHCHILOV I, HUTTER F. Decoupled weight decay regularization[EB/OL]. [2019-05-10]. <https://arxiv.org/pdf/1711.05101.pdf>.
- [19] MAYER N, ILG E, HAUSSER P, et al. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2016: 4040-4048.
- [20] BUTLER D J, WULFF J, STANLEY G B, et al. A naturalistic open source movie for optical flow evaluation[C]//Computer Vision – ECCV 2012. Berlin, Heidelberg: Springer, 2012: 611-625.
- [21] MENZE M, GEIGER A. Object scene flow for autonomous vehicles[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. New York: IEEE, 2015: 3061-3070.
- [22] KONDERMANN D, NAIR R, HONAUER K, et al. The HCI benchmark suite: stereo and flow ground truth with uncertainties for urban autonomous driving[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. New York: IEEE, 2016: 19-28.
- [23] KUMAR A, KASHIYAMA T, MAEDA H, et al. Real-time citywide reconstruction of traffic flow from moving cameras on lightweight edge devices[J]. *ISPRS Journal of Photogrammetry and Remote Sensing*, 2022, 192: 115-129.
- [24] CHOE C, CHOE M, JUNG S. Run your 3D object detector on NVIDIA jetson platforms: a benchmark analysis[J]. *Sensors*, 2023, 23(8): 4005.
- [25] PONT-TUSSET J, PERAZZI F, CAELLES S, et al. The 2017 DAVIS challenge on video object segmentation [EB/OL]. [2028-03-28]. <https://arxiv.org/abs/1803.00557>.