ISSN 1003-501X CN 51-1346/O4

光电工程

Opto-Electronic Engineering

中文核心期刊 中国科技核心期刊
Scopus CSCD

融合Swin Transformer的立体匹配方法STransMNet

王高平，李珣，贾雪芳，李哲文，王文杰

引用本文:

王高平，李珣，贾雪芳，等. 融合Swin Transformer的立体匹配方法STransMNet[J]. 光电工程，2023， **50**(4): 220246.

Wang G P, Li X, Jia X F, et al. STransMNet: a stereo matching method with swin transformer fusion[J]. *Opto-Electron Eng*, 2023, **50**(4): 220246.

https://doi.org/10.12086/oee.2023.220246

收稿日期: 2022-10-08; 修改日期: 2023-01-11; 录用日期: 2023-01-19

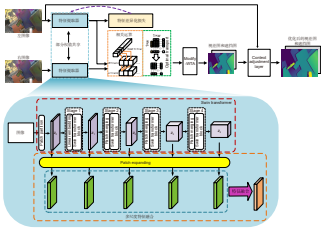
相关论文

Opto-Electronic Engineering
光 电 工 程

Article
2023 年, 第 50 卷, 第 4 期

DOI: 10.12086/oee.2023.220246

融合 Swin Transformer 的立体匹配方法 STransMNet



王高平[1]，李　珣[1,2*]，贾雪芳[1]，李哲文[1]，王文杰[1]

[1]西安工程大学电子信息学院，陕西 西安 710600；
[2]陕西省人工智能联合实验室西安工程大学分部，陕西 西安 710600

摘要：针对基于 CNN 的立体匹配方法中特征提取难以较好学习全局和远程上下文信息的问题，提出一种基于 Swin Transformer 的立体匹配网络改进模型 (stereo matching net with swin transformer fusion, STransMNet)。分析了在立体匹配过程中，聚合局部和全局上下文信息的必要性和匹配特征的差异性。改进了特征提取模块，把基于 CNN 的方法替换为基于 Transformer 的 Swin Transformer 方法；并在 Swin Transformer 中加入多尺度特征融合模块，使得输出特征同时包含浅层和深层语义信息；通过提出特征差异化损失改进了损失函数，以增强模型对细节的注意力。最后，在多个公开数据集上与 STTR-light 模型进行了对比实验，误差 (End-Point-Error, EPE) 和匹配错误率 3 px error 均有明显降低。

关键词：立体匹配；Swin Transformer；深度学习；STransMNet

中图分类号：TP391.4　　**文献标志码：**A

王高平，李珣，贾雪芳，等. 融合 Swin Transformer 的立体匹配方法 STransMNet[J]. 光电工程，2023，**50**(4): 220246
Wang G P, Li X, Jia X F, et al. STransMNet: a stereo matching method with swin transformer fusion[J]. *Opto-Electron Eng*, 2023, **50**(4): 220246

STransMNet: a stereo matching method with swin transformer fusion

Wang Gaoping[1], Li Xun[1,2*], Jia Xuefang[1], Li Zhewen[1], Wang Wenjie[1]

[1]School of Electronics and Information, Xi'an Polytechnic University, Xi'an, Shaanxi 710600, China;
[2]Xi'an Polytechnic University Branch of Shaanxi Artificial Intelligence Joint Laboratory, Xi'an, Shaanxi 710600, China

Abstract: Feature extraction in the CNN-based stereo matching models has the problem that it is difficult to learn global and long-range context information. To solve this problem, an improved model STransMNet stereo matching network based on the Swin Transformer is proposed in this paper. We analyze the necessity of the aggregated local and global context information. Then the difference in matching features during the stereo matching process is discussed. The feature extraction module is improved by replacing the CNN-based algorithm with the Transformer-based Swin Transformer algorithm to enhance the model's ability to capture remote context information. The multi-scale fusion module is added in Swin Transformer to make the output features contain shallow and deep semantic information. The loss function is improved by introducing the feature differentiation loss to enhance the model's attention to details. Finally, the comparative experiments with the STTR-light model are conducted on multiple

收稿日期：2022-10-08；修回日期：2023-01-11；录用日期：2023-01-19
基金项目：国家自然科学基金资助项目 (61971339)；陕西省自然科学基础研究计划项目 (2022JM407)
*通信作者：李珣，lixun@xpu.edu.cn。

public datasets, showing that the End-Point-Error (EPE) and the matching error rate of 3 px error are significantly reduced.

Keywords: stereo matching; Swin Transformer; deep learning; STransMNet

1 引 言

在物体分类[1]、目标检测[2]和语义与实例分割[3-4]等机器视觉任务中，图像的深度信息被广泛应用。双目深度估计利用左右相机的几何关系和视差图来计算深度图。其中双目相机的几何关系通过相机标定和出厂参数获得，而视差图则要通过立体匹配获得。立体匹配是一个寻找 3D 点在左右图像上投影的像素的过程[5-6]。传统的立体匹配算法把匹配过程分为四个步骤：匹配代价计算、代价聚合、视差计算和视差优化[7]。由于传统方法估计的视差图噪点多、匹配准确率低等问题很难对其结果进行应用。随着深度学习方法的发展，特别是其在图像处理领域的快速应用，基于卷积神经网络 (convolutional neural network, CNN) 端到端训练的模型被用来估计视差图。Nikolaus[8] 等人最早提出用于立体匹配的神经网络模型 DispNet，他们还贡献了一个开源的大型双目图像数据集 Sceneflow。在立体匹配网络的特征提取阶段，模型[8-12]采用权值共享的孪生神经网络[13]。DispNet 和 GCNet[9] 在提取图像特征时，利用大卷积核来缩小特征尺寸，DispNet 中使用 7×7 和 5×5 的大卷积核，GCNet 使用 5×5 的大卷积核。而 Chen[10] 等人提出的金字塔立体匹配网络 (pyramid stereo matching net, PSMNet) 中利用三个级联的 3×3 小卷积核达到一个 7×7 卷积核同样的感受野。PSMNet 在 4 个卷积块后加入空间金字塔池化模块 (spatial pyramid pooling, SPP)[14-15]，以融合不同尺度和不同位置的上下文信息。在形成立体匹配代价体时，通常有两种方式。第一种是特征堆叠[9]，左右图像特征 ($H\times W\times C$，H、W 和 C 分别表示特征的高、宽和通道) 堆叠之后，形成一个四维匹配代价体 ($H\times W\times 2C\times D$，$D$ 表示预先指定的视差范围)。针对四维代价体，代价聚合阶段通常采用计算量较大且内存消耗较高的 3D 卷积[9-10,16-20]。第二种方式是相关运算[8]，提取的左右图像特征相关运算后，形成一个三维代价体 ($H\times W\times D$)。针对三维代价体，利用普通的二维卷积进行代价聚合[8,12,19]。2020 年，Xu 和 Zhang[11] 在 AANet (adaptive aggregation network) 中提出尺度内代价聚合和跨尺度代价聚合模块，使用 2D 卷积完成代价聚合，降低了模型的计算量。在得到匹配代价体之后，传统算法采用赢家通吃 (winner takes all, WTA)[5] 的方法计算视差，但是 WTA 不可微，难以应用到基于学习的方法中。DispNet[8] 直接利用卷积来预测视差，GCNet[9]、StereoNet[12] 和 AANet[11] 中则采用可微的 Soft Argmin，STTR (stereo transformer)[19] 利用改进版的 WTA (Modify-WTA)[20] 计算视差。计算的初始视差还需要进行视差优化，以进一步提升匹配精度。StereoNet[12] 中提出分层细化：边缘感知上采样模块。STTR[19] 中提出上下文信息调整层 (context adjustment layer) 模块。通常立体匹配模型会预先指定视差范围，但这并不合理。距离较近的物体具有大范围视差，限制视差范围也就限制了对较近物体深度的探测能力。2021 年，Li[19] 等人提出的 STTR 突破了视差受限问题，使得视差范围扩展至整个图像宽度。但 STTR 仍然采用基于 CNN 的方法提取特征，这限制了模型捕获远程语义信息的能力。STTR-light[19] 是 STTR 的轻量级版本，与 STTR 的网络结构相同。但是 STTR 和 STTR-light 的亚分辨率图像尺寸不同。与 STTR 相比，STTR-light 虽然性能有微弱下降，但是节省了内存与计算量。

与基于 CNN 的方法相比，基于 Transformer 的方法捕获全局特征的性能更好。Transformer 最开始出现在自然语言处理领域，因为其优异的性能，很多学者开始把 Transformer 的注意力思想应用到图像领域中[21-22]。2017 年，Hu[21] 等人提出 Squeeze-and-excitation networks，该模型通过计算特征通道维度的注意力，大大提升了网络的学习能力。此后 Woo[22] 等人在通道维度的基础上增加了空间维度注意力计算，进一步提升了网络的学习能力。基于注意力机制在图像领域的出色表现，Dosovitskiy[23] 等人首次设计了用于图像分类的纯 Transformer 网络 VIT (vision Transformer)。VIT 把图像分成多个 patches，然后转化为类似于自然语言输入的序列 tokens，但是 VIT 的 patch 细粒度较大，无法提取不同尺度的目标特征。因此，Han[24] 等人在 VIT 基础上提出 Transformer in

Transformer，在 patch 内部增加了 Transformer 运算。Fang[25] 等人提出用于物体检测的纯 Transformer 结构。虽然 VIT 提升了图像分类网络的精度，但是其内存和算力开销巨大。针对这个问题，Swin Transformer (Transformer using Shifted Windows)[26] 提出了 Shift Windows 方法，与 VIT 相比在节省内存和算力等开销的同时还提高了图像分类精度。

虽然基于 CNN 的深度学习方法具有出色的特征表示能力，但由于卷积运算的固有局部性，它们通常表现出对显式远程关系建模的局限性，在弱纹理、形状和尺寸差异很大的目标上，单纯使用 CNN 的方法估计的结果往往是不理想的。因此，本文提出融合 Swin Transformer 的 STranSMNet 模型，利用 Transformer 方法捕获远程语义信息。此外，本文提出特征差异化损失，改进了 STTR-light 模型的损失函数，加强模型对特征细节的注意力。

2 立体匹配网络 STransMNet

STTR-light 网络结构如图 1 所示，该网络由基于 CNN 的漏斗形特征提取模块[15]、Self and Cross-attention 注意力计算模块[19]、Optimal Transport 匹配代价约束模块[27-29]、Modify-WTA[20] 视差回归模块和 Context Adjustment Layer 视差优化模块[19] 组成。

STTR-light 的特征提取模块采用类似 PSMNet[10] 中的基于 CNN 方法的沙漏形架构。通过不同尺寸的池化变换，依次生成多尺度特征，接着通过上采样使多尺度特征变换为相同大小，最后在通道维度堆叠到一起，该结构旨在融合全局上下文信息[10]，相比之下，Swin Transformer 中的 Transformer 方法更适合捕获全局特征。Transformer 方法中每一个 token 产生 3 个向量 $\boldsymbol{Q}$、$\boldsymbol{K}$ 和 $\boldsymbol{V}$，每个 token 的 $\boldsymbol{Q}$ 与所有 token 的 $\boldsymbol{K}$ 做查询，得到对应的注意力系数，再与 $\boldsymbol{K}$ 相应的 $\boldsymbol{V}$ 相乘，最后把所有相乘的结果相加作为输出。为了获得更好的匹配特征，本文以 Swin Transformer 为基准，加入多尺度特征融合，形成基于 Swin Transformer 的特征提取模块；并在损失函数中加入特征差异化损失；通过修改 STTR-light 的结构以集成上述内容，提出 STransMNet 模型，STranSMNet 网络结构如图 2 所示。

STransMNet 首先利用改进的 Swin Transformer 提取左右图像特征。然后利用结构简单的相关运算[8] 计算匹配代价。相关运算是一种计算特征相似度的方法，图 2(a) 中 $\mathrm{L}: 1\times W\times C$ 和 $\mathrm{R}: 1\times W\times C$ 分别表示左图像一极线特征和右图像一极线特征，这两个特征相关运算之后，得到相应的匹配代价体 $\mathrm{Cost}: 1\times 1\times W$，通道维度 C 被压缩为 1。同时，在训练模型时需要计算左图像特征的特征差异化损失，该损失用于辅助训练特征提取器。然后通过 Optimal Transport[27] 模块对匹配代价进行唯一性约束。Optimal Transport 是一种软分配方案，相比于传统算法中的硬分配，它具有可微性。接着利用 Modify-WTA[20] 回归生成亚分辨率视差图，通过上采样恢复为原始图像分辨率的视差图。然而恢复的视差图缺乏上下文信息，利用 Context Adjustment Layer[19] 模块融合左图像与恢复的视差图、遮挡图信息，生成优化后的视差图和遮挡图。Context Adjustment Layer 模块有两条支路，第一条支路输入堆叠的遮挡图和左图像，经过两个卷积块得到估计的遮挡图，第一个卷积块中卷积核大小为 3×3，激活函数为 ReLU；第二个卷积块中卷积核大小为 5×5，激活函数为 Sigmoid。第二条支路输入堆叠的视差图和左图像，经过八个残差块后得到优化后的视差图。

2.1 基于 Swin Transformer 的特征提取模块

不同于 VIT，Swin Transformer 没有直接计算全

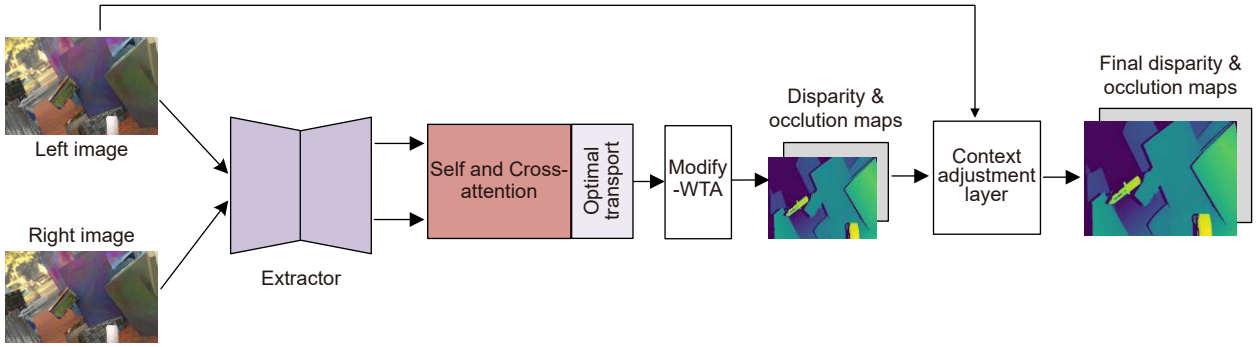


图 1 STTR-light 网络结构

Fig. 1 The network structure of STTR-light

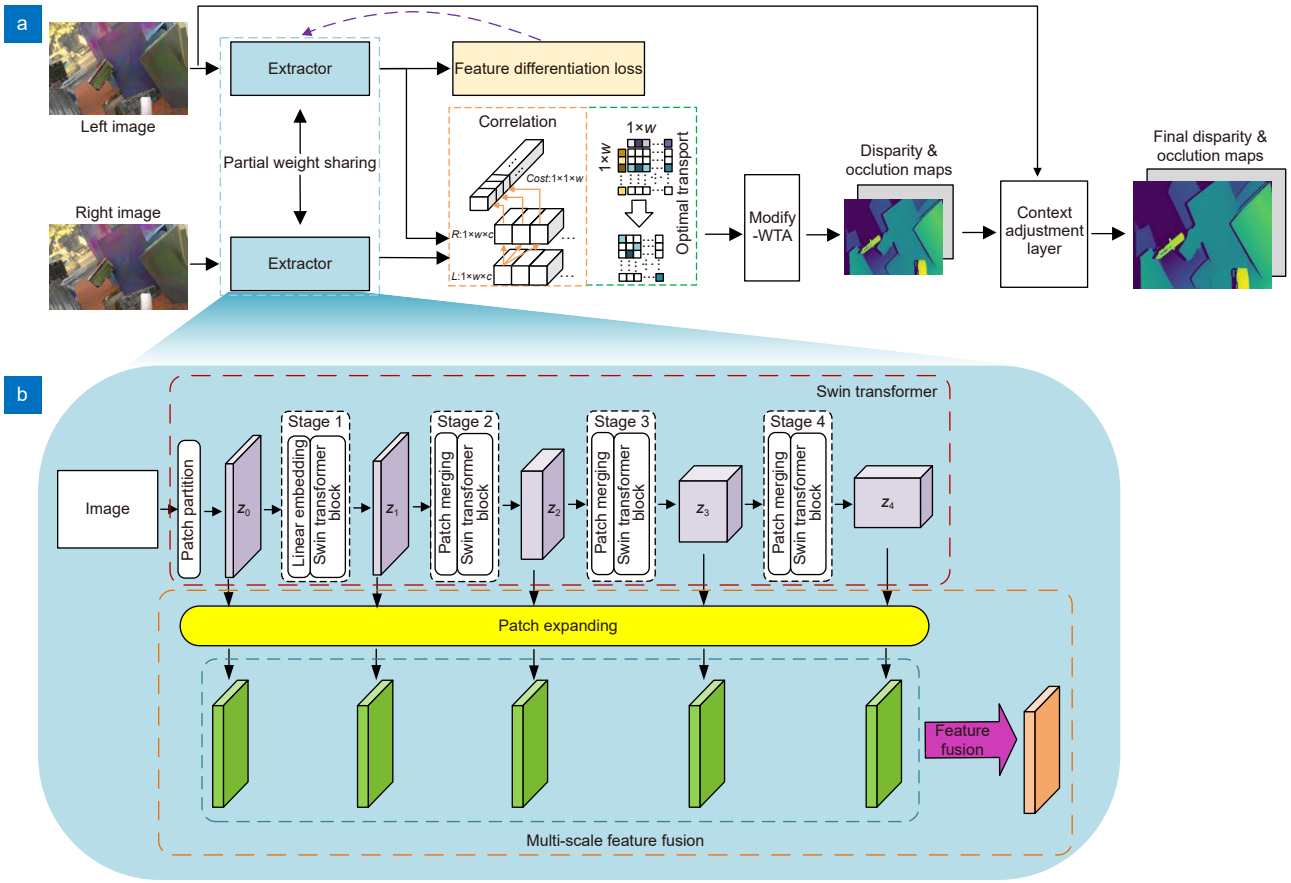


图 2 (a) STransMNet 网络结构; (b) 特征提取器的结构

Fig. 2 (a) The network structure of STransMNet; (b) The structure of extractor

局注意力，而是首先把图像拆分成大小固定的 Window；其次在 Window 内部计算局部注意力；最后通过 Shift Window，融合相邻的四个 Window 信息。如图 2(b) 所示，Swin Transformer 提取特征过程分为 4 个 Stage，在第二、三、四 Stage 中，通过下采样 patch merging 缩减图像特征的宽度和高度，同时增加通道数量。Patch merging 在下采样过程中，不会丢失信息，它首先通过间隔采样的方式得到四个小尺寸特征图，然后对四个小尺寸特征图进行堆叠和线性变换以融合成两个小尺寸特征图。Stage 1 到 Stage 4 中依次有 1、1、3 和 1 个注意力计算模块，该模块由普通的多头自注意力模块串联窗口移动多头自注意力模块组成。注意力计算模块计算过程为

$$\hat{z}^{l} = \text{W-MSA}\left(\text{LN}\left(z^{l-1}\right)\right) + z^{l-1}, \tag{1}$$

$$z^{l} = \text{MLP}\left(\text{LN}\left(\hat{z}^{l}\right)\right) + \hat{z}^{l}, \tag{2}$$

$$]\hat{z}^{l+1} = \text{SW-MSA}\left(\text{LN}\left(z^{l}\right)\right) + z^{l}, \tag{3}$$

$$z^{l+1} = \text{MLP}\left(\text{LN}\left(\hat{z}^{l+1}\right)\right) + \hat{z}^{l+1}, \tag{4}$$

式中，W-MSA、SW-MSA、MLP 和 LN 分别表示窗口内多头自注意力、窗口移动多头自注意力、多层感知机和线性层运算；l 表示网络层数；$\hat{z}$表示 (窗口移动) 多头自注意力输出的特征；z 表示多层感知机输出的特征。把经过极线校正的左右图像输入到 Swin Transformer 模块中，输出多尺度特征图z_0、z_1、z_2、z_3和z_4 (图 2(b) 中紫色方块)。通过 Patch Partition 对原图像 4 倍下采样后得到z_0，z_0大小为$(H/4)\times(W/4)\times 96$；$z_1$大小为$(H/4)\times(W/4)\times 96$；$z_2$大小为$(H/8)\times(W/8)\times 192$；$z_3$大小为$(H/16)\times(W/16)\times 384$；$z_4$大小为$(H/32)\times(W/32)\times 768$。$z_0$中每个特征融合了原图像中相邻的 4 个像素信息，称为亚分辨率图像。在后续的匹配工作中，本文计算的匹配代价和亚分辨率视差图与该亚分辨图像尺寸相同。Swin Transformer 的 4 个 Stage 依次生成z_1、z_2、z_3和z_4，每个特征融合了当前尺度的 4 个相邻 Window 内的像素上下文信息。z_4总共聚合

32 个 Window 内像素的上下文信息，即全局的上下文信息。在 Patch Expanding[30] 模块中，多尺度特征通过线性层的变换，使得它们大小相同，均为 $(H/4)\times(W/4)\times 128$ (图 2(b) 中绿色方块)。最后在通道维度上以相加方式融合多尺度特征 (图 2(b) 中橙色方块)。浅层特征包含局部丰富的细节信息，而深层特征包含全局关联信息。通过加入多尺度特征融合模块，使得改进后的 Swin Transformer 输出的图像特征包含了浅层和深层语义信息。

在提取左右图像特征的过程中，各 Stage 的参数是部分权值共享的。其中式 (1) 和式 (3) 中的参数权值完全共享；而式 (2) 和式 (4) 中参数单独训练。虽然权值共享可以使模型收敛速度加快，但是本文提出的特征差异化损失只能监督左图像或者右图像。如果完全权值共享，则相当于对左右图像同时监督。部分权值共享在一定程度上加快了模型的收敛速度，也使得模型既能捕获左右图像特征的共性，又能捕获它们之间的差别。

2.2 特征差异化损失

为了提高模型关注细节的能力，本文提出特征差异化损失。它通过强制对左图像同一极线上的像素特征进行分类训练，使得每一个特征都比较独特。由于立体匹配只在极线上进行左右图像的像素匹配，故只需对同一极线上的像素特征计算该损失，监督标签为 $y=[0,1,2,\cdots,W-1]$，W为特征宽度，且所有极线共用该标签。虽然模型只计算左图像的特征差异化损失，但是因为左右图像的特征提取模块是部分权值共享的，所以右图像极线上的各个像素特征也具有差异性。特征差异化损失L_{diff}计算为

$$L_{\text{pole},j}=-\frac{1}{W}\sum_{i=1}^{W}y_i\log\left(\sigma(\text{LN}(z_{\text{p},i}))\right), \tag{5}$$

$$L_{\text{diff}}=\frac{1}{H}\sum_{j=1}^{H}L_{\text{pole},j}, \tag{6}$$

式中：$\boldsymbol{H}$ 为特征高度；$L_{\text{pole},j}$为第j极线上的损失；$z_{\text{p},i}$为极线上第i个特征；y_i为$z_{\text{p},i}$对应的标签；σ为 softmax 运算。特征差异化损失表征了左图像同一极线上像素之间的差异性，差异越大，该损失越小，反之越大。基于特征差异化损失改进的 STTR-light 损失函数具体计算为

$$L=w_1L_{\text{d1,r}}+w_2L_{\text{d1,f}}+w_3L_{\text{be,f}}+w_4L_{\text{rr}}+w_5L_{\text{diff}}, \tag{7}$$

式中：w_1~w_5 是损失函数权重；$L_{\text{d1,r}}$和$L_{\text{d1,f}}$分别是亚分辨率和原始分辨率视差图的平均 Smooth L1 损失；$L_{\text{be,f}}$是遮挡图的交叉熵损失，它表征了预测的遮挡图与真实遮挡图之间的误差。预测的遮挡图越准确，$L_{\text{be,f}}$越小，反之越大；计算为

$$L_{\text{be,f}}=-\frac{1}{N}\sum_{i=1}^{N}\log\left(\sigma(z_{\text{occ},i})\right)-\frac{1}{M}\sum_{j=1}^{M}\log\left(\sigma(z_{\text{noc},j})\right), \tag{8}$$

z_{occ}表示遮挡图中的遮挡区域，N是该区域的像素个数。z_{noc}表示遮挡图中的非遮挡区域，M是该区域的像素个数。L_{rr}是相对响应损失，计算为

$$L_{\text{rr}}=-\frac{1}{N_T}\sum_{i=1}^{N_T}\log\left(t_i\right)-\frac{1}{M_T}\sum_{j=1}^{M_T}\log\left(\tilde{t}_j\right), \tag{9}$$

式中：t_i表示匹配矩阵 $\boldsymbol{T}$ 中匹配像素集合中的元素，N_T是该集合像素总个数。$\tilde{t}_j$表示匹配矩阵 $\boldsymbol{T}$ 中由于遮挡导致不匹配的像素集合中的元素，M_T是该集合像素总个数。L_{rr}目标是最大化对匹配像素的关注。匹配矩阵中匹配像素计算的响应值越高，并且不匹配像素计算的响应值越低，L_{rr}越小，反之越大。

3 实验与分析

实验使用的工作站系统为 Ubuntu 16.04，配备一块 Intel i7-6800 CPU、一块 NVIDIA GeForce Titan X 1080TI 11 GB GPU 和八块 32GB 内存条。模型代码使用 Python 和 Pytorch 构建。

3.1 数据初始化

3.1.1 数据集

Sceneflow[8] 是一个利用 3D 开源软件 Blender 合成的数据集。它包含三个子数据集，分别是 Monkaa、Driving 和 FlyThings3D。其中 FlyThings3D 子集提供了遮挡图，而 Monkaa 和 Driving 子集没有，故本文选择 FlyThings3D 数据集用作本文的预训练数据集。图像大小为960×540，训练集有 21818 对图像 (一对包含一张左图像和一张右图像)。由于平台算力限制，难以训练庞大的整个数据集，故随机采样 2000 对作为本文训练集；验证集有 4248 对，随机采样 500 对作为本文验证集。

KITTI[31] 是一个在真实场景中采集的数据集。其中深度信息由激光雷达扫描得到，因此该数据集提供稀疏的视差图。KITTI 包含 2012 和 2015 子集，为了增加训练样本，本文合并这两个子集，合并后有真实视差图的共有 394 对图像。其中 80% 用作训练，20% 用作验证。另有 790 对图像作为测试集，测试集

未公布真实视差图，图像大小为1242×375。

3.1.2 数据初始化

模型首先在 Sceneflow 上预训练，然后在 KITTI 上微调训练。优化器选用 AdamW。在 Sceneflow 上预训练时学习率保持不变，为 0.0001，共训练 600 个 epochs (简写为 κ，表示训练轮数)。在 KITTI 上微调训练 400 个 epochs，学习率α采用指数衰减策略，初始值α_0为 0.0001，衰减权重γ为 0.99，学习率计算如公式 (10) 所示。

$$\alpha = \alpha_0 \times \gamma^{\kappa} . \tag{10}$$

3.1.3 实验评价指标

在评价实验结果时选择立体匹配的通用评价指标 3 px error、EPE 和 Occ IOU，它们的计算方法如式 (11)、式 (12) 和式 (13) 所示。

$$e_{\text{3pxError}} = \frac{1}{N}\sum_{i=1}^{N}\left(\left|d_i - \hat{d}_i\right| > 3\right), \tag{11}$$

式中，e_{3pxError}表示 3 px error；d表示真实视差值；$\hat{d}$表示预测视差值；N表示像素总个数；$|\cdot|$表示取绝对值；$|\cdot| > 3$表示匹配误差大于 3 像素；i表示像素索引。3 px error 展示了限定误差为 3 像素的情况下错误率，3 px error 越低，模型立体匹配的性能越好。

$$e_{\text{EPE}} = \frac{1}{N}\sum_{i=1}^{N}\left(\left|d_i - \hat{d}_i\right|\right), \tag{12}$$

式中，e_{EPE}表示 EPE，它展示了平均误差像素，是模型估计的视差与真实视差之间的误差平均值。同样 EPE 越低，模型立体匹配的性能越好。

$$\xi_{\text{OccIOU}} = \frac{1}{2}\left(\frac{\sum\limits_{i\in map_{\text{of}}}\left(o_{\text{cc},i}\&\widehat{o}_{\text{cc},i}\right)}{\sum\limits_{i\in map_{\text{of}}}\left(o_{\text{cc},i}|\widehat{o}_{\text{cc},i}\right)} + \frac{\sum\limits_{i\in map_{\text{nof}}}\left(n_{\text{oc},i}\&\widehat{n}_{\text{oc},i}\right)}{\sum\limits_{i\in map_{\text{nof}}}\left(n_{\text{oc},i}|\widehat{n}_{\text{oc},i}\right)}\right), \tag{13}$$

式中：ξ_{OccIOU}表示 Occ IOU；map_{of}和map_{nof}分别表示遮挡图和非遮挡图；o_{cc}和$\hat{o}_{\text{cc}}$分别表示遮挡图中布尔型的真实值和预测值；n_{oc}和$\hat{n}_{\text{oc}}$表示非遮挡图中布尔型的真实值和预测值；对于布尔型的值，&表示与运算；|表示或运算。

3.2 消融实验与分析

3.2.1 消融实验分析

为了验证所提方法的有效性，我们在 Sceneflow 上进行了 4 组消融实验。第一组实验，STTR-light 模型。第二组实验，将 STTR-light 的特征提取模块改为基于 Swin Transformer 的特征提取模块。第三组实验，在第二组实验的基础上去掉 STTR-light 的 Self and Cross-attention，相应部分改用相关运算。第四组实验，在第三组实验的基础上加入特征差异化损失函数。实验结果如表 1 所示。

表中加粗数值表示效果最好的，把基于 CNN 的特征提取器替换为基于 Swin Transformer 的模块后，3 px error 指标降低 0.32%，EPE 指标降低 0.08。表明 Swin Transformer 对远程上下文信息的学习能够提高匹配精度。使用相关运算替换 Self and Cross-attention 后，3 px error 增加 0.04%，EPE 增加 0.03，两个指标变化不是很剧烈，可见相关运算对模型性能影响较小。加入特征差异化损失后，3 px error 降低至 1.03%，EPE 降低到 0.42，模型性能达到最佳状态，表明差异化的特征确实有助于提高模型的立体匹配性能。

3.2.2 不同损失权重对立体匹配性能的影响分析

如式 (7) 所示，模型总损失是五个损失加权和，设置不同的损失权重，在 KITTI 数据集上实验结果如表 2 所示。

实验结果表明不同的损失权重组合也会对模型性能产生影响。当各个权重相等时，Occ IOU 最佳，为 0.97。若把 $L_{\text{dl,f}}$ 的权重设置最高时，$L_{\text{dl,r}}$ 次之，模型的综合性能最佳，3 px error 为 0.84，EPE 为 0.39，Occ IOU 为 0.96。

3.2.3 特征差异化损失分析

为了验证特征差异化损失在提取图像特征过程中的作用，本文对 Swin Transformer 模块提取的特征进

表 1 消融实验

Table 1 Ablation study

实验	基于Swin Transformer模块	相关运算	特征差异化损失	3 px error / % ↓	EPE ↓	Occ IOU ↑
第1组				1.68	0.56	0.94
第2组	√			1.36	0.48	0.96
第3组	√	√		1.40	0.51	0.95
第4组	√	√	√	**1.03**	**0.42**	**0.97**

表 2 不同的损失权重实验结果

Table 2 Experimental results of different loss weights

$L_{d1,r}$	$L_{d1,f}$	L_{rr}	$L_{be,f}$	L_{diff}	3 px error /%↓	EPE ↓	Occ IOU ↑
0.2	0.2	0.2	0.2	0.2	0.85	0.41	**0.97**
0.2	0.2	0.2	0.1	0.3	0.93	0.51	0.84
0.3	0.3	0.1	0.1	0.2	0.89	0.43	0.85
0.3	0.4	0.1	0.1	0.1	**0.84**	**0.39**	0.96
0.4	0.3	0.1	0.1	0.1	0.87	0.40	0.91

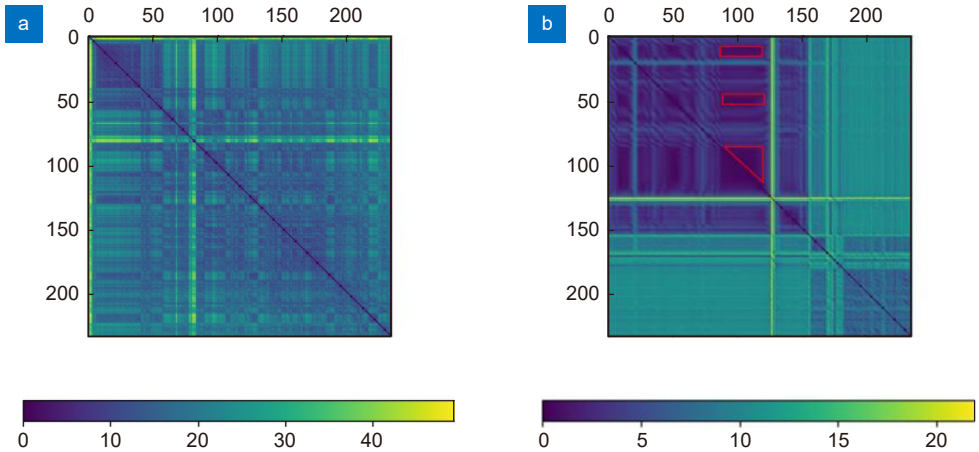


图 3 左图像上的像素特征之间的欧式距离。(a) 加入特征差异化损失; (b) 未加入特征差异化损失

Fig. 3 Euclidean distance between the pixel features on the left image.
(a) There is a feature differentiation loss; (b) No a feature differentiation loss

行可视化分析。取左图像一极线上提取的像素特征，计算各像素特征之间的欧氏距离如图 3 所示。

图 3(a) 和 3(b) 分别展示了加入和未加入特征差异化损失提取的特征之间的欧氏距离。颜色越亮，表示距离越大，相似性越高。图中斜对角线是像素特征与自己的距离，值为 0。其他位置是同极线上各像素特征之间的距离。图 3(a) 中，各列色彩明暗不同，可见像素特征之间的特征均有差异，最大的欧氏距离超过了 40。图 3(b) 中，第 100 列附近的红框中存在色彩相同的区域，可见像素之间的特征存在相同或者相近状况，最大欧氏距离小于 30。像素特征差异越大，越容易相互区分，越有利于像素匹配。

3.3 对比实验与分析

3.3.1 泛化性能对比分析

为了评估所提模型的泛化性能，使用预训练的权重在 MPI Sintel、KITTI、Middlebury2014 和 SCARED 数据集上进行泛化性能对比实验。对比方法包括 PSMNet、AANet、STTR-light 和 STTR。这几个方法都是近些年基于深度学习的具有代表性的端到端立体匹配网络。实验结果如表 3 和表 4 所示。

表 3 模型泛化性能实验结果 (一)

Table 3 Test results Ⅰ of model generalization performance

模型	MPI Sintel			KITTI		
	3 px error /% ↓	EPE ↓	Occ IOU ↑	3 px error /% ↓	EPE ↓	Occ IOU ↑
PSMNet[10]	6.81	3.31	N/A	27.79	6.56	N/A
AANet[11]	5.91	**1.89**	N/A	12.42	1.99	N/A
STTR-light[19]	5.82	2.95	0.69	7.2	1.56	0.95
STTR[19]	5.75	3.01	**0.86**	6.74	1.50	**0.98**
本文算法	**5.23**	2.78	0.84	**6.51**	**1.44**	0.97

表 4 模型泛化性能实验结果 (二)

Table 4 Test results Ⅱ of model generalization performance

模型	Middlebury			SCARED		
	3 px error /% ↓	EPE ↓	Occ IOU ↑	3 px error /% ↓	EPE ↓	Occ IOU ↑
PSMNet[10]	12.96	3.05	N/A	OOM	OOM	N/A
AANet[11]	12.80	2.19	N/A	6.39	1.36	N/A
STTR-light[19]	**5.36**	**2.05**	0.76	3.30	**1.19**	0.89
STTR[19]	6.19	2.33	0.95	3.69	1.57	**0.96**
本文算法	6.24	2.12	**0.96**	**3.15**	1.26	0.94

表 4 中，OOM 表示内存超出，实验结果表明本文模型在未训练的数据集上具有较好的泛化性能。其中在 MPI Sintel、KITTI 和 SCARED 数据集上，本文算法的 3 px error 均为最低值；在 Middlebury 数据集上，3 px error 比 STT-light 高，但是 Occ IOU 超过了其他模型。

3.3.2 微调训练后对比分析

本文在 KITTI 和 Sceneflow 数据集上做了多个方法的对比实验。模型首先加载在 Sceneflow 上训练的权重文件，然后在 KITTI 进行了 400 个 epochs 的微调训练[32]。实验结果如表 5 所示。

在 Sceneflow 和 KITTI 数据集上实验结果表明，我们提出的方法相较于之前的方法 3 px error 和 EPE 均有降低。比较 3 px error，本文算法获得了最好的成绩，在提供稠密视差图的 Sceneflow 数据集上，3 px error 降低至 1.03%，比 PSMNet 和 AANet 分别降低 2.91% 和 2.86%，比改进前的 STTR-light 模型降低 0.81%；比 STTR 模型降低 0.30%，在提供稀疏视差图的 KTTI 数据集上，3 px error 降低至 0.84%，是所有方法中最低的。就 EPE 指标而言，本文算法同样最优，在合成的 Sceneflow 上，EPE 降低到 0.42，比 STTR-light 模型降低 0.14，比 STTR 降低 0.06。在 KTTI 数据集上，本文算法的 EPE 为 0.39，比 STTR-light 模型降低 0.17，比 STTR 降低 0.05。对比 Occ IOU，本文算法在 Sceneflow 上比 STTR-light 低，在 KITTI 上比 STTR 低。实验结果表明，本文算法综合匹配性能表现最好，这是因为在立体匹配的过程中，匹配像素特征越独特，匹配效果越好。在局部无纹理或者弱纹理区域，只融合局部特征无法提取出区别于周围邻域像素的特征。只有融合全局特征，才能够提取出更好的匹配特征。本文算法结合基于 Swin Transformer 的特征提取模块和特征差异化损失使得提取的特征具有良好的分辨性，进一步提高了立体匹配的精度。

3.3.3 模型运行效率对比分析

本文在 KITTI 上，对模型的运行效率从参数量 (Params)、计算量 (FLOPs)、运行显存 (Memory) 和运行时间 (Runtime) 四个方面进行了对比，实验结果如表 6 所示。

实验结果表明，STTR-light 的参数量、计算量和显存占用是最少的。STTR-light 和 PSMNet 的特征提取器类似，但是 STTR-light 视差计算和优化部分采用参数量更少的二维卷积，所以 STTR-light 模型比 PSMNet 小。相比于 STTR，STTR-light 采用更大的下采样倍率得到亚分辨率图像，所以 STTR-light 运行效率比 STTR 高。对比运行时间，AANet 运行时间最

表 5 对比试验结果

Table 5 Comparative experiments

模型	Sceneflow			KITTI		
	3 px error / % ↓	EPE ↓	Occ IOU ↑	3 px error / % ↓	EPE ↓	Occ IOU ↑
PSMNet[10]	3.94	1.11	N/A	1.25	0.57	N/A
AANet[11]	3.89	0.82	N/A	1.93	0.64	N/A
STTR-light[19]	1.84	0.56	**0.98**	1.68	0.56	0.94
STTR[19]	1.43	0.48	0.91	1.12	0.44	**0.97**
本文算法	**1.03**	**0.42**	0.97	**0.84**	**0.39**	0.96

表 6 模型运行效率对比

Table 6 Comparison of model operation efficiency

模型	Params ↓ / M	FLOPs ↓ / G	Memory ↓ / G	Runtime ↓ / s
PSMNet[10]	5.22	613.90	4.08	0.63
AANet[11]	3.68	119.64	1.63	**0.09**
STTR-light[19]	**2.33**	**110.21**	**0.43**	0.65
STTR[19]	2.51	510.93	1.23	0.67
本文算法	27.85	136.11	2.90	0.73

少。因为本文算法引入参数量较大的 Swin Transformer，致使参数量和运行时间最大，但计算量比 STTR 和 PSMNet 少，运行显存低于 PSMNet。

3.3.4 可视化对比分析

本文在 Sceneflow 和 KITTI 上，对不同场景进行了可视化，如图 4 和图 5 所示。

由于 STTR-light 和本文模型在估计视差时也预测了遮挡图，图 4 中这两个模型预测的视差图体现了观察者或者双目相机观察目标时的遮挡信息 (图中的阴影)。真实视差图中也存在遮挡。而 PSMNet 模型估计的视差图没有体现遮挡。通过观察可知，有遮挡信息的视差图更加真实。图 4 第一行图的红框部分是一个车轮，PSMNet 估计的视差图比较平滑，但是丢失车轮里面的细节信息，比如车辐条。而 STTR-light 模型估计的视差图中可以看到一些杂乱的车辐条，而本文方法估计的视差图中的车辐条较为清晰完整。图 4 第二行中有一块模糊的绿色方块，PSMNet 模型的预测出的视差图中方块的棱角模糊不清，STTR-light 模型预测的虽然有棱角，但是棱角周边存在噪点；本文方法预测的棱角明显，平面平滑完整。在顶部红框中两个方块的交界处，PSMNet 模型输出的视差图连接到一块了；STTR-light 模型输出的视差图界限交错不明显；本文算法输出的视差图交界清晰明了。

图 5 第一行图的红框里面是一个骑自行车的女人，从左图像中可以看出这部分中间存在很多间隙。但是 PSMNet 和 STTR-light 方法预测的视差图中，自行车和女人被预测为一个整体，丢失了细节信息。本文算法估计的视差图中不仅可以看到大体轮廓还可以看到车把手和胳膊等细节。图 5 第二行图像的左上角存在一个红绿灯，PSMNet 和 STTR-light 方法预测的视差图中，该部分特别模糊，而本文算法估计的视差图的该部分是一个较为完整的长方形。

在合成的数据集和真实世界数据集上测试结果表

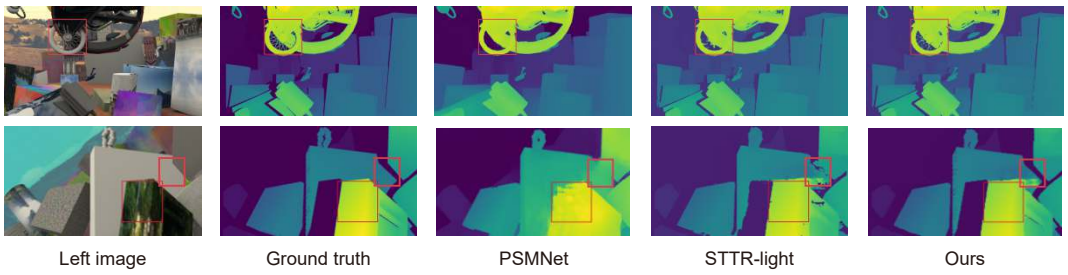


图 4 不同方法在 Sceneflow 数据集上的估计的视差图

Fig. 4 Disparity map estimated by different methods on the Sceneflow datasets

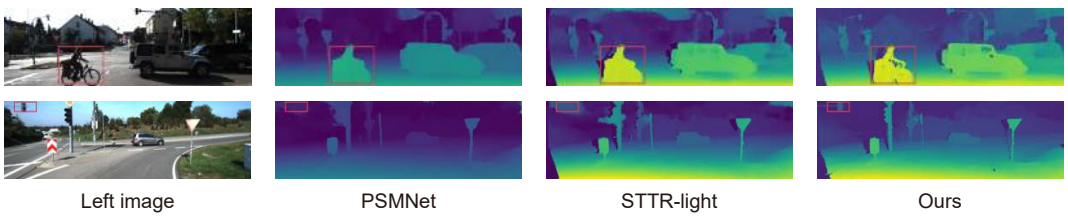


图 5 不同方法在 KITTI 数据集上的估计的视差图

Fig. 5 Disparity map estimated by different methods on the KITTI datasets

明，STTR-light 输出的视差图比 PSMNet 输出的细节更加丰富。一方面是因为 STTR-light 模型估计的视差图中加入了遮挡信息，另一方面是因为模型中的 Self and Cross-attention 模块计算了左右图像极线上的注意力。而本文模型输出的视差图比 STTR-light 更加详细，是因为特征差异化损失强制细化了图像特征。

4 结　论

本文提出 STransMNet，改进了立体匹配网络的特征提取模块、Swin Transformer 和损失函数。实验证明，STransMNet 降低了匹配误差，提升了视差图质量，表明改进的 Swin Transformer 模块捕获远距离上下文信息的优异性能有助于提升立体匹配的精度；特征差异化损失有助于增强视差图的细节信息。虽然本文算法匹配性能较好，但是参数量较大，推理时间较长，总体运行效率不高。为了提高本文算法的时效性，后续工作将引入知识蒸馏方法来提高模型的运行效率；同时，为了在训练时加入多元化无标签的数据，后续工作还需引入无监督或自监督知识来提高模型的泛化能力。

参考文献

[1] Li X, Li L P, Lazovik A, et al. RGB-D object recognition algorithm based on improved double stream convolution recursive neural network[J]. *Opto-Electron Eng*, 2021, **48**(2): 200069.
李珣, 李林鹏, Lazovik A, 等. 基于改进双流卷积递归神经网络的RGB-D物体识别方法[J]. 光电工程, 2021, **48**(2): 200069.

[2] Hoffman J, Gupta S, Leong J, et al. Cross-modal adaptation for RGB-D detection[C]//*2016 IEEE International Conference on RoboticsandAutomation(ICRA)*,2016:5032–5039.https://doi.org/10.1109/ICRA.2016.7487708.

[3] Schwarz M, Milan A, Periyasamy A S, et al. RGB-D object detection and semantic segmentation for autonomous manipulation in clutter[J]. *Int J Robot Res*, 2018, **37**(4–5): 437–451. https://doi.org/10.1177/0278364917713117.

[4] Cao C L, Tao C B, Li H Y, et al. Deep contour fragment matching algorithm for real-time instance segmentation[J]. *Opto-Electron Eng*, 2021, **48**(11): 210245.
曹春林, 陶重犇, 李华一, 等. 实时实例分割的深度轮廓段落匹配算法[J]. 光电工程, 2021, **48**(11): 210245.

[5] Scharstein D, Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms[J]. *Int J Comput Vision*, 2002, **47**(1–3): 7–42. https://doi.org/10.1023/A:1014573219977.

[6] Tippetts B, Lee D J, Lillywhite K, et al. Review of stereo vision algorithms and their suitability for resource-limited systems[J]. *J Real-Time Image Proc*, 2016, **11**(1): 5−25.

[7] Hirschmuller H. Stereo processing by semiglobal matching and mutual information[J]. *IEEE Trans Pattern Anal Mach Intell*, 2008, **30**(2): 328−341.

[8] 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*, 2016: 4040–4048. https://doi.org/10.1109/CVPR.2016.438.

[9] Kendall A, Martirosyan H, Dasgupta S, et al. End-to-end learning of geometry and context for deep stereo regression[C]//*Proceedings of the IEEE International Conference on Computer Vision*, 2017: 66–75. https://doi.org/10.1109/ICCV.2017.17.

[10] Chang J R, Chen Y S. Pyramid stereo matching network[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018: 5410–5418. https://doi.org/10.1109/CVPR.2018.00567.

[11] Xu H F, Zhang J Y. AANet: Adaptive aggregation network for efficient stereo matching[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020: 1956–1965. https://doi.org/10.1109/CVPR42600.2020.00203.

[12] Khamis S, Fanello S, Rhemann C, et al. StereoNet: Guided hierarchical refinement for real-time edge-aware depth prediction[C]//*Proceedings of the 15th European Conference on Computer Vision*, 2018: 596–613. https://doi.org/10.1007/978-3-030-01267-0_35.

[13] Chopra S, Hadsell R, LeCun Y. Learning a similarity metric discriminatively, with application to face verification[C]//*2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005, **1**: 539–546. https://doi.org/10.1109/CVPR.2005.202.

[14] He K M, Zhang X Y, Ren S Q, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. *IEEE Trans Pattern Anal Mach Intell*, 2015, **37**(9): 1904−1916.

[15] Zhao H S, Shi J P, Qi X J, et al. Pyramid scene parsing network[C]//*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017: 6230–6239. https://doi.org/10.1109/CVPR.2017.660.

[16] Nie G Y, Cheng M M, Liu Y, et al. Multi-level context ultra-aggregation for stereo matching[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019: 3278–3286. https://doi.org/10.1109/CVPR.2019.00340.

[17] Zhang F H, Prisacariu V, Yang R G, et al. Ga-Net: guided aggregation net for end-to-end stereo matching[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019: 185–194. https://doi.org/10.1109/CVPR.2019.00027.

[18] Chabra R, Straub J, Sweeney C, et al. StereoDRNet: dilated residual StereoNet[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019: 11778–11787. https://doi.org/10.1109/CVPR.2019.01206.

[19] Li Z S, Liu X T, Drenkow N, et al. Revisiting stereo depth estimation from a sequence-to-sequence perspective with Transformers[C]//*Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021: 6177–6186. https://doi.org/10.1109/ICCV48922.2021.00614.

[20] Tulyakov S, Ivanov A, Fleuret F. Practical deep stereo (PDS): toward applications-friendly deep stereo matching[C]//*Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018: 5875–5885. https://doi.org/10.5555/3327345.3327488.

[21] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018: 7132–7141. https://doi.org/10.

1109/CVPR.2018.00745.

[22] Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module[C]//*Proceedings of the 15th European Conference on Computer Vision*, 2018: 3–19. https://doi.org/10.1007/978-3-030-01234-2_1.

[23] Dosovitskiy A, Beyer L, Kolesnikov A, et al. An image is worth 16x16 words: transformers for image recognition at scale[C]//*9th International Conference on Learning Representations*, 2021.

[24] Han K, Xiao A, Wu E H, et al. Transformer in transformer[C]//*Proceedings of the 35th Conference on Neural Information Processing Systems*, 2021: 15908–15919.

[25] Fang Y X, Liao B C, Wang X G, et al. You only look at one sequence: rethinking transformer in vision through object detection[C]//*Proceedings of the 35th Conference on Neural Information Processing Systems*, 2021: 26183–26197.

[26] 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*, 2021: 9992–10002. https://doi.org/10.1109/ICCV48922.2021.00986.

[27] Courty N, Flamary R, Tuia D, et al. Optimal transport for domain adaptation[J]. *IEEE Trans Pattern Anal Mach Intell*, 2017, **39**(9): 1853−1865.

[28] Liu Y B, Zhu L C, Yamada M, et al. Semantic correspondence as an optimal transport problem[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020: 4463–4472. https://doi.org/10.1109/CVPR42600.2020.00452.

[29] Sarlin P E, DeTone D, Malisiewicz T, et al. Superglue: learning feature matching with graph neural networks[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020: 4937–4946. https://doi.org/10.1109/CVPR42600.2020.00499.

[30] Cao H, Wang Y Y, Chen J, et al. Swin-Unet: Unet-like pure Transformer for medical image segmentation[C]//*Proceedings of the International Conference on Computer Vision*, 2022: 205–218. https://doi.org/10.1007/978-3-031-25066-8_9.

[31] Menze M, Geiger A. Object scene flow for autonomous vehicles[C]//*Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015: 3061–3070. https://doi.org/10.1109/CVPR.2015.7298925.

[32] He K M, Girshick R, Dollár P. Rethinking ImageNet pre-training[C]//*Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019: 4917–4926. https://doi.org/10.1109/ICCV.2019.00502.

作者简介

王高平 (1994-)，男，硕士研究生，主要研究方向为立体视觉。

E-mail: 200421119@stu.xpu.edu.cn

【通信作者】李珣 (1981-)，男，博士，副教授，主要研究方向为基于深度学习的多移动目标检测与跟踪以及多运动机器人协同控制技术。

E-mail: lixun@xpu.edu.cn

贾雪芳 (1996-)，女，硕士研究生，主要研究方向为点云关键点提取。

E-mail: 819032030@qq.com

李哲文 (1996-)，男，硕士研究生，主要研究方向为机器人控制与 SLAM。

E-mail: 519095004@qq.com

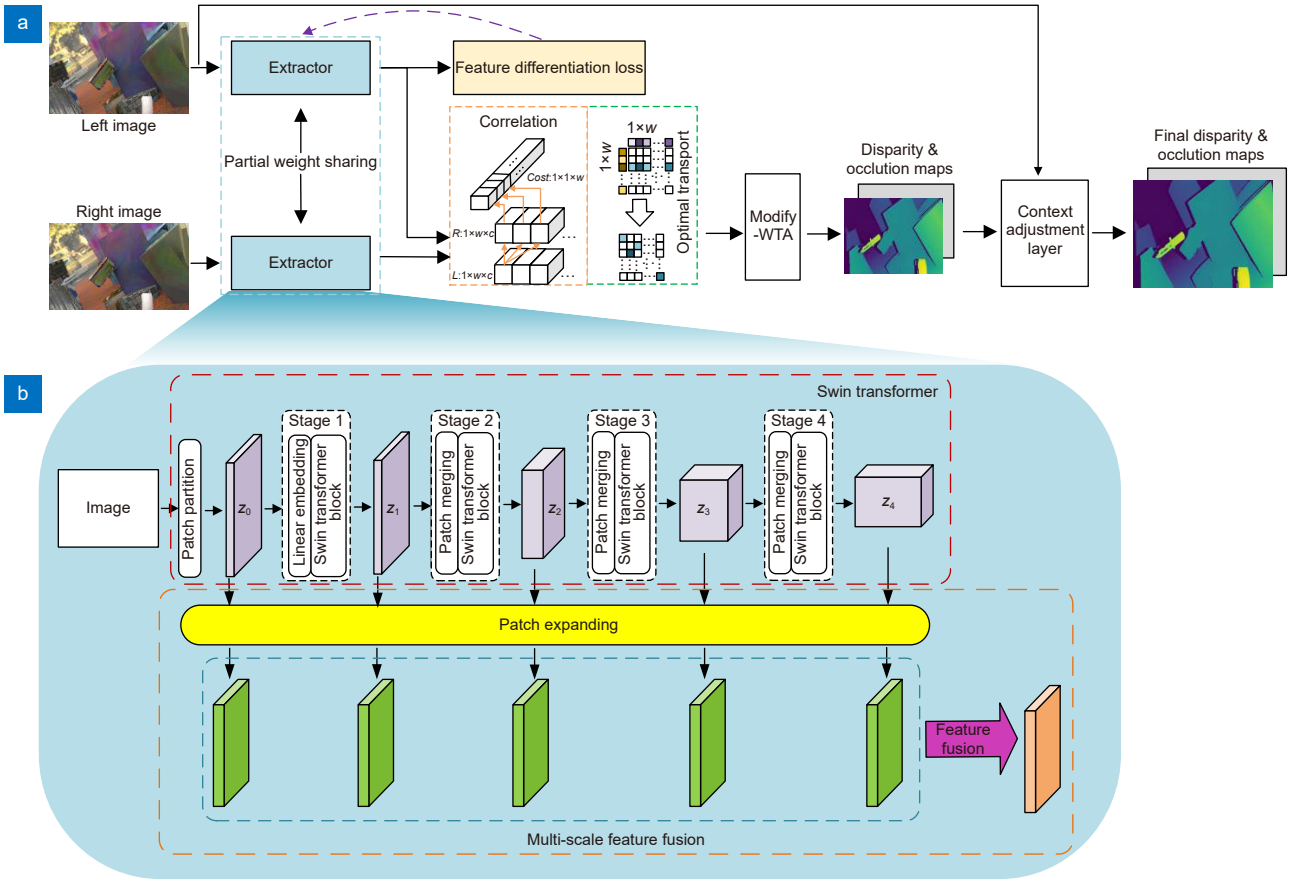
王文杰 (1988-)，男，博士，主要研究方向为机器人和智能医疗装备技术。

E-mail: wangwenjie@xpu.edu.cn

扫描二维码，获取PDF全文

STransMNet: a stereo matching method with swin transformer fusion

Wang Gaoping[1], Li Xun[1,2*], Jia Xuefang[1], Li Zhewen[1], Wang Wenjie[1]



The pipeline of STransMNet

Overview: In the stereo matching process, unique pixel features are extracted by aggregating local and global context information. The pixel features on the pole lines of the left and right images are then matched. With the rapid application of deep learning (DL) methods in the field of image processing, end-to-end neural networks of DL are used to estimate the disparity maps. Although CNN-based algorithms have excellent feature representation capabilities, they often exhibit limitations in modeling explicit long-range relationships due to the inherent locality of the convolution operations. For objects with weak textures and large differences in shape and size, the results of using CNN alone are often unsatisfactory. To solve this problem, an improved model STransMNet stereo matching network based on the Swin Transformer is proposed in this paper. We analyze the necessity of the aggregated local and global context information. Then the difference in matching features during the stereo matching process is discussed. The feature extraction module is improved by replacing the CNN-based algorithm with the Transformer-based Swin Transformer algorithm. The rectified left and right images are fed into Swin Transformer module to generate multi-scale features. Then the multi-scale features are fed into the patch expanding module, the transformation of the linear layer, to make them the same size. Finally, the multi-scale features are fused in the channel dimension. The additional multi-scale fusion module makes the features output by the improved Swin Transformer fuse shallow and deep semantic

Foundation item: National Natural Science Foundation of China (61971339) and Shaanxi Natural Science Basic Research Project (2022JM407).
[1]School of Electronics and Information, Xi'an Polytechnic University, Xi'an, Shaanxi 710600, China; [2]Xi'an Polytechnic University Branch of Shaanxi Artificial Intelligence Joint Laboratory, Xi'an, Shaanxi 710600, China

* E-mail: lixun@xpu.edu.cn

information. The Swin Transformer used to extract the left and right image features is partially shared by the weights. Although weight sharing makes the model converge faster, our proposed feature differentiation loss can only supervise left or right images. If the full weights are shared, it is equivalent to supervising the left and right images at the same time. Partial weight sharing speeds up the convergence of the model to a certain extent. In addition, partial weight sharing enables the model to extract not only the commonalities of left and right image but also the differences. Furthermore, a feature differentiation loss is proposed in this work to improve the model's ability to pay attention to details. The loss is trained by forcing the classification of pixel features on the epipolar line of the left image, which makes each pixel feature unique. The experimental results on the Sceneflow and KITTI datasets show that our algorithm reduces the 3 px error and EPE compared to the previous algorithms. Experiments show that the proposed STransMNet model reduces the matching error and improves the quality of the disparity maps. It shows that the excellent performance of the improved Swin Transformer in capturing long-distance context information is beneficial to improving the accuracy of stereo matching; feature differentiation loss helps to enhance the detailed information of the disparity maps.

Wang G P, Li X, Jia X F, et al. STransMNet: a stereo matching method with swin transformer fusion[J]. *Opto-Electron Eng*, 2023, **50**(4): 220246; DOI: 10.12086/oee.2023.220246