智能系统学报
CAAI TRANSACTIONS ON INTELLIGENT SYSTEMS

结合卷积和轴注意力的光流估计网络

刘爽, 陈璟

引用本文:
刘爽, 陈璟. 结合卷积和轴注意力的光流估计网络[J]. 智能系统学报, 2024, 19(3): 575-583.
LIU Shuang, CHEN Jing. Optical flow estimation network combining convolution and axial attention[J]. *CAAI Transactions on Intelligent Systems*, 2024, 19(3): 575-583.

在线阅读 View online: https://dx.doi.org/10.11992/tis.202210029

DOI: 10.11992/tis.202210029
网络出版地址: https://link.cnki.net/urlid/23.1538.TP.20230914.1040.004

结合卷积和轴注意力的光流估计网络

刘 爽[1], 陈 璟[1,2]
（1. 江南大学 人工智能与计算机学院, 江苏 无锡 214122; 2. 江南大学 江苏省模式识别与计算智能工程实验室, 江苏 无锡 214122）

摘　要: 现有的光流估计网络为了获得更高的精度, 往往使用相关性成本量和门控循环单元 (gate recurrent unit, GRU) 来进行迭代优化, 但是这样会导致计算量大并限制了在边缘设备上的部署性能。为了实现更轻量的光流估计方法, 本文提出局部约束与局部扩张模块 (local constraint and local dilation module, LC-LD module), 通过结合卷积和一次轴注意力来替代自注意力, 以较低的计算量对每个匹配特征点周边区域内不同重要程度的关注, 生成更准确的相关性成本量, 进而降低迭代次数, 达到更轻量化的目的。其次, 提出了混洗凸优化上采样, 通过将分组卷积、混洗操作与凸优化上采样相结合, 在实现其参数数量降低的同时进一步提高精度。实验结果证明了该方法在保证高精度的同时, 运行效率显著提升, 具有较高的应用前景。

关键词: 光流估计; 迭代次数; 卷积神经网络; 轴注意力机制; 门控循环单元网络; 深度学习; 时间优化; 边缘计算平台
中图分类号: TP391　　**文献标志码:** A　　**文章编号:** 1673−4785(2024)03−0575−09

中文引用格式: 刘爽, 陈璟. 结合卷积和轴注意力的光流估计网络 [J]. 智能系统学报, 2024, 19(3): 575–583.
英文引用格式: LIU Shuang, CHEN Jing. Optical flow estimation network combining convolution and axial attention[J]. CAAI transactions on intelligent systems, 2024, 19(3): 575–583.

Optical flow estimation network combining convolution and axial attention

LIU Shuang[1], CHEN Jing[1,2]
(1. School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi 214122, China; 2. Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computing Intelligence, Jiangnan University, Wuxi 214122, China)

Abstract: Existing optical flow estimation networks often utilize correlation cost volume and gated recurrent unit (GRU) to realize iterative optimization for improved accuracy. However, this approach incurs high computational volume and limits deployment performance on edge computing platforms. To realize a lightweight optical flow estimation method, the local constraint and local dilation (LC-LD) module is introduced. This approach combines convolution and primary axis attention to replace self-attention. A low computational volume enables the module to realize attentions with different important degrees for peripheral areas of each matching feature point, generate an accurate correlation cost volume, further reduce the iterations, and achieve lightweight features. In addition, the shuffling convex optimization upsampling method is proposed. This technique combines group convolution, shuffle operation, and convex optimization upsampling, further increasing the precision and reducing the number of parameters. Experimental results show that the proposed method achieves significant improvements in running efficiency while maintaining high accuracy and great potential for application.

Keywords: optical flow estimation; iterations; convolutional neural networks; axial attention mechanism; gated recurrent unit network; deep learning; time optimization; edge computing platform

光流估计是计算机视觉领域中的一项基本任务, 被广泛应用于诸多领域, 如自动驾驶[1]、目标跟踪[2]、避障[3]、三维重建[4] 等。在早期的光流估计方法中, 基于亮度恒定假设和空间一致性假设的传统方法[5-8] 展现了它们的优势, 但传统方法在解决大位移、亮度不恒定等问题上存在固有缺陷, 使研究者不得不寻求新的方法来进行更加精

收稿日期: 2022−10−24.　**网络出版日期:** 2023−09−15.
基金项目: 江苏省青年科学基金项目（BK20150159）.
通信作者: 陈璟. E-mail: chenjing@jiangnan.edu.cn.

确的光流估计。随着相关设备的发展和深度学习的兴起，Dosovitskiy 等[9]将卷积神经网络引入光流估计领域，提出 FlowNet 模型，创造性地将 U 型网络运用到光流估计中，并使用 Encoder-Decoder 网络架构进行光流估计。FlowNet 设计了两个版本，分别为 FlowNetS 和 FlowNetC，后者相较于前者多使用了一个相关层，使两帧图片产生联系。FlowNet 的提出表明使用卷积神经网络可直接对相邻帧进行光流估计。同时伴随模型提出的还有用于光流训练的大型合成数据集 FlyingChairs，为使用卷积神经网络提供了数据保证。为了解决 FlowNet 小位移预测不准确的问题，Ilg 等[10]提出了采用串联多个 FlowNet 模型的方法 FlowNet2。尽管新模型的提出使光流估计变得更加鲁棒，但由于多个模块的堆叠，其参数量达到了 1.6×10^8。此后，更多的光流估计方法被相继提出。如，Ranjan 等[11]设计了一种新型空间金字塔网络模型 SpyNet，通过构建图像金字塔，并使用变形操作将第 2 帧图像映射为第 1 帧图像，由粗到细地估计增量光流，这一做法有效地改善了大位移光流预测精度不足的问题。与 FlowNet2 相比，SpyNet 的参数量大幅度降低，且速度更快。SpyNet 虽然精度相对较低，但它为后续众多基于金字塔结构的模型开辟了新的思路。Sun 等[12]提出的 PWC-Net 和 Hui 等[13]提出的 LiteFlowNet 模型，使用特征金字塔代替图像金字塔，并将相关层引入每层特征空间，获得了更好的对应表示。相较于 SpyNet、LiteFlowNet 模型在减少耗时的同时也进一步提升了精度。为了进一步降低模型的运行时间，Kong 等[14]又提出了使用头部增强的池化金字塔、中心密集空洞相关量等方法来构建更加轻量化的模型 FastFlowNet。为了在边缘计算平台获得更快的推理速度，FastFlowNet 同样选择了对精度进行一定的牺牲。因此如何获得一个推理速度和精度均优的光流估计模型依然值得进一步研究。

近几年来，为了获得更高的精度，Teed 等[15]提出 RAFT(recurrent all-pairs field Transformer)，将相邻帧图片所产生的特征图进行矩阵相乘，获得一个四维相关性成本量用于解码阶段的迭代查询。同时使用 GRU[16]对光流进行增量预测，在获得较高精度的同时，有效地改善了遮挡、大位移等问题。Jiang 等[17-18]为了进一步解决 RAFT 中缺少全局性移动特征和相关性成本量冗余等问题，分别提出 GMA 和 SCV。其中 GMA 使用 Transformer 对每次生成的移动特征进行全局聚合，更加有效地改善了遮挡问题。由于全局聚合存在于每一次迭代中，并且使用了计算量巨大的多头自注意力，因此需要耗费大量的时间。此外，近期基于 Transformer 的工作[19-23]在实现精度提升的同时，常常需要多个注意力模块进行叠加，极大增加了计算复杂度，导致整体结构变得更加繁琐。因此在光流估计领域，基于 Transformer 的光流估计网络仍可以进行更多探索。SCV 使用 K 近邻方法构建稀疏的相关性成本量，较好地减少了内存的使用，并且在训练时也减少了迭代次数。但是为了使用与 RAFT 相同的迭代网络，SCV 在迭代过程中需要耗费更多时间从稀疏的相关性成本量中进行相关性信息查询。

针对现有自注意力的高计算复杂度，本文利用轴注意力将自注意力分解为两个一维自注意力，并且与卷积进行结合，提出局部约束和局部扩张模块来增强特征。与上述使用 Transformer 方法相比避免了使用自注意力造成巨大的时间和内存消耗，增加了在边缘计算平台部署的可能性。与传统轴注意力模块通过迭代两次获得全局信息相比，本文通过使用一次轴注意力获取轴向信息并使用卷积对轴向信息进行约束和扩张，从而使用较少的计算量让每个特征点获得较大范围的周边关系信息，进一步减少运行时间。另外与 GMA 和 SCV 相比，并未在迭代过程中产生额外的时间消耗，而是采用“一次增强，多次迭代受用”的思想，将改善的重心放在生成更加鲁棒的相关性成本量上。使得每次查询可以获得更加精确的结果，且以更少的迭代获得相对较高的精度，更好的实现时间与精度的平衡，这也为轻量化模型提供了一种新的解决思路。

1　本文方法

本文为了进一步增加模型在边缘计算平台部署的可能性，使用 Teed 等[15]提出的 RAFT 小模型作为基线模型 (本文简称为 RAFT(small))，此模型将标准模型通道数降低一半，在保证一定精度的同时拥有更少的计算量。光流估计的主要任务为通过对网络模型进行学习，对给定的前后两帧图片 I_1和I_2，为I_1中每个像素(x_1,y_1)寻找一个移动向量(u,v)，令其与I_2中对应的像素

(x_2,y_2)相匹配。

本文网络总体框架如图 1 所示，主要分为 4 个部分：特征提取、特征增强、计算相关性成本量及迭代更新和混洗凸优化上采样。

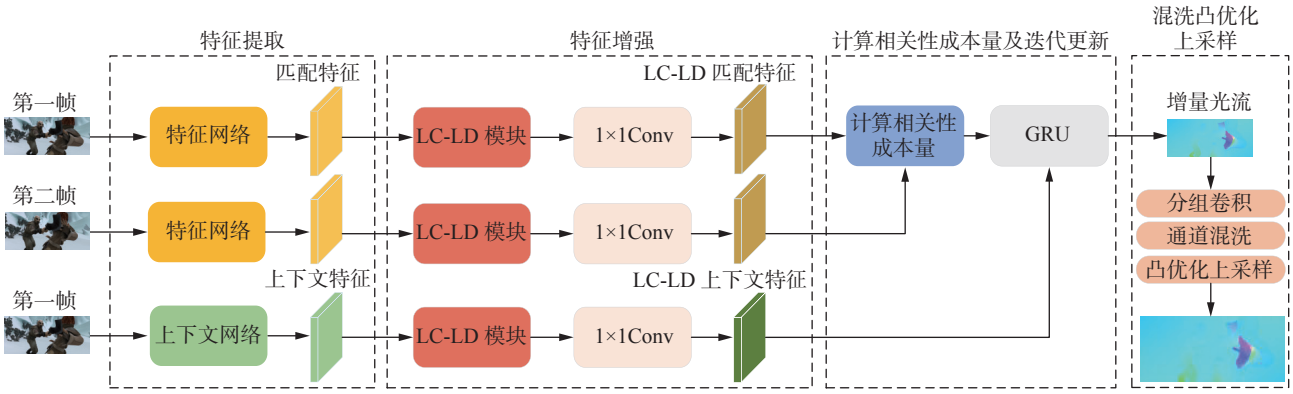

图 1　网络总体框架

Fig. 1　Overall architecture of the network

1.1　特征提取

对于给定的前后两帧图片$\boldsymbol{I}_1,\boldsymbol{I}_2\in\mathbf{R}^{\mathrm{H}\times\mathrm{W}\times3}$，通过一个特征网络，并且不进行权重共享，将其映射为$\boldsymbol{g}_1,\boldsymbol{g}_2\in\mathbf{R}^{H/8\times W/8\times C}$，其中$\boldsymbol{g}_1$、$\boldsymbol{g}_2$为匹配特征，分辨率为原始图片的 1/8，$C$为通道数，本文设置为 96。本文使用 RAFT(small) 中的特征网络，主要由 1 个大卷积层和 6 个瓶颈层组成，前两个瓶颈层分辨率为原图的 1/2，中间两个瓶颈层分辨率为原图的 1/4，最后两个瓶颈层分辨率为原图的 1/8。此外使用一个上下文网络对图片I_1进行上下文特征提取，将其映射为$\boldsymbol{c}_1\in\mathbf{R}^{H/8\times W/8\times C}$，其中$C$为通道数，本文设置为 96。

1.2　特征增强

RAFT 通过矩阵相乘来计算基于匹配特征的相关性成本量，这将用于后期的迭代查询。因此，相关性成本量的准确性决定了是否可以使用较少的迭代次数来获得更优的精度。由于相关性成本量在计算后会成为一个定值，不会进行学习，因此匹配特征的精确程度直接决定了相关性成本量的准确性。为了获得更精确的匹配特征，本文通过结合具有获取局部性信息能力的卷积和拥有较少计算量的轴注意力，提出 (局部约束−局部扩张)LC-LD 模块，代替传统自注意力模块对匹配特征进行进一步增强。LC-LD模块如图 2(a)所示，由 4 个子模块构成，分别为 LC 模块、轴注意力模块、LD 模块和特征融合模块。

1.2.1　注意力模块

视觉注意力机制是人类视觉的一种特有的大脑信号处理机制，通过快速扫描全局图像来筛选出需要重点关注的目标区域，提高视觉信息处理的效率和准确性。深度学习中的注意力机制与人类的选择性视觉注意力机制类似，目标是从大量信息中选择出对当前任务目标最关键的信息。对于传统卷积来说，同样可以将其看作为一种注意力单元，通过不断地融合特征的周边关系信息获取重要的信息。但是由于卷积的局部性，当每次进行卷积操作时，只能获取每个特征点周边较小范围的关系，因此想要获得非局部的长期依赖关系，卷积将不再适用。为了获得全局感受野，周海赟等[24]使用了空洞卷积。但空洞卷积在使用时需要谨慎选择膨胀率，否则卷积将会对填充的 0 进行学习，而非有效的空间区域。此外，使用多头自注意力的视觉 Transformer 方法已经被证明其有效性，但由于使用自注意力时需要进行多次高维的矩阵乘法运算，其计算复杂度为$O(H^2W^2)$，将会消耗大量的计算资源，不利于将带有自注意力机制模块的模型移植到 CUDA 核心数更少的边缘计算平台。因此如何降低自注意力的计算量成为本文思考的重点。鉴于自注意力模块的高计算复杂度，使用轴注意力对其进行分解成为了一个较好的解决方案。轴注意力会在特征图的水平方向和垂直方向分别产生自注意力，两个自注意力模块串行连接，可以有效模拟原有的自注意力机制，并且在保证精度的前提下，有效降低计算复杂度至$O(HW^2+H^2W)$，具有更好的计算效率。此外通过加入前馈神经网络对特征进行进一步转换和通过与原始拥有局部性信息的特征进行融合，进一步增强模块对重要信息的获取能力，构建多头轴注意力模块。如图 2(b) 所示，在特征拥有局部性信息的基础上进一步获取特征的全局性信息。通过实验，使用多头轴注意力模块可以有效的提升精度。

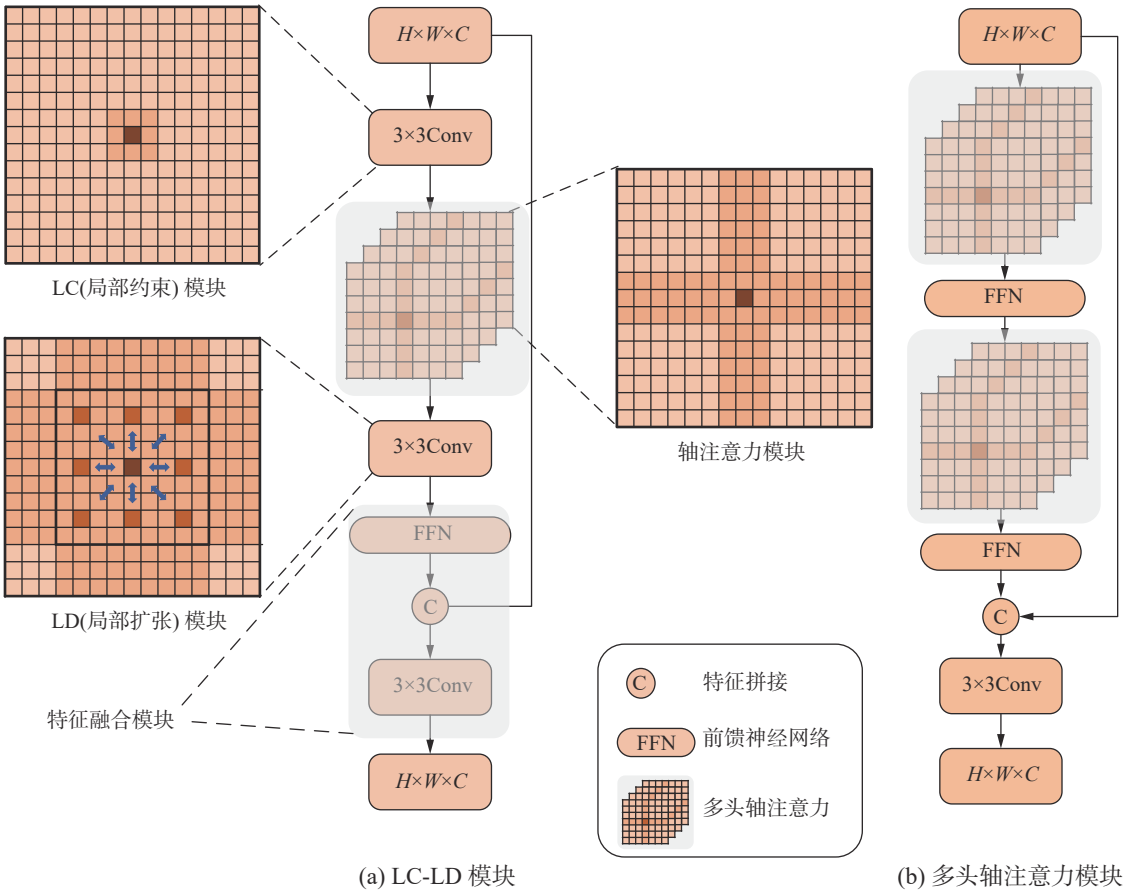

图 2　轴注意力模块对比

Fig. 2　Comparison of axis attention modules

卷积可以通过多次运算生成不同的特征图，从而学习不同的特征。而轴注意力却没有这样的特性。为了让轴注意力拥有关注不同方面信息的能力，本文使用头数为 4 的多头轴注意力，获取不同方面的信息。轴注意力模块可以表示为

$$\boldsymbol{F}_A = \mathrm{T}_{\mathrm{col}}(\mathrm{T}_{\mathrm{row}}(\boldsymbol{F}_C)) \tag{1}$$

式中：$T_{\mathrm{col}}(\cdot)$表示垂直轴注意力，$\mathrm{T}_{\mathrm{row}}(\cdot)$表示水平轴注意力，$\boldsymbol{F}_C$表示 LC 特征，$\boldsymbol{F}_A$表示轴注意力特征。

1.2.2　LC 模块和 LD 模块

轴注意力通过将原始自注意力分解为两个方向的自注意力而实现计算量的进一步降低。但是运行一次轴注意力只能获得每个特征点的轴向信息（水平和垂直方向信息）。为了获取全局性信息，多头轴注意力模块中需要迭代两次轴注意力，从而造成了进一步的时间消耗。如何进一步降低时间消耗成为本文思考的关键。本文受 Li 等[25] 提出 CoTNet 启发，同样使用一个 3×3 卷积来获取特征的上下文信息。但本文并未与 CoTNet 一样仅对 Key 使用卷积操作获取其上下文信息，而是直接在轴注意力计算 Query、Key、Value 之前就对原始特征使用卷积操作，让其自行的去融合特征的局部性信息，扩大后期每个特征进行轴注意力时的感知范围。如图 2(b) 所示，当每个像素点进行局部约束模块之后进入轴注意力模块时，实质对该像素点和其周围像素组成的局部块进行注意，可以使用 LC 模块对其进行约束，较好地改善因只使用注意力而造成泛化性不足的问题。对于给定的匹配特征$\boldsymbol{F} \in \mathbf{R}^{H\times W\times C}$，通过 LC 模块可以表示为$\boldsymbol{F}_C \in \mathbf{R}^{H\times W\times C}$。

如图 2(b) 所示，进行一次轴注意力只能让每个像素点获得轴向信息，通过 LC 模块让其感知单位变为局部块，但是与多头轴注意力模块通过两次轴注意力获取到的全局性信息相比，在面对平均光流长度较长的数据集时，依然存在不足。因此本文通过融合周围同样拥有轴向信息的像素点，进一步弥补因只使用一次轴注意力而造成感受野范围缺失的问题。对于得到的轴注意力特征$\boldsymbol{F}_D \in \mathbf{R}^{H\times W\times C}$，通过 LD 模块可以表示为$\boldsymbol{F}_D \in \mathbf{R}^{H\times W\times C}$。LC-LD 模块可以表示为

$$\boldsymbol{F}_{\text{LC-LD}} = \mathrm{Conv}_{3\times 3}([\mathrm{FFN}(\boldsymbol{F}_D), \boldsymbol{F}]) \tag{2}$$

式中：$\boldsymbol{F}_D$表示 LD 特征，FFN(·)表示前馈神经网络，$\boldsymbol{F}$表示初始特征，[·]表示特征拼接，$\mathrm{Conv}_{3\times 3}(\cdot)$表示 3×3 卷积，$\boldsymbol{F}_{\text{LC-LD}}$表示最终获取到的局部约束与局部扩张特征。

1.3　计算相关性成本量及迭代更新

对于给定的前后两帧图片I_1、I_2，分别通过特征网络模块和特征增强模块生成 LC-LD 匹配特征$\boldsymbol{G}_1$、$\boldsymbol{G}_2$。此外，使用I_1通过上下文网络和特征增强模块，生成 LC-LD 上下文特征$\boldsymbol{C}_1$。通过对$\boldsymbol{G}_1$、$\boldsymbol{G}_2$进行矩阵相乘，得到 LC-LD 四维相关性成本量$\boldsymbol{C}_{\text{orr}}$：

$$\boldsymbol{C}_{\text{orr}}(i,j,p,q)=\boldsymbol{G}_1(i,j)^{\text{T}}\cdot\boldsymbol{G}_2(p,q) \tag{3}$$

式中：$\boldsymbol{G}_1(i,j)$，$\boldsymbol{G}_2(p,q)$分别表示 LC-LD 匹配特征$\boldsymbol{G}_1$和$\boldsymbol{G}_2$中任意一点像素，$\boldsymbol{C}_{\text{orr}}(i,j,p,q)$表示四维的 LC-LD 相关性成本量。

本文使用与 RAFT(small) 中相同的 GRU 更新模块，通过迭代计算增量光流。GRU 更新模块的输入与 RAFT(small) GRU 更新模块不同之处在于，上下文特征变为拥有周边范围信息的 LC-LD 上下文特征$\boldsymbol{C}_1$，用于每次迭代查找的相关性成本量变为更加精确的 LC-LD 相关性成本量。

1.4　混洗凸优化上采样

RAFT 标准模型中采用凸优化上采样对低分辨率光流进行上采样，相比于双线性插值，可以在边界获得更加平滑的效果，但是需要使用大量的参数。为了进一步减少模型参数量，本文将其中拥有 256 个通道的 3×3 卷积替换为组数为 8 的分组卷积，并将通道数降为 192，同时借鉴 ShuffleNet[26] 的思想，对各组之间的特征图进行打乱，进一步保证精度。

1.5　损失函数

与 RAFT 相同，本文同样使用一个残差网络架构，通过每次 GRU 迭代来预测增量光流Δf_i，每一次迭代之后，光流预测为$f_{i+1}=f_i+\Delta f_{i+1}$，初始光流为$f_0=0$。本文使用 L1 损失，对于给定的光流真值$f_{\text{gt}}$和每次迭代之后的预测光流$f_i$：

$$L=\sum_{i=1}^{N}\gamma^{N-i}\left\|f_i-f_{\text{gt}}\right\|_1 \tag{4}$$

式中：本文权重γ设置为 0.8，N为迭代次数，此处与 RAFT 迭代 12 次有所不同，本文模型在训练时，迭代次数设置为 6。

2　实验结果与分析

2.1　数据集

为了评估本文模型的有效性，使用 FlyingChairs、FlyingThings3D[27]、MPI-Sintel[28] 和 KITTI 2015[29] 等 4 个数据集进行训练和评估。

FlyingChairs 和 FlyingThings3D 均为大型合成数据集。FlyingChairs 由 22 872 对图像组成，FlyingThings3D 包含 19 635 个图像。本文主要使用 FlyingChairs 和 FlyingThings3D 训练集对网络进行预训练。MPI-Sintel 由 Clean 和 Final 两个版本数据集构成，训练集各 1 041 张图像，测试集各 552 张图像，其中 Final 版本包含运动模糊、大气变化、雾效果和噪点，比 Clean 版本更加复杂，且更具挑战性。KITTI 2015 为真实道路场景数据集，包含稀疏光流真值，但图像较少，训练集和测试集均只有 200 对图像。

2.2　实验环境及训练细节

本文模型在 PyTorch 框架下构建，使用一张 2080TiGPU 对其进行训练，并且选择 AdamW 优化器对网络进行优化。与 RAFT 标准模型训练步骤相似，首先在 FlyingChairs 和 FlyingThings3D 上进行预训练，然后在 MPI-Sintel 上进行微调，在 FlyingChairs 上训练时，批量大小设置为 10，迭代次数设置为 10 万次，学习率为 0.000 4，图片裁剪大小为 352 像素×480 像素。在 FlyingThings3D 上训练时，批量大小设置为 6，迭代次数设置为 12 万次，学习率为 0.000 125，图片裁剪大小为 400 像素×720 像素。最后在 MPI-Sintel 训练集上进行微调，批量大小设置为 6，学习率为 0.000 4，图片裁剪大小为 368 像素×768 像素。

2.3　性能指标

时间指标：本文采用模型在 Jetson AGX Xavier 边缘计算平台前向推断一对 MPI-Sintel 图像 (436 像素×1 024 像素) 所需的时间作为时间指标。此外为了避免时间统计存在误差，本文采用前向推断 1 000 组图像取均值的方式进行时间统计。

精度指标：对于 MPI-Sintel 数据集，本文采用终点误差 (end point error, EPE) 作为精度指标，其表示同一个像素点的预测光流矢量$(\boldsymbol{u}_1,\boldsymbol{v}_1)$与真实光流矢量$(\boldsymbol{u}_0,\boldsymbol{v}_0)$之间的距离：

$$E_{\text{EP}}=\sqrt{(\boldsymbol{u}_1-\boldsymbol{u}_0)^2+(\boldsymbol{v}_1-\boldsymbol{v}_0)^2} \tag{5}$$

对于 KITTI 2015 数据集，本文采用终点误差和 F1-all 两种作为精度指标，其中 F1-all 表示光流估计错误像素点所占的百分比。

2.4　消融实验

本小节将通过一系列消融实验来验证本文所提方法的有效性。如表 1 所示，分别将每个子模块在 FlyingChairs 和 FlyingThings3D 训练集上进行训练，同时保持其他子模块的设置与最终模型一致，使用 MPI-Sintel 和 KITTI 2015 训练集对模型效果进行评估，最终模型使用的设置将用下划线加以区分。

如表 1 所示，迭代次数为前向推断时所使用的迭代次数，Baseline 为 RAFT(small) 公开的网络权重前向推断阶段迭代 12 次所获得的精度，而其他实验均为前向推断迭代 6 次获得的精度。LC-LD 为本文所提出模型，仅需迭代 6 次即可在 MPI-Sintel 和 KITTI 2015 训练集上超越 Baseline。LC 和 LD 模型：LC 模块可以有效地提升 MPI-Sintel 训练集上的精度，LD 模块可以有效地提升光流平均模长较长的 KITTI 2015 训练集上的精度。Multi-head模型：表示多头轴注意力的头数，当头数 head 设置为 4 时，可以在 MPI-Sintel 和 KITTI 2015 训练集均获得较佳的精度。Attention 模型：使用 Swin Transformer[30] 模块代替 LC-LD 注意力模块中的多头轴注意力，多头轴注意力可以获得更高的精度，并且在 KITTI 2015 训练集上获得更好的泛化性能。Upsampling 模型：与 Convex(凸优化上采样) 相比，本文所提出 Shuffle(混洗凸优化上采样) 在参数量降低的同时，可以进一步提升精度。Axis Attention Module 模型：使用 Multi-head Axis Attention Module 和 Ccnet[31] 中轴注意力模块分别替换本文所提出的 LC-LD 模块，最终，LC-LD 模型可以获得更优的精度。

表 1　消融实验结果

Table 1　Ablation experiment results

实验模型	变量方法	MPI-Sintel(训练)		KITTI 2015(训练)		参数量/10^6	迭代次数	运行时间/ms (Xavier)
		Clean	Final	EPE	F1-all/%			
Baseline	—	2.19	3.38	8.46	26.68	0.99	12	339
LC-LD	w/o	2.27	3.53	10.62	32.32	1.15	6	219
	with	2.01	3.31	7.18	26.60	2.31	6	275
LC	w/o	2.05	3.49	7.57	26.88	2.14	6	271
	with	2.01	3.31	7.18	26.60	2.31	6	275
LD	w/o	2.08	3.29	7.40	27.13	2.14	6	271
	with	2.01	3.31	7.18	26.60	2.31	6	275
Multi-head	1	2.10	3.27	7.92	28.34	2.01	6	256
	2	2.10	3.45	7.44	26.40	2.11	6	262
	4	2.01	3.31	7.18	26.60	2.31	6	275
	6	2.04	3.50	7.59	26.67	2.50	6	293
Attention	Swin	2.12	3.46	10.52	29.38	1.98	6	290
	Axis	2.01	3.31	7.18	26.60	2.31	6	275
Upsampling	Convex	2.09	3.38	8.03	27.51	2.51	6	271
	Shuffle	2.01	3.31	7.18	26.60	2.31	6	275
Axis Attention Module	Multi-head Axis Attention Module	2.12	3.43	8.70	28.59	2.45	6	306
	Ccnet[31]	2.32	3.64	8.24	29.47	1.20	6	235
	LC-LD	2.01	3.31	7.18	26.60	2.31	6	275

注：下划线表示本文最终选择的变量方法。

2.5　实验结果与对比

2.5.1　MPI-Sintel 训练集

使用 2.2 节中参数设置，对网络模型进行训练如表 2 所示，RAFT(small) 为基线模型，train 表示经过 FlyingChairs 和 FlyingThings3D 训练之后在 MPI-Sintel 训练集上所获得的结果，test 表示使用上述模型在 MPI-Sintel 训练集上进行微调，在 MPI-Sintel 测试集上所获得的结果。与基线模型 RAFT(small) 进行对比，本文所提模型仅需迭代 12 次，即可在 MPI-Sintel 测试集上超越基线模型迭代 32 次的精度，并且在边缘计算平台前向推断时间降低 46%。

与其他轻量化方法进行对比，本文所提方法在时间和在 MPI-Sintel 测试集上的精度上均超越了 FlowNet2、SpyNet 和 LiteFlowNet。与 LiteFlowNet2[32] 相比，使用本文迭代 6 次的模型 LC-LDs，可以达到与其相近的时间，并且在 MPI-Sintel 和 KITTI 2015 训练集上可以取得更优的精度，比其具有更好的泛化性能。此外，相较于 FlowNetC 和 PWC-Net，LC-LDs 以使用较少的时间代价，实现了精度的全面提升。与 FastFlowNet 相比，由于无法完全避免使用 GRU 进行迭代

优化，因此为了获得一定的精度，依然需要一定次数的迭代，从而会消耗相对较多的时间。综合上述实验，本文模型可以更好地实现精度与时间的平衡。图 3 给出了 LC-LD 与 RAFT(small) 网络模型在 MPI-Sintel(Final) 测试集上的可视化对比。

表 2　MPI-Sintel 和 KITTI 2015 训练集上的定量评估

Table 2　Quantitative evaluation on MPI-Sintel datasets and KITTI 2015 datasets

方法	MPI-Sintel (Clean)		MPI-Sintel (Final)		KITTI 2015		迭代次数	运行时间/ms (Xavier)
	train	test	train	test	train			
	(EPE)	(EPE)	(EPE)	(EPE)	(EPE)	(F1-all/%)		
FlowNetC[9]	3.57	6.08	5.25	7.88	—	—	—	191
FlowNet2[10]	2.02	4.16	**3.14**	5.74	10.06	28.20	—	805
SpyNet[11]	4.12	6.64	5.57	8.36	—	—	—	601
PWC-Net[12]	2.55	3.86	3.93	5.13	10.35	33.67	—	208
LiteFlowNet[13]	2.48	4.54	4.04	5.38	10.39	28.50	—	595
LiteFlowNet2[32]	2.24	3.45	3.78	4.90	9.83	25.88	—	282
FastFlowNet[14]	2.89	4.89	4.14	6.08	12.24	33.10	—	**126**
RAFT(small)[15]	2.13	3.51	3.28	4.50	7.49	**25.27**	32	782
FlowFormer(small)[22]	**1.20**	—	**2.64**	—	**4.57**	**16.62**	—	2631
LC-LD	2.00	**3.31**	3.28	**4.34**	7.06	26.16	12	417
LC-LDs	2.01	3.67	3.31	4.72	7.18	26.60	**6**	275

(a) RAFT(small)

(b) LC-LD

(c) Ground truth

图 3　在 MPI-Sintel(Final) 测试集上的可视化比较

Fig. 3　Visual comparison on MPI-Sintel (Final) test set

与精度较高的 FlowFomer(small) 进行对比，虽然原文作者对其进行了一定的轻量化，由于其对四维相关性成本量进行编码，耗费了大量计算量，固然可以获得较高的精度，但是相较于本文方法运行时间依然存在较高的消耗，无法适用于边缘计算平台。

2.5.2　KITTI 2015

本文使用 KITTI 2015 训练集作为测试集，验证本文所提模型在真实场景数据集上的泛化性能。使用 2.2 节中参数设置，在 FlyingChairs 和 FlyingThings3D 数据集上进行训练，然后在 KITTI 2015 训练集上进行效果评估。如表 2 所示，与 RAFT(small) 进行对比，本文模型迭代 12 次即可在 KITTI 2015 训练集上获得与 RAFT(small) 迭代 32 次相近的精度。与其他光流估计方法进行对比，本文模型均获得了较好的精度。通过在具有挑战性的真实场景数据集上进行测试，证明本文所提模型在使用较少迭代次数时依然可以获得较好的泛化性能。

2.6　时间优化

使用本文迭代 3 次的模型前向推断一对 MPI-Sintel 图像的速率可以达到 4.9 f/s。为了进一步

降低模型在 Xavier 平台上的前向推断时间，将本文所提模型的数据类型从 float32 变换为 float16，这使得模型在边缘计算平台运行效率提升的同时 (速率提升为 8 f/s)，模型精度并不会产生太大的变化。如表 3 所示，与运行时间最少的轻量化光流估计模型 FastFlowNet 进行比较，前向推断时间相近时，本文模型在 MPI-Sintel 和 KITTI 2015 训练集上的精度均获得了提升。同时通过测试，FastFlowNet 若使用相同的方式变换数据类型为 float16，时间反而上升，因此在表 3 中 FastFlowNet 表示模型在 float32 数据类型下的前向推断时间 (速率为 7.9 f/s)。

表 3　时间优化定量评估

Table 3　Quantitative evaluation of time optimization

方法	Sintel (train)		KITTI 2015 (train)		运行时间/ms (Xavier)
	Clean	Final	EPE	F1-all/%	
FastFlowNet[14]	2.890	4.140	12.240	33.100	126
LC-LD(float32)	**2.384**	**3.853**	8.214	**29.111**	205
LC-LD(float16)	2.386	3.857	**8.213**	29.125	**125**

3　结束语

本文针对现有光流估计模型迭代次数较多的问题，提出 LC-LD 模块来增强相关性成本量的方法，降低了 GRU 迭代次数。通过将具有局部表示能力的卷积和具有轴向表示能力的轴注意力进行结合，有效提升了视觉特征的表现能力，改善了传统轴注意力需要迭代两次方可获得较大感受野的不足，缓解了因只使用轴注意力而导致泛化性差的问题。同时提出混洗凸优化上采样，进一步降低模型的参数量。在 MPI-Sintel 和 KITTI 2015 数据集上的测试结果证明了所设计网络的有效性。在未来的工作中将会继续探索本文模型使用更少的迭代，且进一步提升其在 MPI-Sintel 测试集上的精度。

参考文献：

[1] 李志慧, 胡永利, 赵永华, 等. 基于车载的运动行人区域估计方法 [J]. 吉林大学学报(工学版), 2018, 48(3): 694–703.
LI Zhihui, HU Yongli, ZHAO Yonghua, et al. Locating moving pedestrian from running vehicle[J]. Journal of Jilin University (engineering and technology edition), 2018, 48(3): 694–703.

[2] 陈戈, 董明明. 基于特征点检测与光流法的运动目标跟踪算法 [J]. 电子测量技术, 2017, 40(12): 214–219.
CHEN Ge, DONG Mingming. Moving object tracking algorithm based on feature point detection and optical flow[J]. Electronic measurement technology, 2017, 40(12): 214–219.

[3] SOUHILA K, KARIM A. Optical flow based robot obstacle avoidance[J]. International journal of advanced robotic systems, 2007, 4(1): 2.

[4] 李秀智, 杨爱林, 秦宝岭, 等. 基于光流反馈的单目视觉三维重建 [J]. 光学学报, 2015, 35(5): 515001.
LI Xiuzhi, YANG Ailin, QIN Baoling, et al. Monocular camera three dimensional reconstruction based on optical flow feedback[J]. Acta optica sinica, 2015, 35(5): 515001.

[5] HORN B K P, SCHUNCK B G. Determining optical flow[J]. Artificial intelligence, 1981, 17(4): 185–203.

[6] BROX T, MALIK J. Large displacement optical flow: descriptor matching in variational motion estimation[J]. IEEE transactions on pattern analysis and machine intelligence, 2011, 33(3): 500–513.

[7] ZACH C, POCK T, BISCHOF H. A duality based approach for realtime TV-L 1 optical flow[C]//Joint Pattern Recognition Symposium. Berlin: Springer, 2007: 214−223.

[8] KROEGER T, TIMOFTE R, DAI Dengxin, et al. Fast optical flow using dense inverse search[C]//European Conference on Computer Vision. Cham: Springer, 2016: 471−488.

[9] DOSOVITSKIY A, FISCHER P, ILG E, et al. FlowNet: learning optical flow with convolutional networks[C]// 2015 IEEE International Conference on Computer Vision. Santiago: IEEE, 2015: 2758−2766.

[10] ILG E, MAYER N, SAIKIA T, et al. FlowNet 2.0: evolution of optical flow estimation with deep networks[C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 1647−1655.

[11] RANJAN A, BLACK M J. Optical flow estimation using a spatial pyramid network[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017: 2720−2729.

[12] SUN Deqing, YANG Xiaodong, LIU Mingyu, et al. PWC-net: CNNs for optical flow using pyramid, warping, and cost volume[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 8934−8943.

[13] HUI T W, TANG Xiaoou, LOY C C. LiteFlowNet: a lightweight convolutional neural network for optical flow estimation[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 8981−8989.

[14] KONG Lingtong, SHEN Chunhua, YANG Jie. FastFlowNet: a lightweight network for fast optical flow estimation[C]//2021 IEEE International Conference on Robotics and Automation. Xi'an: IEEE, 2021: 10310−10316.

[15] TEED Z, DENG Jia. RAFT: recurrent all-pairs field transforms for optical flow[C]//European Conference on Computer Vision. Cham: Springer, 2020: 402−419.

[16] CHO K, VAN MERRIENBOER B, BAHDANAU D, et al. On the properties of neural machine translation: encoder-decoder approaches[EB/OL]. (2014−09−03)[2022−10−24]. http://arxiv.org/abs/1409.1259.

[17] JIANG Shihao, CAMPBELL D, LU Yao, et al. Learning to estimate hidden motions with global motion aggregation[C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 9752−9761.

[18] JIANG Shihao, LU Yao, LI Hongdong, et al. Learning optical flow from a few matches[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021: 16587−16595.

[19] JAEGLE A, BORGEAUD S, ALAYRAC J B, et al. Perceiver IO: a general architecture for structured inputs & outputs[EB/OL]. (2021−07−30)[2022−10−24]. http://arxiv.org/abs/2107.14795.

[20] XU Haofei, ZHANG Jing, CAI Jianfei, et al. Unifying flow, stereo and depth estimation[EB/OL]. (2022−11−10)[2023−10−24]. http://arxiv.org/abs/2211.05783.

[21] SUI Xiuchao, LI Shaohua, GENG Xue, et al. CRAFT: cross-attentional flow transformer for robust optical flow[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 17581−17590.

[22] HUANG Zhaoyang, SHI Xiaoyu, ZHANG Chao, et al. FlowFormer: a transformer architecture for optical flow[EB/OL]. (2022−03−30)[2022−10−24]. http://arxiv.org/abs/2203.16194.

[23] ZHAO Shiyu, ZHAO Long, ZHANG Zhixing, et al. Global matching with overlapping attention for optical flow estimation[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022: 17571−17580.

[24] 周海赟, 项学智, 翟明亮, 等. 结合注意力机制的深度学习光流网络 [J]. 计算机科学与探索, 2020, 14(11): 1920–1929.
ZHOU Haiyun, XIANG Xuezhi, ZHAI Mingliang, et al. Deep optical flow learning networks combined with attention mechanism[J]. Journal of frontiers of computer science and technology, 2020, 14(11): 1920–1929.

[25] LI Yehao, YAO Ting, PAN Yingwei, et al. Contextual transformer networks for visual recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2023, 45(2): 1489–1500.

[26] ZHANG Xiangyu, ZHOU Xinyu, LIN Mengxiao, et al. ShuffleNet: an extremely efficient convolutional neural network for mobile devices[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 6848−6856.

[27] MAYER N, ILG E, HÄUSSER P, et al. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 4040−4048.

[28] BUTLER D J, WULFF J, STANLEY G B, et al. A naturalistic open source movie for optical flow evaluation[C]//European Conference on Computer Vision. Berlin: Springer, 2012: 611−625.

[29] MENZE M, GEIGER A. Object scene flow for autonomous vehicles[C]//2015 IEEE Conference on Computer Vision and Pattern Recognition. Boston: IEEE, 2015: 3061−3070.

[30] LIU Ze, LIN Yutong, CAO Yue, et al. Swin transformer: hierarchical vision transformer using shifted windows[C]//2021 IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 9992−10002.

[31] HUANG Zilong, WANG Xinggang, HUANG Lichao, et al. CCNet: criss-cross attention for semantic segmentation[C]//2019 IEEE/CVF International Conference on Computer Vision. Seoul: IEEE, 2019: 603−612.

[32] HUI T W, TANG Xiaoou, LOY C C. A lightweight optical flow CNN-revisiting data fidelity and regularization[J]. IEEE transactions on pattern analysis and machine intelligence, 2021, 43(8): 2555–2569.

作者简介：

刘爽，硕士研究生，主要研究方向为计算机视觉。E-mail：2430393663@qq.com。

陈璟，博士，副教授，主要研究方向为生物信息学、计算机视觉。主持江苏省青年基金项目 1 项，参加国家自然基金项目 3 项，获得省部级奖 4 项，申请发明专利 13 个，授权发明专利 4 个，发表学术论文 20 余篇。E-mail：chenjing@jiangnan.edu.cn。