文章编号:1007－6735(2025)04－0438－11　DOI: 10.13255/j.cnki.jusst.20240606001

结合卷积和交叉变换网络的光流估计

温远斌，　黄影平，　李瀚灵
（上海理工大学 光电信息与计算机工程学院，上海 200093）

摘要：光流估计是计算机视觉领域的核心任务。近年来，基于深度学习的光流估计方法取得了显著进展。但光流算法容易受到图像遮挡、重叠物体和相似物体的影响，从而导致运动估计准确性下降。为了解决这一问题，提出了一种结合卷积和交叉变换网络的光流估计模型。首先，利用卷积和交叉注意力机制对连续两帧图像进行特征提取与增强。其次，结合上下文特征网络编码提供的上下文信息，通过特征聚合模块产生聚合特征，使用 ConvGRU 模块进行光流的迭代更新，并通过上采样模块产生光流特征估计图，完成光流估计任务。最后，进行 MPI-Sintel 和 KITTI 2015 数据集的综合实验对比分析。实验结果表明，所提出的方法可以有效地提高光流估计的精度，并具有良好的泛化性。

关键词：光流估计；卷积神经网络；交叉变换网络；注意力机制；凸优化上采样
中图分类号：TP 391.4　　**文献标志码：**A

Convolution and cross transformer network for optical flow estimation

WEN Yuanbin,　HUANG Yingping,　LI Hanling
(*School of Optical−Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China*)

Abstract: Optical flow estimation is a core task in the field of computer vision. In recent years, deep learning based optical flow estimation methods have made significant progress. However, optical flow algorithms are easily affected by image occlusion, overlapping objects and similar objects, leading to a decrease in the accuracy of motion estimation. To overcome this problem, an optical flow estimation model combining convolutional and cross transformer network was proposed. Firstly, feature extraction and enhancement were performed on two consecutive frames of images using convolution and cross

收稿日期: 2024−06−06
基金项目: 国家自然科学基金资助项目 (62276167); 上海市自然科学基金资助项目 (20ZR1437900)
第一作者: 温远斌（2000—），男，硕士研究生．研究方向：深度学习、计算机视觉．E-mail：222190436@st.usst.edu.cn
通信作者: 黄影平（1966—），男，教授．研究方向：汽车电子系统、计算机视觉等．E-mail：huangyingping@usst.edu.cn

引文格式: 温远斌，黄影平，李瀚灵．结合卷积和交叉变换网络的光流估计[J]．上海理工大学学报，2025，47(4)：438-448.
Citation: WEN Yuanbin, HUANG Yingping, LI Hanling. Convolution and cross transformer network for optical flow estimation[J]. Journal of University of Shanghai for Science and Technology, 2025, 47(4)：438-448.

attention mechanisms. Secondly, combining the contextual information provided by the context feature network encoding, the feature aggregation module was used to generate aggregated features, and the ConvGRU module was used to iteratively update the optical flow. The optical flow feature estimation maps were generated through the upsampling module to complete the optical flow estimation task. Finally, a comprehensive experimental comparative analysis was conducted on the MPI Sintel and KITTI 2015 datasets. The experimental results show that the proposed method could effectively improve the accuracy of optical flow estimation and had good generalization ability.

Keywords: *optical flow estimation*; *convolutional neural network*; *cross transformer network*; *attention mechanism*; *convex optimization upsampling*

光流是一种二维矢量场，用于捕捉图像序列中运动目标的运动轨迹和速度。除了记录物体的运动和相邻帧之间的位移信息外，光流还携带着有关运动目标形状和结构的丰富信息。光流估计是计算机视觉领域的一个重要问题。它通过分析图像序列中相邻帧之间像素的亮度变化，以估算物体的运动。光流估计在许多领域具有广泛的应用，包括目标跟踪[1]、动作识别[2]、运动检测[3]和机器导航[4]等。

由于卷积神经网络强大的表征能力和大量训练数据的涌现，基于深度学习的方法在光流估计领域中逐渐成为主流。注意力机制能够使模型在处理输入数据时聚焦于重要的区域，从而提高模型对关键信息的感知能力。Transformer 模型[5]作为一种基于自注意力机制的架构，能够在输入过程中有效地捕获数据序列中的长程依赖关系。因为在光流估计中，相邻帧之间的像素位移通常是连续和平滑的，建模长程依赖关系有助于捕捉这种连续性，使得光流估计更加准确。通过引入 Transformer 架构，光流估计模型可以更好地处理运动目标之间的相互作用，以及动态场景中的运动规律。

当前，光流估计在处理图像中的遮挡、重叠物体或相似物体时，容易发生误判，进而影响运动估计的准确性。针对此问题，本文提出了一种结合卷积和交叉变换网络的光流估计模型。所设计的注意力机制将特征块内自注意力与特征块间自注意力相互交叉应用。其中，特征块内自注意力属于局部注意力范畴，而特征块间自注意力则属于全局注意力范畴。本文工作的主要贡献如下：a. 以 Cross Attention Transformer 网络为基础构架，提出了一种结合卷积和交叉变换网络的光流估计模型。b. 设计了一种特征聚合模块，该模块基于参考帧的自相似性计算注意力矩阵，并聚合特征，可有效提高光流估计的精确度和鲁棒性。c. 在 MPI-Sintel 数据集[6]和 KITTI 2015 数据集[7]上进行了大量的实验验证。实验结果表明，该方法与同类模型相比，在性能上具有一定优势。

1　相关工作

1.1　经典光流估计方法

经典光流估计方法大致分为 3 种方法：a. 基于亮度变化的光流算法。这类算法基于图像中的亮度变化来推断像素之间的运动。它们假设相邻帧之间的亮度值在空间和时间上是连续变化的，然后利用亮度梯度来估计像素的运动。b. 基于特征匹配的光流算法。这类算法首先检测图像中的特征点（如角点、边缘点等），然后利用特征点之间的匹配关系来计算它们的运动。c. 基于能量最小化的光流算法。这类算法将光流估计问题转化为一个能量最小化问题，并利用优化算法（如变分法、最小二乘法等）来求解。例如，Lucas-Kanade 等[8]提出了一种基于局部图像亮度一致性的迭代优化方法，该方法可以在图像中搜索相应点的位置，并通过不断迭代优化来逐步提高匹配的准确性。Horn 等[9]使用可变框架将光流表述为连续优化问题，并通过执行梯度步进来估计密集流场，将亮度恒定性假设形式化为一个能量函数，然后使用变分法来求解这个能量函数的最小值，从而得到光流场的估计。Farnebäck 等[10]的贡献在于引入了一种基于多项式展开的新颖的双帧运动估计方法，为计算机视觉领域提供了一种新的思路。在两帧图像之间，物体的运动模式可以近似为一

个多项式，通过多项式展开来建模两帧图像之间的像素值变化，并通过优化来确定多项式的系数，从而得到像素级别的运动估计。然而，在处理运动目标出现大位移或图像场景亮度变化不均匀时，经典方法仍然存在固有的缺陷。这些问题导致了光流估计的不准确性和不稳定性。因此，研究者不得不寻求新的方法来克服这些困难，以实现更加精确和可靠的光流估计。

1.2 基于卷积神经网络的光流估计

近年来，随着深度学习技术的不断发展，基于卷积神经网络的方法已经开始在光流估计研究中得到广泛应用。光流估计模型的训练通常需要大量带有真实值的数据，但现有的光流数据集，如MPI-Sintel、KITTI和Middlebury[11]等，往往无法提供足够的数据来支持深度学习模型的训练。因此，为了解决这一问题，研究者开始依赖于大规模人工合成的预训练数据集，其中包括FlyingChairs[12]和FlyingThings3D[13]等。Dosovitskiy等[12]提出了一个基于卷积神经网络的光流估计模型FlowNet，该模型创新地运用编码和解码的网络架构。这项研究验证了卷积神经网络在光流估计任务中的显著优势，并通过创建大规模的合成数据集FlyingChairs，为以后的光流任务提供了充足的数据支持。随后，Ilg等[14]提出了改进版本FlowNet2，通过迭代堆叠进一步提高了光流估计精度，但是这增加了模型参数和计算成本。

为了解决计算成本过大的问题，在后续研究中，许多模型都将多尺度处理策略引入深度学习架构中。例如，Ranjan等[15]提出了一种名为SpyNet的创新性空间金字塔网络模型，其采用了自粗到细的光流估计策略，从而显著简化了网络结构。虽然该模型的精度较低，但为基于金字塔网络的光流估计任务提供了新的探索方向。例如，Sun等[16]提出的PWC-Net模型和Hui等[17]提出的LiteFlowNet模型，均采用金字塔网络处理图像特征，并在金字塔结构的每一层上构建了Warp层和匹配代价层，从而获得更好的对应表示。为了进一步提高模型的运行速度，Kong等[18]提出了FastFlowNet，采用了头部增强的池化金字塔方法构建轻量化模型，以实现快速准确的光流预测。FastFlowNet遵循了广泛采用的从粗到细的残差结构，并从金字塔中提取特征，但在精度方面仍然不够。因此，在光流估计任务中，如何平衡运行速度和精度，仍然值得进一步研究。

大多数成功的光流架构设计都是通过运用更好的特征编码和解码设计来实现的。例如，Kong等[19]提出了一种基于迭代优化和遮挡检测的金字塔网络设计方案。Teed等[20]提出了RAFT网络模型，该方法采用多尺度全局4D匹配代价层和循环迭代光流解码器这种新颖架构，并利用门控单元GRU[21]对光流进行增量预测，不仅在提高精度方面表现良好，而且有效地解决了遮挡和大位移等问题。

1.3 基于注意力机制的光流估计

深度学习中的注意力机制是一种模仿人类视觉和认知系统的方法，它允许神经网络在处理输入数据时集中注意力于相关的部分。通过引入注意力机制，神经网络能够自动地学习并选择性地关注输入中的重要信息，提高模型的性能和泛化能力。注意力机制通常应用于对序列数据（如文本、语音或图像序列）的处理。其中，最典型的注意力机制包括自注意力机制[5]、空间注意力机制[22]和通道注意力机制[23]。这些注意力机制允许模型对输入序列的不同位置分配不同的权重，以便在处理每个序列元素时专注于最相关的部分。

在自然语言处理领域，以注意力机制为基础的网络架构Transformer已取得巨大成功。相比于传统卷积结构，Transformer架构可以提取出不同尺度的特征，获得更为丰富的特征信息，这启发了视觉任务中自我关注的发展。随后，基于Transformer的架构被引入到许多其他视觉任务中，如检测[24]、点云处理[25]、图像恢复[26]、视频修复[27]等，通常也获得了不错的效果。这种卓越性通常被归因于Transformer能够有效地建模远程关系，这也是光流估计所需的特性之一。在RAFT的基础上，Jiang等[28]提出全局运动聚合单元模块，根据查询和关键特征计算出的注意矩阵用于聚合作为运动隐藏表示的值特征，聚合后的运动特征与局部运动特征以及背景特征连接起来，由门控单元GRU解码，较好地解决了图像遮挡区域的光流估计问题。Huang等[29]提出了一种新颖的基于Transformer的神经网络架构FlowFormer，有效地将代价信息聚合得更加紧凑，通过动态位置代价查询反复解码代价特征，这一处理将在特征空间上的探索推向极致。为了减少计算成本，Chu等[30]提出了一种空间可分离的自注意层，通过在平面

上排列令牌来传播信息。尽管改进的 Transformer 光流方法在精确性和泛化性方面取得了显著进展，但现有方法过于专注光流解码器的改进，而忽视了特征编码器对光流估计精度优化的影响。这导致了全局移动特征的部分缺失以及相关成本量之间存在冗余的问题。因此，需要更多的研究关注特征编码器的优化，以实现更全面和有效的光流估计。本文结合了卷积和交叉变换网络来对特征进行提取与增强，以实现特征编码。所设计的注意力机制将特征块内自注意力与特征块间自注意力相互交叉应用。这一设计能够更有效地捕获整体场景的信息，从而有助于应对物体位置变化和局部遮挡等挑战。因此，该模型能够提高对复杂场景的理解和处理能力。具体介绍见 3.3。

2　研究方法

2.1　方法概述

本文采用的网络框架如图 1 所示。所提出的网络模型能够实现端到端的学习过程，即在输入图像序列后，经过模型计算后能够直接输出光流估计结果。如图 1 中红色虚线所示，本文网络模型主要由以下 4 个部分组成：特征提取、相关量计算、特征增强、光流更新与上采样。

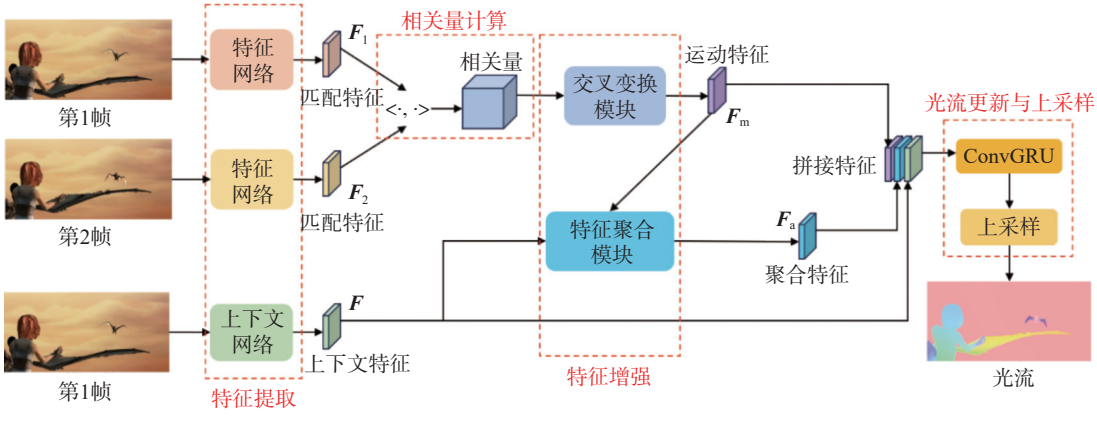

图 1　网络架构

Fig.1　The network architecture

首先，在特征网络中利用卷积神经网络对连续两帧图像进行特征提取，并将产生的匹配特征 $\boldsymbol{F}_1$ 和 $\boldsymbol{F}_2$ 通过矩阵点积操作形成特征相关量。其次，特征相关量通过交叉变换模块得到运动特征 $\boldsymbol{F}_{\mathrm{m}}$，此时该特征具有更好的局部与全局特征信息。另外，第 1 帧图像被送入上下文特征网络进行编码得到上下文特征 $\boldsymbol{F}_{\mathrm{c}}$，并在特征聚合模块中，结合特征 $\boldsymbol{F}_{\mathrm{c}}$ 与 $\boldsymbol{F}_{\mathrm{m}}$ 产生聚合特征 $\boldsymbol{F}_{\mathrm{a}}$，以进一步提升对图像间运动的感知和理解。随后，拼接 $\boldsymbol{F}_{\mathrm{m}}$、$\boldsymbol{F}_{\mathrm{a}}$ 和 $\boldsymbol{F}_{\mathrm{c}}$ 这 3 个特征，通过 ConvGRU 模块进行迭代更新处理。最后，在上采样模块中，将特征图扩展到与原始图像相同的尺寸，并产生光流特征估计图，从而完成光流估计任务。

2.2　特征提取

特征提取阶段分为匹配特征编码和上下文特征编码。对于前后连续的两帧图像 $\boldsymbol{I}_1$ 和 $\boldsymbol{I}_2 \in \mathbb{R}^{H\times W\times 3}$，其中 H 和 W 分别表示输入特征图的高和宽。匹配特征编码器采用卷积神经网络从输入图像中提取特征，并以 1/8 分辨率输出特征 $\boldsymbol{F}_1$ 和 $\boldsymbol{F}_2 : \mathbb{R}^{H\times W\times 3} \mapsto \mathbb{R}^{(H/8)\times(W/8)\times D}$，其中 D 为通道数，设置为 96。该编码器核心结构包括首层 7×7 的卷积操作，输出通道数为 32，后接 ReLU 激活函数以引入非线性映射。同时根据所选的归一化策略选择适当的归一化层，以加速网络收敛并提高训练稳定性。随后是 3 个残差块，每个块都采用不同的输入通道数和步幅，以增强特征的表征能力。此外，为了缓解过拟合现象，模型还支持可选的 Dropout 层。最终，通过 1×1 的卷积操作，将特征映射转换为所需的最终输出维度。同时，上下文特征编码器对图片 $\boldsymbol{I}_1$ 进行上下文特征提取，与匹配特征编码器结构一样，但不互相共享权重，其输出特征为 $\boldsymbol{F}_{\mathrm{c}} \in \mathbb{R}^{(H/8)\times(W/8)\times D}$，其中 D 为通道数，设置为 96。匹配特征网络和上下文网络共同构成了本文方法的第 1 阶段，只需要执行一次。

2.3　相关量计算

本文通过在所给定的图像对之间构建一个完整的相关量来计算视觉相似性。对于给定的前后两帧图片 $\boldsymbol{I}_1$ 和 $\boldsymbol{I}_2$，分别通过特征提取模块，生成匹

配特征$\boldsymbol{F}_1 \in \mathbb{R}^{H\times W\times D}$和$\boldsymbol{F}_2 \in \mathbb{R}^{H\times W\times D}$。取所有特征向量对之间的像素点进行矩阵点积，得到四维相关量M：

$$M(\boldsymbol{F}_1,\boldsymbol{F}_2)\in \mathbb{R}^{H\times W\times H\times W} \tag{1}$$

2.4 特征增强

2.4.1 交叉变换模块

为了获得更精确的匹配特征，受文献 [31] 的启发，本文设计了一种交叉变换模块，该模块将特征块内注意力与特征块间注意力相互交叉应用。前者是在图像特征块的内部进行自注意力机制以获取局部特征信息，而后者是在特征块之间使用自注意力机制以获取全局特征信息。交叉变换模块的网络主体结构如图 2 所示，旨在结合图像特征块内部和特征块之间的注意力，通过堆叠基本块构建一个交叉变换网络。

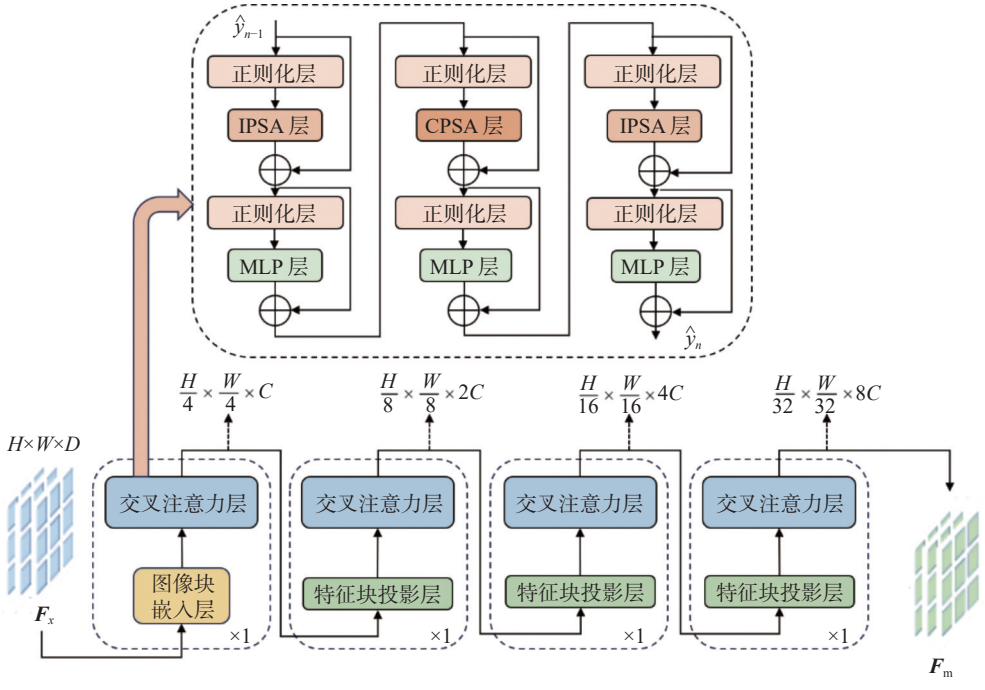

图 2 交叉变换模块

Fig.2 Cross transformer module

首先，本文将特征相关量通过卷积网络映射为二维特征$\boldsymbol{F}_x \in \mathbb{R}^{H\times W\times D}$，以作为交叉变换模块的输入。其次，采用线性投影处理，并借鉴 ViT[32] 中的图像块处理模式，利用特征块嵌入层将通道数扩展至C_1。随后，引入交叉注意力层替代 ViT 中的注意力层，以在不同尺度上进行特征提取。

经过上述预处理，输入图像特征进入第 1 阶段，先经过特征块嵌入层处理形成N个特征块，其数量为$(H_1/N)\times(W_1/N)$，每个特征块的形状为$N\times N\times C_1$。第 1 阶段输出的特征图的形状为$H_1\times W_1\times C_1$，记为$\boldsymbol{F}_{\theta 1}$。此后，进入第 2 阶段，由特征块投影层执行空间深度操作，将形状为$2\times 2\times C$的像素块映射到形状为$1\times 1\times 4C$的特征块，并通过线性投影层将其投影到形状为$1\times 1\times 2C$。随后通过交叉注意力的多次迭代，生成形状为$(H_1/2)\times(W_1/2)\times C_2$的特征映射$\boldsymbol{F}_{\theta 2}$，其特征映射大小缩减一倍，维度增加一倍，类似于 ResNet[33]、Swin[34] 中的操作。经过 4 个阶段的处理后，可以获得 4 个不同尺度和维度的特征图$\{\boldsymbol{F}_{\theta 1},\boldsymbol{F}_{\theta 2},\boldsymbol{F}_{\theta 3},\boldsymbol{F}_{\theta 4}\}$，并将其拼接为运动特征$\boldsymbol{F}_{\mathrm{m}}$，为后续网络模块提供不同尺度的特征图。

如图 2 上层部分所示，CAB 层由两个特征块内自注意力层 IPSA（inner-patch self-attention）和 1 个特征块间自注意力层 CPSA（cross-patch self-attention）组成，其主要运算流程如下：

$$\widehat{\boldsymbol{y}}_{\mathrm{temp1}} = \mathrm{IPSA}\left(\mathrm{LN}\left(\widehat{\boldsymbol{y}}_{n-1}\right)\right)+\widehat{\boldsymbol{y}}_{n-1} \tag{2}$$

$$\widehat{\boldsymbol{y}}_{\mathrm{temp2}} = \mathrm{MLP}\left(\mathrm{LN}\left(\widehat{\boldsymbol{y}}_{\mathrm{temp1}}\right)\right)+\widehat{\boldsymbol{y}}_{\mathrm{temp1}} \tag{3}$$

$$\widehat{\boldsymbol{y}}_{\mathrm{temp3}} = \mathrm{CPSA}\left(\mathrm{LN}\left(\widehat{\boldsymbol{y}}_{\mathrm{temp2}}\right)\right)+\widehat{\boldsymbol{y}}_{\mathrm{temp2}} \tag{4}$$

$$\widehat{\boldsymbol{y}}_{\mathrm{temp4}} = \mathrm{MLP}\left(\mathrm{LN}\left(\widehat{\boldsymbol{y}}_{\mathrm{temp3}}\right)\right)+\widehat{\boldsymbol{y}}_{\mathrm{temp3}} \tag{5}$$

$$\widehat{\boldsymbol{y}}_{\mathrm{temp5}} = \mathrm{IPSA}\left(\mathrm{LN}\left(\widehat{\boldsymbol{y}}_{\mathrm{temp4}}\right)\right)+\widehat{\boldsymbol{y}}_{\mathrm{temp4}} \tag{6}$$

$$\hat{y}_n = \mathrm{MLP}\left(\mathrm{LN}\left(\hat{y}_{\mathrm{temp5}}\right)\right) + \hat{y}_{\mathrm{temp5}} \quad (7)$$

式中：$\hat{y}_{\mathrm{temp}i}$为带有正则化层 LN 的一个模块（例如 IPSA 层、MLP 层）的输出；$\hat{y}_n$为输入序列中第 n 个位置经过注意力机制计算后得到的输出。这种机制模型可以有选择地关注不同模态中的重要信息，从而有效地捕捉关键信息，进一步提升了相关性成本量的精确性。通过这种方式，本文能够在光流估计过程中更准确地捕捉图像间的运动信息，提高了光流估计的精度和鲁棒性。

2.4.2　特征块内自注意力

受卷积神经网络的局部特征提取特性的启发，本文将卷积方法的局部性引入 Transformer，以在每个图像特征块中进行像素级自注意力，称为特征块内自注意力。如图 3 所示，可以将图像特征分成若干个块，把每个特征块看作是一种注意力范围，而不是整个图像特征。同时，Transformer 可根据输入生成不同的注意力图，类似于卷积方法中的动态参数，这与具有固定参数的 CNN 相比具有显著优势。

具体操作如下：首先，初始化输入的维度、分辨率和其他超参数。在前向传播中，输入特征经过规范化层进行处理，以减少内部协变量偏移。将特征重塑为补丁大小的形状，并将其分割成大小相等的补丁，以便后续的注意力计算。其次，在 IPSA 层中，每个补丁内的特征独立地与自身以及其他补丁进行注意力计算，从而聚焦于局部特征之间的关系。通过自注意力机制，对补丁内的特征进行注意力计算，并获得补丁之间的交互信息。然后，在获得了补丁之间的注意力权重后，进行逆操作以重构原始特征。通过逆操作，将注意力权重应用到补丁特征上，并根据其位置重新组合特征。这一步骤旨在维持特征的空间结构，并将局部特征的重要性信息融合到全局特征中。最后，通过残差连接和 MLP 层进一步处理特征。

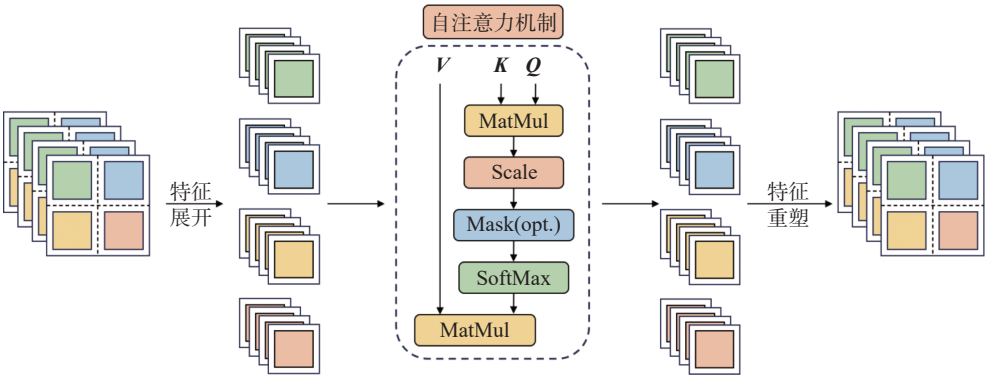

图 3　特征块内自注意力

Fig.3　Inner-patch self-attention

2.4.3　特征块间自注意力

为了更好地利用每个单通道特征图中的全局信息，本文提出了特征块间自注意力。如图 4 所示,该方法将每个通道的特征图分离，并将其划分为$(H/N)\times(W/N)$个图像特征块，然后使用自注意力机制获取整个特征图的全局信息。

具体操作与 IPSA 层不同的是，CPSA 层将注意力机制应用于完整的补丁特征表示，以捕捉全局和局部之间的关系。在 CPSA 层中，通过自注意力机制对完整的补丁特征进行注意力计算，从而获得补丁之间的全局交互信息。

2.4.4　特征聚合模块

为了获取更好的上下文特征信息，受文献 [28] 的启发，本文设计了一种特征聚合模块。如图 5 所示，为了捕捉第 1 帧图像的自相似性，在特征聚合模块中，首先，将上下文特征映射成为一个查询特征映射$\boldsymbol{Q}$和一个键特征映射$\boldsymbol{K}$。接着，对这两个特征映射执行点积操作，并应用 Softmax 函数得到一个注意力矩阵，以编码特征的自相似性。此外，还将查询特征映射$\boldsymbol{Q}$与一组位置嵌入向量进行点积运算，从而增强了注意力矩阵的位置信息。同时，通过学习到的值特征映射$\boldsymbol{V}$对从相关体中编码的运动特征映射进行投影。最终，利用得到的注意力矩阵的加权和，生成了聚合特征$\boldsymbol{F}_{\mathrm{a}}$。

这种方法的好处在于，通过引入自注意力机制和位置嵌入编码，能够有效地捕获图像中的自相似性和位置信息，从而提高了光流估计的精确度和鲁棒性。同时，通过在全局范围内聚合物体

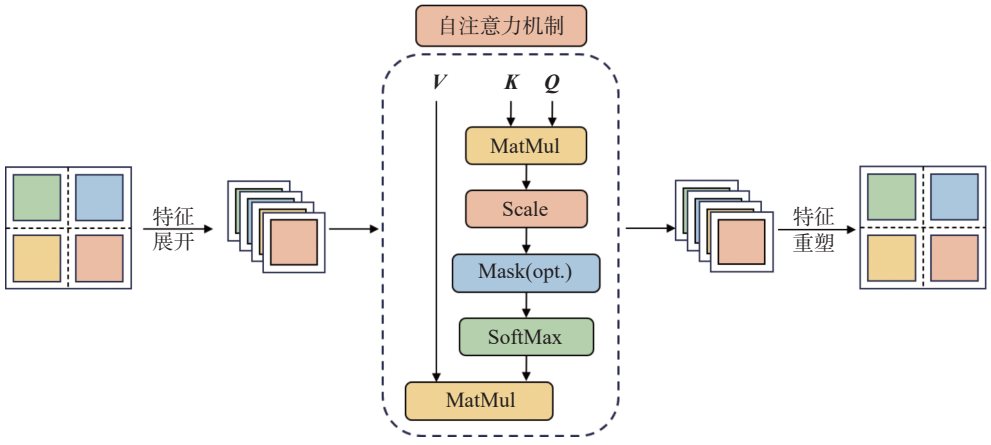

图 4 特征块间自注意力

Fig.4 Cross-patch self-attention

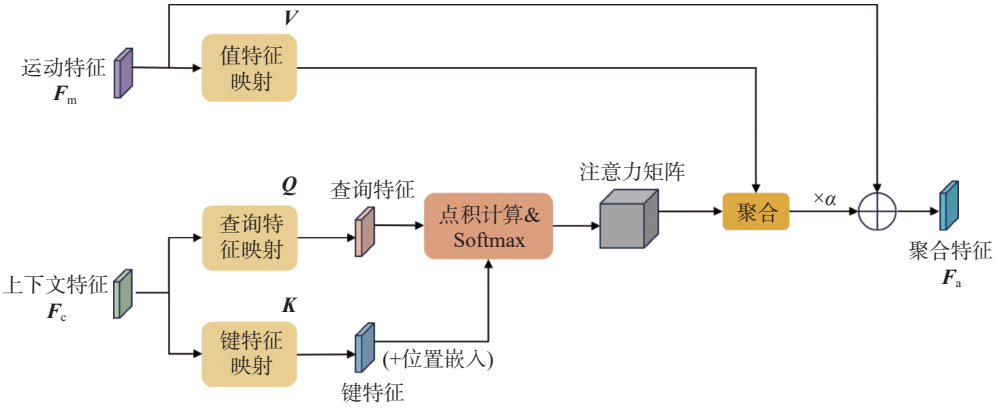

图 5 特征聚合模块

Fig.5 Feature aggregation module

的运动特征，有助于减少局部运动估计的误差，进一步提升了光流估计的性能。

2.5 光流更新与上采样

本文采用了 ConvGRU 模块，并遵循了与 RAFT 一文中相似的设计进行流程细化，通过迭代计算增量光流$\boldsymbol{f}(\boldsymbol{x}+1) \leftarrow \boldsymbol{f}(\boldsymbol{x})+\Delta \boldsymbol{f}(\boldsymbol{x}+1)$，初始光流为$\boldsymbol{f}(0)=0$。与之不同的是，本文所用的递归模块的输入是来自拼接后的全局特征信息，以实现更准确的流量估计。具体而言，在每次迭代步骤中，ConvGRU 模块将检索到的运动特征$\boldsymbol{F}_{\mathrm{m}}$，聚合特征$\boldsymbol{F}_{\mathrm{a}}$、上下文特征$\boldsymbol{F}_{\mathrm{c}}$和当前估计流$\boldsymbol{f}(\boldsymbol{x})$的级联作为输入$\boldsymbol{x}_{\mathrm{t}}$，输出预测流残差。

$$\boldsymbol{x}_{\mathrm{t}}=\operatorname{Concat}\left(\boldsymbol{F}_{\mathrm{m}}, \boldsymbol{F}_{\mathrm{a}}, \boldsymbol{F}_{\mathrm{c}}, \boldsymbol{f}(\boldsymbol{x})\right) \tag{8}$$

$$\Delta \boldsymbol{f}(\boldsymbol{x})=\operatorname{ConvGRU}\left(\boldsymbol{x}_{\mathrm{t}}\right) \tag{9}$$

随后，本文采用了凸优化上采样的方法对低分辨率光流进行上采样，相比于传统的双线性插值方法，该方法在图像边界处能够获得更加平滑的效果，有效地改善了光流估计中弱纹理和复杂结构等运动边缘的模糊问题，从而提高了光流上采样的质量。

2.6 损失函数

本文使用绝对损失函数 L 来衡量预测值与实际值之间的绝对差异。对于给定的光流真实值$\boldsymbol{f}(\boldsymbol{gt})$和每次迭代之后的光流预测值$\boldsymbol{f}(\boldsymbol{x})$：

$$L=\sum_{x=1}^{k} \gamma^{k-x}\|\boldsymbol{f}(\boldsymbol{x})-\boldsymbol{f}(\boldsymbol{gt})\|_1 \tag{10}$$

式中：k 为迭代次数，设置为 6；γ为权重，设置为 0.8。

3 实验与结果分析

3.1 数据集

为了验证本文所提出的模型的实用性和有效性，在 FlyingChairs 和 FlyingThings3D 这两个数据集进行预训练，在 MPI-Sintel 和 KITTI 2015 这两个数据集上进行评估。FlyingChairs 与 FlyingThings3D 均为规模巨大的人工合成数据集，前者采用图像

的随机仿射变换生成第 2 帧，而后者拥有三维运动的物体和光照视觉渲染效果，涵盖大量的大位移与大遮挡等场景，分别包含 22872 和 19635 个训练图像对，这两个数据集被用于模型的预训练。MPI-Sintel 是由开源动画电影制作并合成的光流数据集，该数据集由 Clean 和 Final 这两个子数据集组成，每个子数据集共包含 1041 个测试图像对和 552 个训练图像对，其中 Clean 数据集仅含简单的光照渲染和镜面反射，而 Final 数据集则涵盖了运动模糊、大气效应等复杂场景，更加具挑战性。KITTI 2015 数据集目前广泛用于自动驾驶场景下的计算机视觉算法评测，它具有稀疏光流真值的现实道路场景，包括 200 对训练图像和 200 对测试图像。

3.2　评价指标

本文采用了在光流领域常用的评价指标。具体而言，在 MPI-Sintel 数据集中，使用端点误差（end-point-error，EPE）来度量光流预测值$\boldsymbol{f}(\boldsymbol{x})$与光流真实值$\boldsymbol{f}(\boldsymbol{gt})$之间的距离平均值 E：

$$E=\sqrt{(\boldsymbol{f}(\boldsymbol{x})-\boldsymbol{f}(\boldsymbol{gt}))^2} \tag{11}$$

而在 KITTI 2015 数据集中，除端点误差这一指标外，还使用 $F1$ 来表示图像整体区域中光流异常的比率。当一个图像点的光流真实值$\boldsymbol{f}(\boldsymbol{gt})$与光流预测值$\boldsymbol{f}(\boldsymbol{x})$同时满足以下两个条件时，认为该图像点的光流异常值超出了所设阈值。

$$\begin{cases} \|\boldsymbol{f}(\boldsymbol{gt})-\boldsymbol{f}(\boldsymbol{x})\|_2>3 \\ \|\boldsymbol{f}(\boldsymbol{gt})-\boldsymbol{f}(\boldsymbol{x})\|_2/\|\boldsymbol{f}(\boldsymbol{gt})\|_2>0.05 \end{cases} \tag{12}$$

3.3　实验设置

本文所构建的模型使用 PyTorch 框架，基于两张 2080TiGPU，采用 AdamW 优化器和随机初始化权重策略，在不同数据集上采用单周期的线性学习率，并进行混合精度训练，以提高训练效率。同时，针对不同数据集设置不同的最大学习率，各个数据集上的训练参数设置。

本文训练步骤：首先在 FlyingChairs 和 FlyingThings3D 这两个数据集进行预训练，然后直接在 MPI-Sintel 和 KITTI 2015 训练集上进行微调并测试模型性能。具体在各个数据集上的训练参数设置如表 1 所示。

表 1　实验训练参数设置

Tab.1　Training parameter settings of experiment

训练集	权重源	训练数据集	学习率	批量大小	迭代次数/次	裁剪尺寸
FlyingChairs		FlyingChairs	$2e^{-4}$	8	10^5	[368, 496]
FlyingThings3D	FlyingChairs	FlyingThings3D	$2e^{-4}$	8	12×10^4	[400, 720]
MPI-Sintel	FlyingThings3D	MPI-Sintel+FlyingThings3D	$1.25e^{-4}$	6	10^5	[368, 768]
KITTI 2015	MPI-Sintel	KITTI 2015	$1.25e^{-4}$	6	5×10^4	[288, 960]

3.4　消融实验

消融实验旨在对模型的关键模块进行分析，展示各个子模块对总体性能的影响。首先在 FlyingChairs 和 FlyingThings3D 这两个训练集上训练，然后在 MPI-Sintel 和 KITTI 2015 这两个数据集对模型进行了评估。在一系列消融实验过程中，保证其他设置与最终模型一致。如表 2 所示，展示了每个参数对模型性能的影响，用下划线来标识本文方法所采用的设置。

如表 2 所示，基准是指在 RAFT(small) 模型上进行训练。特征聚合模块：本文训练了一个去除特征聚合网络的模型。结果表明，在使用特征聚合模块的情况下，本文方法取得了更好的结果。这证实了特征聚合模块对整个模型训练的益处。这也表明，该模块通过全局范围内聚合物体的运动特征，有助于降低局部运动估计的误差。更新模块：ConvGRU 是在普通的门控单元 GRU 的基础上进行了扩展，它将普通的循环结构替换为卷积结构。结果表明，卷积门控单元更新在更新过程中呈现出更显著的效果，这表明其具有更有效地捕获和控制信息流的能力。上采样模块：本文对双线性上采样与本文提出的凸上采样模块进行了比较。结果显示，本文提出的上采样模块在产生更优结果方面表现更为突出，尤其是在运动边界附近的情况下。注意力模块：本文进行了交叉注意力机制和传统自注意力机制的对比实验，并深入观察了它们之间的差异。实验结果表明，所采用的交叉注意力机制在精度上表现更为突出，凸显了其在特征处理上相对于传统自注意力机制更为优越。

表 2 消融实验结果

Tab.2 Ablation experiment results

实验	变量	MPI-Sintel		KITTI 2015	
		EPE (Clean)	EPE (Final)	EPE	F1-all/(%)
基准		2.19	3.28	8.46	26.68
特征聚合模块	无	2.32	3.56	7.98	26.97
	有	2.12	3.34	7.16	26.63
更新模块	GRU	2.14	3.31	7.76	26.23
	ConvGRU	2.10	3.15	7.21	26.25
上采样模块	双线性插值	2.16	3.28	7.82	26.72
	凸优化	2.10	3.13	7.20	26.22
注意力模块	自注意力	2.24	3.62	8.25	26.72
	交叉注意力	2.00	3.43	7.23	24.87

3.5 实验结果与其他方法的比较

3.5.1 MPI-Sintel 数据集

如表 3 所示，以 RAFT(small)作为基准模型，在 FlyingChairs 和 FlyingThings3D 这两个数据集上预训练后，将所得到的权重用于在 MPI Sintel 数据集上进行评估验证。train 表示在训练集上取得的结果，test 表示该模型在测试集上微调后的结果。

本文研究方法在 Clean 和 Final 测试集上的端点误差分别为 4.15 和 4.42，精度均达到最优，说明该研究方法在 MPI Sintel 测试集上表现优异。此外，Final 图像序列相较于 Clean 图像序列包含着更复杂的噪声干扰条件，说明本文方法相较于其他对比方法表现出了更好的鲁棒性。

表 3 MPI-Sintel 数据集上的定量评估

Tab.3 Quantitative evaluation on MPI-Sintel datasets

方法	MPI Sintel (Clean)		MPI Sintel (Final)	
	EPE (train)	EPE (test)	EPE (train)	EPE (test)
FlowNet[12]	3.57	6.08	5.25	7.88
FlowNet2[14]	2.02	4.16	3.14	5.74
SpyNet[15]	4.12	6.64	5.57	8.36
PWC-Net[16]	2.55	3.86	3.93	5.13
LiteFlowNet[17]	2.48	4.54	4.04	5.38
FastFlowNet[18]	2.89	4.89	4.14	6.08
RAFT(small)[20]	2.13	3.51	3.28	4.50
本文方法	**2.00**	**3.43**	**3.12**	**4.42**

如图 6 所示，呈现了本文网络模型与基准模型 RAFT(small)网络模型在 MPI-Sintel(Final)测试集上的可视化对比，红色线圈表示说明本文方法在物体运动遮挡方面比 RAFT(small)网络模型处理得更好。

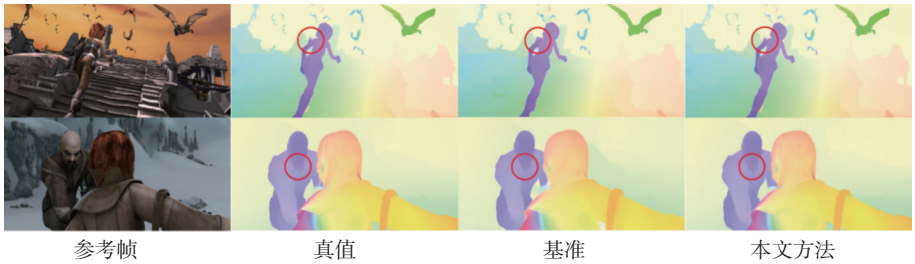

图 6 MPI-Sintel(Final)测试集上的可视化比较

Fig.6 Visual comparison on MPI-Sintel (Final) test set

3.5.2 KITTI 2015 数据集

本文方法用在 FlyingChairs 和 FlyingThings3D 数据集上进行预训练，在 KITTI 2015 的训练集上进行微调, 并在其测试集上与现有代表方法进行对比实验，以验证本文模型在真实道路场景数据集上的泛化能力。

如表 4 所示，本文模型在 KITTI 2015 数据集上取得了不错的性能表现。其端点误差达到了 7.23，光流异常值为 24.87%，这两项评价指标均优于其他对比方法，说明本文模型在经过真实场景数据集训练后，其预测光流的能力能够达到一定的精确性。如图 7 所示，呈现了本文网络模型在 KITTI 2015 测试集上的光流预测可视化。

表 4 KITTI 2015 数据集上的定量评估

Tab.4 Quantitative evaluation on KITTI 2015 datasets

方法	KITTI 2015	
	EPE	F1-all/(%)
FlowNet[12]	—	—
FlowNet2[14]	10.06	28.20
SpyNet[15]	—	—
PWC-Net[16]	10.35	33.67
LiteFlowNet[17]	10.39	28.50
FastFlowNet[18]	12.24	33.10
RAFT(small)[20]	7.49	25.27
本文方法	**7.23**	**24.87**

图 7　KITTI 2015 测试集上的可视化

Fig.7　Visualization on KITTI 2015 test set

4　结　论

提出了一种结合卷积和交叉变换网络的光流估计模型。首先，该模型利用卷积神经网络对连续两帧图像进行特征提取，随后引入了交叉注意力机制进行特征增强，提高对图像之间复杂关系的理解能力。其次，结合上下文特征网络编码提供的上下文信息，通过特征聚合模块产生聚合特征，进一步提升对图像全局特征的感知和理解。最后，使用 ConvGRU 模块进行光流的迭代更新，并通过上采样模块产生光流特征估计图，完成光流估计任务。通过在 MPI-Sintel 和 KITTI 2015 数据集上进行大量的对比实验和消融实验，本文方法可以有效地提高光流估计的精度，并具有良好的泛化性。在以后的研究中，将致力于通过结合模型优化和推理加速技术，实现使用更少的迭代次数去实现光流估计的目标，从而为实际应用场景提供更加可靠和高效的解决方案。

参考文献：

[1] CHOI H, KANG B, KIM D E. Moving object tracking based on sparse optical flow with moving window and target estimator[J]. Sensors, 2022, 22(8): 2878.

[2] SUN S Y, KUANG Z H, SHENG L, et al. Optical flow guided feature: a fast and robust motion representation for video action recognition[C]//Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 1390–1399.

[3] MIN Q, HUANG Y P. Motion detection using binocular image flow in dynamic scenes[J]. EURASIP Journal on Advances in Signal Processing, 2016, 2016(1): 49.

[4] DE CROON G C H E, DE WAGTER C, SEIDL T. Enhancing optical-flow-based control by learning visual appearance cues for flying robots[J]. Nature Machine Intelligence, 2021, 3(1): 33−41.

[5] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. Long Beach: Curran Associates Inc. , 2017: 6000–6010.

[6] BUTLER D J, WULFF J, STANLEY G B, et al. A naturalistic open source movie for optical flow evaluation[C]//12th European Conference on Computer Vision on Computer Vision. Florence: Springer, 2012: 611–625.

[7] GEIGER A, LENZ P, URTASUN R. Are we ready for autonomous driving? The KITTI vision benchmark suite[C]//2012 IEEE Conference on Computer Vision and Pattern Recognition. Providence: IEEE, 2012: 3354–3361.

[8] LUCAS B D, KANADE T. An iterative image registration technique with an application to stereo vision[C]//Proceedings of the 7th International Joint Conference on Artificial Intelligence. Vancouver: Morgan Kaufmann Publishers Inc. , 1981: 674–679.

[9] HORN B K P, SCHUNCK B G. Determining optical flow[J]. Artificial Intelligence, 1981, 17(1/3): 185−203.

[10] FARNEBÄCK G. Two-frame motion estimation based on polynomial expansion[C]//13th Scandinavian Conference on Image Analysis. Halmstad: Springer, 2003: 363–370.

[11] BAKER S, SCHARSTEIN D, LEWIS J P, et al. A database and evaluation methodology for optical flow[J]. International Journal of Computer Vision, 2011, 92(1): 1−31.

[12] 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. Santiago: IEEE, 2015: 2758–2766.

[13] MAYER N, ILG E, HÄUSSER 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. Las Vegas: IEEE, 2016: 4040–4048.

[14] 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. Honolulu: IEEE, 2017: 1647–1655.

[15] 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. Honolulu: IEEE, 2017: 2720–2729.

[16] SUN D P, YANG X D, LIU M Y, et al. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume[C]//Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 8934–8943.

[17] HUI T W, TANG X O, LOY C C. LiteFlowNet: a lightweight convolutional neural network for optical flow estimation[C]//Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 8981–8989.

[18] KONG L T, SHEN C H, YANG J. FastFlowNet: a lightweight network for fast optical flow estimation[C]//2021 IEEE International Conference on Robotics and Automation (ICRA). Xi'an: IEEE, 2021: 10310–10316.

[19] HUR J, ROTH S. Iterative residual refinement for joint optical flow and occlusion estimation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Long Beach: IEEE, 2019: 5747–5756.

[20] TEED Z, DENG J. RAFT: recurrent all-pairs field transforms for optical flow[C]//16th European Conference on Computer Vision. Glasgow: Springer, 2020: 402–419.

[21] CHO K, VAN MERRIËNBOER B, BAHDANAU D, et al. On the properties of neural machine translation: encoder-decoder approaches[C]//Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation. Doha: Association for Computational Linguistics, 2014: 103–111.

[22] JADERBERG M, SIMONYAN K, ZISSERMAN A, et al. Spatial transformer networks[C]//Proceedings of the 29th International Conference on Neural Information Processing Systems. Montreal: MIT Press, 2015: 2017–2025.

[23] HU J, SHEN L, SUN G. Squeeze-and-excitation networks[C]//Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018: 7132–7141.

[24] CARION N, MASSA F, SYNNAEVE G, et al. End-to-end object detection with transformers[C]//16th European Conference on Computer Vision. Glasgow: Springer, 2020: 213–229.

[25] GUO M H, CAI J X, LIU Z N, et al. PCT: point cloud transformer[J]. Computational Visual Media, 2021, 7(2): 187−199.

[26] CHEN H T, WANG Y H, GUO T Y, et al. Pre-trained image processing transformer[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. Nashville: IEEE, 2021: 12294–12305.

[27] ZENG Y H, FU J L, CHAO H Y. Learning joint spatial-temporal transformations for video inpainting[C]//16th European Conference on Computer Vision. Glasgow: Springer, 2020: 528–543.

[28] JIANG S H, 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. Montreal: IEEE, 2021: 9752–9761.

[29] HUANG Z Y, SHI X Y, ZHANG C, et al. FlowFormer: a transformer architecture for optical flow[C]//17th European Conference on Computer Vision. Tel Aviv: Springer, 2022: 668–685.

[30] CHU X X, TIAN Z, WANG Y Q, et al. Twins: revisiting the design of spatial attention in vision transformers[C]//Proceedings of the 35th International Conference on Neural Information Processing Systems. New York: Curran Associates Inc. , 2021: 716.

[31] LIN H Z, CHENG X, WU X Y, et al. CAT: cross attention in vision transformer[C]//2022 IEEE International Conference on Multimedia and Expo (ICME). Taipei, China: IEEE, 2022: 1–6.

[32] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16x16 words: transformers for image recognition at scale[C]//Proceedings of the 9th International Conference on Learning Representations. Vienna: OpenReview. net, 2020.

[33] HE K M, ZHANG X Y, REN S Q, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016: 770–778.

[34] LIU Z, LIN Y T, CAO Y, et al. Swin transformer: hierarchical vision transformer using shifted windows[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. Montreal: IEEE, 2021: 9992–10002.

（编辑：董 伟）