第 58 卷 第 24 期/2021 年 12 月/激光与光电子学进展 研究论文

激光与光电子学进展 先进成像

一种优化的深度学习立体匹配算法

黄继辉，张荣芬，刘宇红*，陈至栩，王子鹏

贵州大学大数据与信息工程学院，贵州 贵阳 550025

摘要 现如今用于立体匹配的深度学习算法都存在网络结构复杂、消耗大的问题。为解决此类问题，提出了一种参数量只有参考网络 PSMNet 一半的立体匹配端到端网络结构。所提结构在特征提取模块保留大致框架的同时，减少多余卷积层，并融合空间注意力机制和通道注意力机制来汇聚上下文信息；在代价计算模块，通过加大偏移步长减少视差计算输入的视差维度，使视差计算的参数量和消耗大幅度减少；在视差计算中，对匹配成本特征体的输出进行多视差预测；在 L1 损失函数的基础上加入交叉熵损失函数，这样可在降低消耗的同时保证了模型匹配精度。在 KITTI 数据集和 SceneFlow 数据集上对所提模型进行测试，实验结果表明：与基准方法相比，所提模型的参数量减少了 58%，精度提升 24%。

关键词 视觉光学；立体匹配；端到端网络；注意力机制；视差计算

中图分类号 TP391 **文献标志码** A **doi**：10.3788/LOP202158.2433002

Optimized Deep Learning Stereo Matching Algorithm

Huang Jihui, Zhang Rongfen, Liu Yuhong*, Chen Zhixu, Wang Zipeng

College of Big Data and Information Engineering, Guizhou University, Guiyang, Guizhou 550025, *China*

Abstract Nowadays, deep learning algorithms used for stereo matching have the problems of complex network structure and high consumption. In order to solve such problems, an end to end stereo matching network structure with only half the parameters of the reference network PSMNet is proposed. In the feature extraction module of the proposed network, the general framework is retained, its redundant convolutional layers are reduced, and meanwhile the spatial attention mechanism and channel attention mechanism are integrated to gather contextual information. In the cost calculation module, the input disparity dimension of the disparity calculation is reduced by increasing the offset, and therefore, the parameter amount and consumption of disparity calculation are greatly reduced. In the disparity calculation, the multi-disparity prediction is performed for the output of the matching cost feature body. And the cross-entropy loss function is added to the L1 loss function, which ensures the matching accuracy when reducing the consumption of the model. The proposed algorithm is tested on the KITTI dataset and SceneFlow dataset. The experimental results show that compared with the benchmark method, the parameter amount of the proposed model is reduced by 58% while the accuracy is increased by 24%.

Key words visual optics; stereo matching; end-to-end network; attention mechanism; disparity calculation

OCIS codes 330.1400; 150.5670; 150.1135

1 引 言

和传统传感器测距相比，通过双目摄像头进行测距具有隐蔽性好、稳定性好等优点，其原理是：通过计算得到空间某一点在左、右两幅画面中的视差值，根据双目成像原理即可得到该点的物理距离。在左、右画面中寻找相似点的过程就称为立体匹配。双目立体匹配是获取周围环境信息的重要手段，现

收稿日期：2020-12-14；**修回日期**：2021-01-28；**录用日期**：2021-03-08

基金项目：贵州省科技计划项目(黔科合平台人才[2016]5707)

通信作者：*liuyuhongyx@sina.com

在已广泛应用于三维重建、医疗影像、增强现实(AR)等方面。立体匹配算法分为传统算法和深度学习算法，传统算法包括如 BM(Block Matching)[1]、SGM(Semi-Global Matching)[2]、Census[3-4] 等，这类算法在弱纹理区域、遮挡区域得到的视差图效果很不理想，且在精度和速度方面也不如人意。深度学习的方法通过卷积、池化等操作，可以获得多层次的特征用于代价计算，使用端到端的深度学习网络进行图像匹配，可提高算法的鲁棒性，该方法在速度和精度也都优于传统算法。2015 年 Žbontar 和 LeCun[5] 将深度学习应用到立体匹配中，提出了基于 MC-CNN (Matching Cost with a Convolutional Neural Network)的网络结构进行代价计算，这标志着卷积神经网络(CNN)在立体匹配方面应用的开端。之后 Luo 等[6] 提出了基于孪生网络的方法，该方法将视差计算看作一种多标签分类问题。Mayer 等[7] 提出了使用端到端的网络 DispNet 进行立体匹配，将立体匹配的整个流程通过一个网络实现，通过输入一对图像就可以直接计算出视差图，并将一个大型数据集 SceneFlow 用于网络训练。Kendall 等[8] 提出 GC-Net(Geometry and Context Network)，将提取完的特征经过 3D 卷积，利用 3D 卷积可以实现在优化匹配成本特征体的同时进行视差计算，进而回归出最终的视差图。Chang 和 Chen[9] 提出的 PSMNet(Pyramid Stereo Matching Network)，在以 RESNet 为主干网络的特征提取模块中加入特征金字塔池化(Spatial Pyramid Pooling)模块，以更好地利用全局上下文信息，在视差计算阶段加入 3D 卷积并利用 Stackhourglass 和 Basic 结构显著改善弱纹理和光照不均匀区域的视差图效果。Zhang 等[10] 提出引入半全局聚合层和局部引导聚合层的 GA-Net(Guided Aggregation Net)，通过将传统方法和现代方法相结合，计算效率得到提高。但这些方法都存在网络结构复杂、消耗大的问题，本文主要针对该问题将特征提取模块进行简化，删除多余层以减小卷积核，同时通过引入注意力机制减少 3D 卷积中的视差维度，并对 3D 卷积的输出进行多视差预测，从而在保证高精度的同时提高了效率。

2　基于深度学习的立体匹配

将深度学习的理念用于立体匹配的步骤如下：首先将左右图像输入到两个权重共享的 Siamese 特征提取网络，Siamese 特征提取网络大部分都是以 RESNet[11] 为基础网络。这里的特征提取是为了找出像素点之间的联系。接着通过移动两个特征图进行代价计算，通过特征融合构建匹配成本特征体。然后将成本特征体输入到一个堆叠沙漏型的 3DCNN 进行代价聚合和视差计算。最后通过上采样视差回归得到精确的视差图。本文以 PSMNet 为基准网络，如图 1 所示。

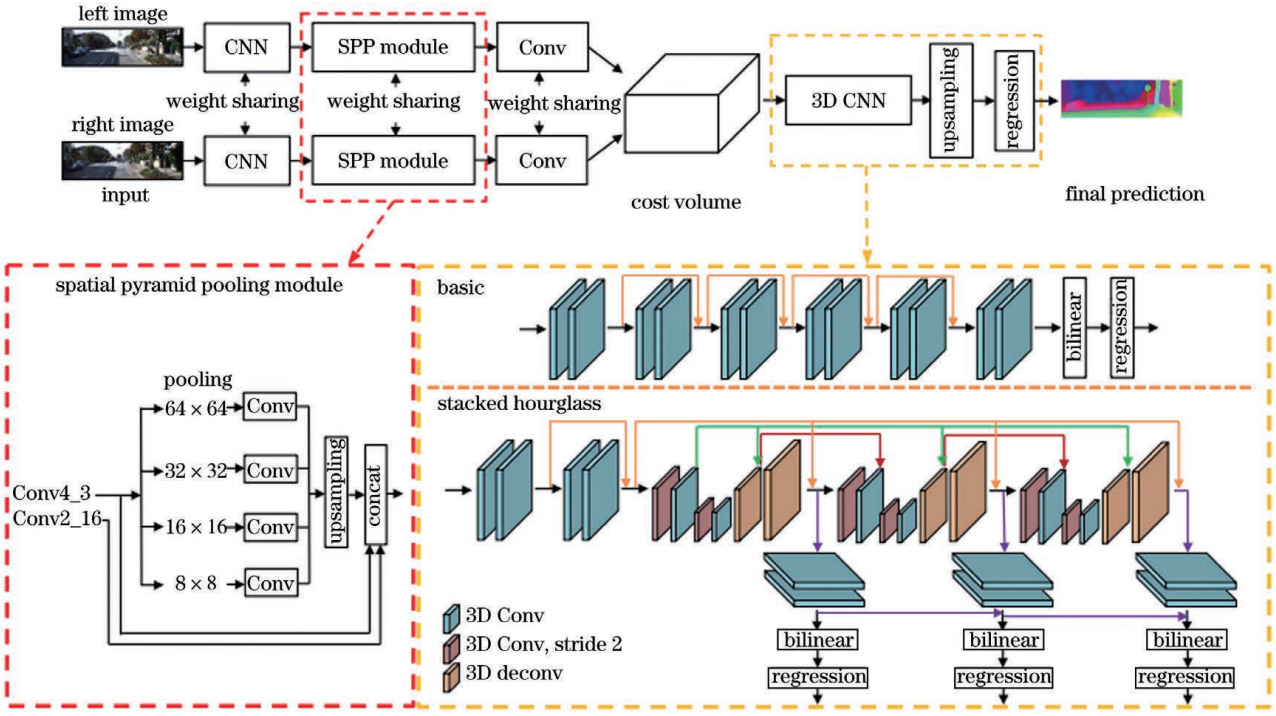

图 1　原网络结构

Fig. 1　Original network structure

在此基础上进行改进，改进结构如图 2 所示，首先通过 CNN 提取特征，在特征提取中引入注意力机制，将提取到的左右特征进行代价计算（cost volume），生成匹配成本特征体，然后将其输入 3DCNN 进行视差计算，最后通过上采样和视差回归得到视差图。

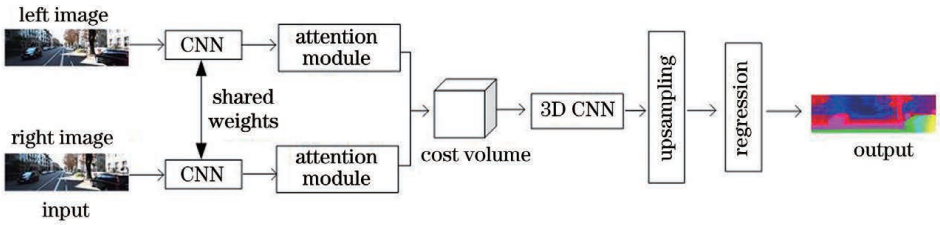

图 2　本文提出的网络结构

Fig. 2　Network structure proposed in this article

2.1　RESNet 简化

PSMNet 的特征提取模块采用一个完整的 RESNet，RESNet 用来提取高度抽象的特征以表示语义信息，而立体匹配任务用不到语义信息，因此不需要很深的特征提取网络。受文献[12]启发，利用小型的特征提取结构也可以获得较为精准的视差图，由此试图建立一个结构简单、参数少、性能同样好的网络，其具体结构参数如表 1 所示。

表 1　所提的具体网络结构

Table 1　Specific network structure mentioned

Network	Layer	Setting	Output
Feature extraction	Layer0_1	3×3, 32	$\frac{1}{2}H\times\frac{1}{2}W\times32$
	Layer0_2	1×1, 32	$\frac{1}{2}H\times\frac{1}{2}W\times32$
	Layer1_x	1×1, 32 1×1, 32	$\frac{1}{2}H\times\frac{1}{2}W\times32$
	Layer2_x	3×3, 64 3×3, 64 (4 pairs)	$\frac{1}{4}H\times\frac{1}{4}W\times64$
	Layer3_x	1×1, 128 1×1, 128	$\frac{1}{4}H\times\frac{1}{4}W\times128$
	Attention mode	Channel, spatial	$\frac{1}{4}H\times\frac{1}{4}W\times128$
	Layer4	1×1, 32	$\frac{1}{4}H\times\frac{1}{4}W\times32$
Cost volume	Cascade		$\frac{1}{4}H\times\frac{1}{4}W\times\frac{1}{8}D\times64$
3DCNN	3DLayer0	3×3×3, 32 3×3×3, 32	$\frac{1}{4}H\times\frac{1}{4}W\times\frac{1}{8}D\times32$
	3DLayer1	3×3×3, 32 3×3×3, 32	$\frac{1}{4}H\times\frac{1}{4}W\times\frac{1}{8}D\times32$
	3DStack1_1	3×3×3, 64 3×3×3, 64	$\frac{1}{8}H\times\frac{1}{8}W\times\frac{1}{16}D\times64$
	3DStack1_2	3×3×3, 64 3×3×3, 64	$\frac{1}{16}H\times\frac{1}{16}W\times\frac{1}{32}D\times64$
	3DStack1_3	3×3×3, 64 (deconv)	$\frac{1}{8}H\times\frac{1}{8}W\times\frac{1}{16}D\times64$

表 1 续

Network	Layer	Parameter	Output
3DCNN	3DStack1_4	3×3×3, 32 (deconv)	$\frac{1}{4}H\times\frac{1}{4}W\times\frac{1}{8}D\times32$
	3DStack2_1	3×3×3, 64 3×3×3, 64	$\frac{1}{8}H\times\frac{1}{8}W\times\frac{1}{16}D\times64$
	3DStack2_2	3×3×3, 64 3×3×3, 64	$\frac{1}{16}H\times\frac{1}{16}W\times\frac{1}{32}D\times64$
	3DStack2_3	3×3×3, 64 (deconv)	$\frac{1}{8}H\times\frac{1}{8}W\times\frac{1}{16}D\times64$
	3DStack2_4	3×3×3, 32 (deconv)	$\frac{1}{4}H\times\frac{1}{4}W\times\frac{1}{8}D\times32$
	3DStack3_1	3×3×3, 64 3×3×3, 64	$\frac{1}{8}H\times\frac{1}{8}W\times\frac{1}{16}D\times64$
	3DStack3_2	3×3×3, 64 3×3×3, 64	$\frac{1}{16}H\times\frac{1}{16}W\times\frac{1}{32}D\times64$
	3DStack3_3	3×3×3, 64 (deconv)	$\frac{1}{8}H\times\frac{1}{8}W\times\frac{1}{16}D\times64$
	3DStack3_4	3×3×3, 32 (deconv)	$\frac{1}{4}H\times\frac{1}{4}W\times\frac{1}{8}D\times32$
	Classify	3×3×3,32 3×3×3,2	$\frac{1}{4}H\times\frac{1}{4}W\times\frac{2}{8}D\times1$
Disparity regression		Upsampling Regression	$H\times W\times D$ $H\times W$

该特征提取模块大大地简化了原有的特征提取模块，其中 H 和 W 分别代表输入图像的高和宽。特征提取中出现的不同特征图维度 32，64，128 及最后的输出仍为输入的 1/4，通道数为 32，这和 PSMNet 特征提取中出现的维度一样，仍然保留原先大致的网络架构。该特征提取模块只删除大量重复的网络层，将原本全为 3×3 的卷积核改为个别为 3×3、其余全为 1×1 的卷积核，模型整体参数量相比 PSMNet 减少 58%，提升了运算速度，同时该模型还能保证提取到需要的细节信息。

2.2 注意力机制

在 PSMNet 中间加入特征金字塔（SPP）模块[13]，通过不同尺度的卷积核来增大感受野，以聚合多个尺度的上下文信息。但金字塔池化操作会丢失大量的细节信息，同时采用较大的感受野会造成某个特征点周围相近的特征在其对应图像区域变得相似，特征点本身的信息也不突出了。因此本文在特征提取模块中加入注意力机制，以聚合上下文关系。注意力机制起源于机器翻译，现如今在卷积神经网络领域已成为一个重要概念，它可以在卷积神经网络中达到简单又有效的部署，没有增加很多参数，可以有效增强网络结构的表达力，使得网络可以更加关注学习特征区域，减少对一些不重要特征的关注。受文献[14]的启发，本文将通道注意力模块和空间注意力模块相结合，并将其集成到 CNN 中进行端到端的训练。和文献[15]的并联结构相比，本文所选结构更为简洁，且先经过通道注意力在空间维度进行压缩再经过空间注意力相比直接将特征图分别输入两个模块，消耗更少。具体结构如图 3 所示。

2.3 通道注意力模块

如图 3 上方的虚线框所示。假设输入特征图 F 的大小为 $H\times W\times C$，先分别经过 MaxPool 和 AvgPool 得到两个 $1\times1\times C$ 的通道描述，平均池化和最大池化可用来聚合特征映射的空间信息，接着把它们输入到一个有两层权重共享的神经网络，第一层神经元的个数为 C/r，其中 r 为缩减率，激活函数为 ReLU，第二层神经元的个数为 C，然后将得到

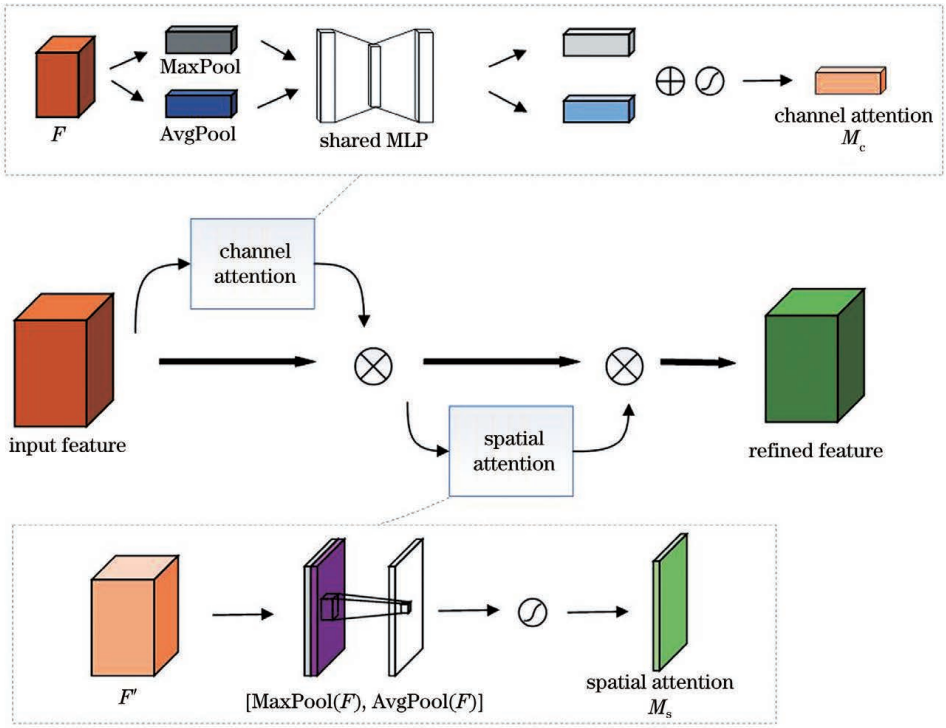

图 3　注意力机制

Fig. 3　Attention mechanism

的两个特征相加再用一个激活函数对其进行处理，得到通道注意力的权重系数M_c，最后将权重系数和输入特征图 F 相乘即可得到在通道注意力模块的输出特征图。通道注意力机制可表示为

$$M_c(F)=\sigma\{\mathrm{MLP}[\mathrm{AvgPool}(F)]+\mathrm{MLP}[\mathrm{MaxPool}(F)]\}, \tag{1}$$

式中：MLP 为多层感知机模型。

2.4　空间注意力模块

如图 3 下方的虚线框所示。空间注意力模块的输入就是通道注意力模块的输出，假定输入特征 F' 大小为 $H\times W\times C$，分别进行 MaxPool 和 AvgPool 得到两个 $H\times W\times 1$ 的通道描述，将两个通道描述拼接在一起，其经过一个 7×7 的卷积层被降维为一个通道，再经激活函数的处理，得到空间注意力的权重系数 M_s。最后将输入和 M_s 相乘，即可得到在空间注意力模块的输出：

$$M_s(F')=\sigma\{f^{7\times 7}\{[\mathrm{AvgPool}(F');\mathrm{MaxPool}(F')]\}\}, \tag{2}$$

式中：σ 为 sigmoid 激活函数。

由上所述，总的注意力机制处理流程可表示为

$$F''=M_s[M_c(F)\otimes F]\otimes[M_c(F)\otimes F], \tag{3}$$

式中：F''为经过所提注意力模块之后的特征图；⊗表示对应元素相乘操作。

2.5　改进的 3DCNN

在模型中引入 3DCNN，相比于其他使用距离度量的函数，该方法可以明显提高网络的性能。但是 3D 卷积复杂度高，参数量和运算量显著增加。因此本文对 3D 卷积的输入进行改进，使其计算量显著下降，得到图 4 所示的代价计算结构。

匹配成本特征体是通过将左图像特征和右图像特征沿着特征通道连接起来得到的，如图 4 所示，图中叠加的矩形就表示特征的连接。在最大视差 D 内，将偏移的左右图像特征在特征通道维度进行串联，同时为了整合视差维度的信息，将视差维度也进行串联，进而构建出一个 4 维的匹配成本特征体。左右图像特征产生微小($d=1$)的偏移时，如图 4 上方虚线框所示，显然会存在冗余信息，从而导致在后续 3D 卷积中耗费大量资源。若左右图像特征产生相对较大($d>1$)的偏移，如图 4 下方虚线框所示，

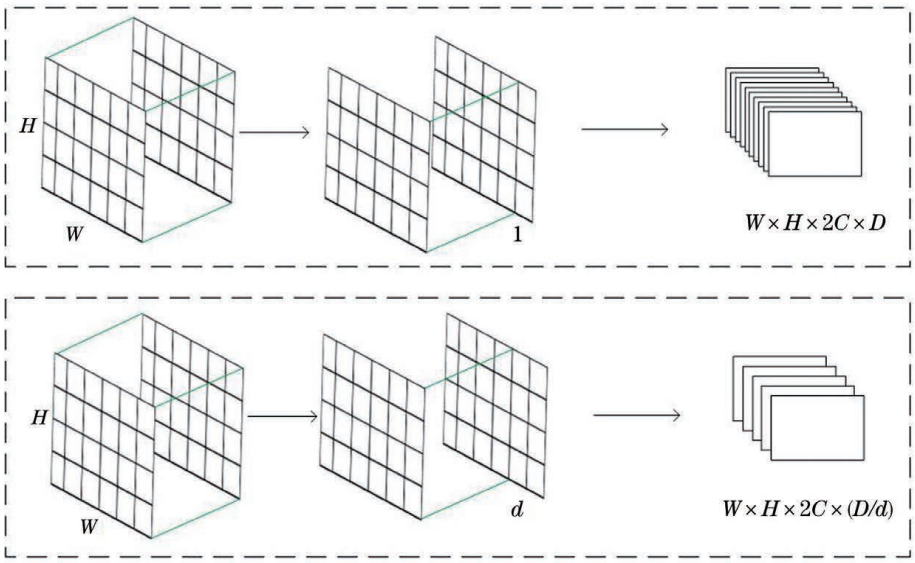

图 4　代价计算结构

Fig. 4　Cost calculation structure

则匹配成本特征体的视差维度变为原来的 $1/d$，后续 3D 卷积模块中的显存消耗和数据量均能降低为原来的 $1/d$。

但是，偏移步长的增加会在一定程度上导致模型在视差维度上的细化能力变弱。为了削弱这种影响，在每个偏移 d 内，对匹配成本特征体进行多视差预测，其数量记为 q。如偏移步长取 d，那么代价计算后的视差维度就为 D/d，即视差维度上的采样数为 D/d，多视差预测就是在每两个采样点中间再加入 $q-1$ 个采样点，使得模型可以在后续处理过程中学习到视差概率分布的细化函数，这对视差预测值有一定的改善。在 3D 卷积的最后一层加入此操作对参数量和消耗的影响极小。

3　视差回归计算

本文采用完全可微的 Soft-argmin 函数来得到属于每个像素点的最优视差值，从而得到相对平滑的视差图。对匹配成本特征体进行 3DCNN 和上采样处理后，得到每个像素在所规定的视差维度内的成本，成本越大则代表特征体越不匹配。通过 Softmax 操作得到每个像素属于每个视差的概率，然后将每个视差和其对应的概率值进行加权求和，得到每个像素点的预测视差值。由于本文在匹配成本特征体的视差维度上进行了改进，所以 K 值在此处并不是传统意义上的最大视差 D，而是 $D\times q/d$，则预测视差值可表示为

$$\hat{d}_t=\sum_{k=0}^{K}d_t\times\sigma(-c_k),\tag{4}$$

式中：$\hat{d}_t$ 为预测视差值；c_k 为成本值；$\sigma(\cdot)$ 为 Softmax 操作；$d_t=k\times D/K$，其中 K 为视差维度，D 为最大视差，即 192。

4　损失函数优化

本文采用平滑的 L1 损失函数为基础函数来训练网络，它具有更强的鲁棒性且对离群值不敏感，且 L1 损失函数常被用于边界框回归问题求解[16]。加入交叉熵损失函数 L_{CE}，它表示真实概率分布和预测概率分布之间的差异，L_{CE} 越小代表模型预测效果越好。但是在 SceneFlow 数据集中，部分像素点的真实视差值超过所规定的最大视差范围，在计算损失时要将这部分像素点剔除。L_{CE} 可表示为

$$L_{CE}=\frac{1}{N}\sum_{i=0}^{N}\sum_{k=0}^{K}\{-\exp(-|d_t-d_{gt}|/b)\times\ln[\sigma(-c_k)]\},\tag{5}$$

$$L(d_t,\hat{d}_t)=\frac{1}{N}\sum_{i=1}^{N}\mathrm{smooth}_{L1}(d_{gt}-\hat{d}_t),\tag{6}$$

其中，

$$\mathrm{smooth}_{L1}(x)=\begin{cases}0.5x^2, & |x|<1\\ |x|-0.5, & \text{otherwise}\end{cases},\tag{7}$$

式中：N 表示所有像素点的数量；b 为散度，实验中取 $b=1$；d_{gt} 表示每个像素的真实视差值。

综上所述，所提出的网络结构的损失函数为

$$L=\alpha L_1+L_{CE},\tag{8}$$

式中：α 为 L_1 损失函数的权重系数，实验中取 0.2。

5　实　　验

在 SceneFlow 数据集、KITTI2015 数据集[17]和

KITTI2012 数据集上进行训练和测试实验。首先介绍实验所用到的三个数据集，其次介绍了实现细节及实验指标，最后对不同情况下的实验结果进行对比分析。

5.1 数据集介绍

SceneFlow 数据集是一个大型的合成数据集，包含 35454 对训练图片和 4370 对测试图片，照片分辨率为 960×540，每对图片都提供了稠密的真实视差图。SceneFlow 数据集包含三个子集，即 FlyingThing3D、Monkaa 和 Driving。FlyingThing3D 主要包括沿随机 3D 轨迹飞行的日常物体；Monkaa 包含非刚性和柔和的关节运动，及形态各异的动物皮毛；Driving 包含驾驶中的街景。

KITTI2015 数据集是从驾驶视角采集到的道路实景数据，包含 200 对训练数据和 200 对测试数据，其中训练数据是采集到的稀疏视差图，分辨率为 1240×376。KITTI2012 数据集也是通过移动车辆采集到的实景数据，包含 194 对训练图片和 195 对测试图片。

5.2 实验细节及指标

在 Ubuntu16.04 下，采用 Pytorch 框架，网络训练和测试的设备为 NVIDIA GeForce GTX 1080Ti。使用 Adam 优化器[18]，延迟率参数 $\beta_1=0.9$，$\beta_2=0.999$。对于所有训练数据，图像大小设置为 512×256。为了提高模型的泛化能力，对训练数据进行颜色增强和空间变换增强。先在 SceneFlow 上以 0.001 的学习率训练 30 轮，然后以 0.0001 的学习率训练 25 轮。接着采用迁移学习的方法，将在 SceneFlow 上训练好的模型在 KITTI 上进行微调。

实验评价指标为终点误差(epe)和 3 像素误差，epe 为每个像素点的真实视差值和预测视差值之间的平均欧氏距离。3 像素误差(3px)为真实视差值和预测视差值之间误差大于 3 的像素点占图中所有像素点的比例。

5.3 消融实验

在本节中，首先对优化后的 RESNet 模块进行测试，接着对提出的注意力模块进行测试，最后对本文方法进行对比。同时选取本文的基准网络 PSMNet 进行对比。epe 指标为 SceneFlow 测试集上的结果。表 2 中 d、q 分别代表代价计算中较大偏移和特征体输出时的多视差预测，“√”表示模型包含这一模块，实验结果如表 2 所示。

表 2 不同网络结构的对比

Table 2 Comparison of different network structures

Network	Optional module			
	RESNet simplified	Attention mechanism	d, q	epe /pixel
PSMNet				1.09
Ours	√			1.13
	√	√		0.98
	√	√	√	0.83

由表 2 的实验结果可知，基准 epe 为 1.09，经过本文处理后的 epe 为 0.83，幅度减小约 24%。这充分证明本文所提方法的可行性。从表 2 第 2 行可得，在仅加入 RESNet 优化模块时，在参数量和消耗大幅度减少情况下依然可以取得较低的指标。从表 2 第 3 行可得，加入注意力机制能够更好地汇聚上下文信息，使指标降低约 10%。

5.4 与其他经典算法对比

首先对本文和其他经典算法如 MC-CNN、GC-Net、DispNet、CRL[19]在 SceneFlow 测试集上的 epe 指标进行对比，接着在 KITTI2015 数据集上对 3px 进行对比，结果如表 3、表 4 所示。

表 3 在 SceneFlow 测试集上的效果对比

Table 3 Comparison of effects on SceneFlow test set

Network	epe /pixel	Number of parameters /10^6
PSMNet	1.09	5.20
MC-CNN	3.79	–
GC-Net	2.51	3.50
DispNet	1.68	42.00
CRL	1.32	78.00
Ours	0.83	2.20

表 4 KITTI2015 数据集的比较

Table 4 Comparison on KITTI2015 dataset

Network	3px /%	Running time /s
PSMNet	2.32	0.41
MC-CNN	3.89	67.00
GC-Net	2.87	0.90
DispNet	4.34	0.06
CRL	2.67	0.47
Ours	2.09	0.26

表 3 所示为一些基于深度学习的端到端网络模型的指标。可以看出：所提算法在参数量最少、消耗

最小的情况下，仍能取得很高的精度；与基准网络 PSMNet 相比，所提算法的参数量减少约 58%，由前文可知，epe 表示两个视差值之间的平均欧氏距离，数值越小代表输出越准确，精度越高，因此精度提升约 24%；与 GC-Net 相比，所提算法的参数量减少约 37%，精度提升约 67%；与 DispNet 和 CRL 相比，所提算法的参数量更是大幅度减少，消耗占比大量降低，这从侧面反映出本文所提模块的高效性。表 4 是本文方法与 PSMNet、MC-CNN、GC-Net、DispNet 等网络在 KITTI2015 数据集上进行对比的结果，running time 是 KITTI2015 测试集的一对图片的运行时间。从表 4 中数据可知，本文所提算法比基准算法的速度更快，效果更好。

图 5 是本文所提方法和 PSMNet 在 KITTI2015 数据集上的可视化结果，图中方框位置为改进的位置，可以看出 PSMNet 在路标、红绿灯、电线杆等位置表现较模糊，而本文所提方法可以得到清晰的轮廓。

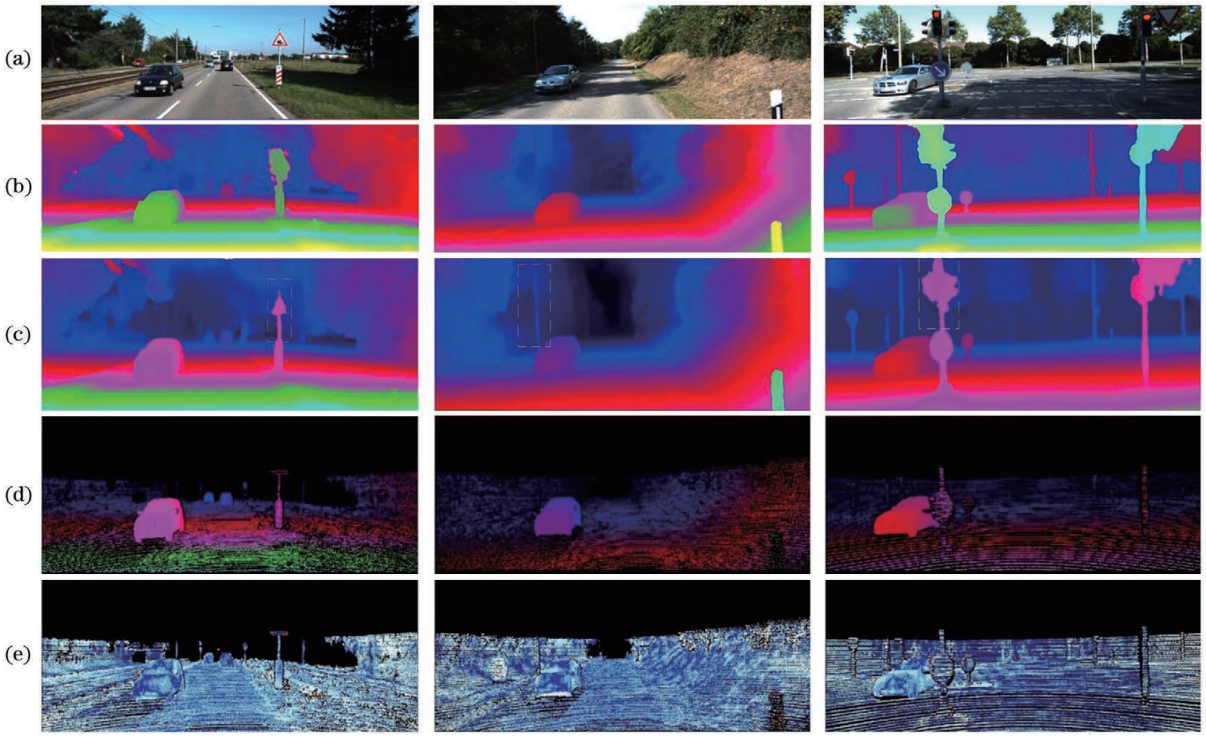

图 5　KITTI2015 数据集上的可视化结果。(a)左视角图像；(b)PSMNet 预测视差图；(c)本文预测视差图；(d)真实视差图；(e)误差图

Fig. 5　Visualization results on KITTI2015 data set. (a) Left view images; (b) PSMNet predicted disparity maps; (c) predicted disparity maps of this article; (d) true disparity maps; (e) error maps

5.5　不同超参数对比

在 SceneFlow 测试集上对超参数 d、q 进行分析。不同的 d 和 q 会影响匹配成本特征体的视差维度，间接地对匹配效果产生影响。在代价计算中，d 值越大，计算速度越快，3DCNN 的参数量大幅度减少，整体效率变高。采用很大的 d 值，采样间隔变大，导致采样信息不均匀，使得效率变高的同时精度直线降低，因此需选用合适的 d 值。q 值越大，模型的细化能力就越强，同时计算量也在增加；q 值越小，模型的精度就会受到影响。因此选择超参数进行对比实验，其中，Time 为 SceneFlow 测试集上 20 张图片所需运行时间的平均值。GPU 为训练过程中所占用的内存。所得结果如表 5 所示。

表 5　超参数在 SF-test 上的对比

Table 5　Comparison of hyperparameters on SF-test

d	q	epe /pixel	Time /s	GPU /GB
1	1	0.81	0.88	14.00
2	2	0.83	0.76	11.80
3	3	0.89	0.62	9.80
4	4	0.96	0.49	8.90

由表 5 可知，两个超参数都为 1 时得到的效果最好，但此时的消耗最大。综合观察表中数据可得，当两个超参数都为 2 时能得到较好的结果，此时的 epe 指标仅下降 2.5%，而时间缩短了约 14%，消耗占比也合适。与两个超参数都为 4 相比，两个超参

数都为 2 时虽然消耗多，但 epe 提高了约 16%。综上所述，选择超参数都为 2 时的结果为最终结果。

6 结 论

提出了一种优化的基于深度学习的立体匹配方法，通过分析传统方法对基于深度学习的立体匹配方法加以改进。对基于深度学习的立体匹配方法的特征提取模块的冗余部分进行删减，这可在保留其框架的同时减少了参数量，同时在特征提取中加入注意力机制，避免因使用 SPP 造成细节信息丢失。通过对特征建立联系来聚合上下文信息。对 3DCNN 的输入和输出在视差维度上进行改进，使其效率更高。所提算法在占用资源更少的情况下，获得了更佳的性能。进一步的工作将结合其他视觉任务，以实现多任务融合网络在边缘计算平台的应用。

参 考 文 献

[1] Konolige K. Small vision systems: hardware and implementation[M]// Robotics research. London: Springer, 1998: 203-212.

[2] Hirschmuller H. Stereo processing by semiglobal matching and mutual information[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(2): 328-341.

[3] Xiao H, Tian C, Zhang Y, et al. Stereo matching algorithm based on improved Census transform and gradient fusion[J]. Laser & Optoelectronics Progress, 2021, 58(2): 0215008.
萧红, 田川, 张毅, 等. 基于改进 Census 变换与梯度融合的立体匹配算法[J]. 激光与光电子学进展, 2021, 58(2): 0215008.

[4] Li D H, Shen H Y, Yu X, et al. Binocular ranging method using stereo matching based on improved census transform[J]. Laser & Optoelectronics Progress, 2019, 56(11): 111503.
李大华, 沈洪宇, 于晓, 等. 一种改进 Census 变换的双目匹配测距方法[J]. 激光与光电子学进展, 2019, 56(11): 111503.

[5] Žbontar J, LeCun Y. Computing the stereo matching cost with a convolutional neural network[C]//2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 7-12, 2015, Boston, MA, USA. New York: IEEE Press, 2015: 1592-1599.

[6] Luo W J, Schwing A G, Urtasun R. Efficient deep learning for stereo matching[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 5695-5703.

[7] 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 (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 4040-4048.

[8] Kendall A, Martirosyan H, Dasgupta S, et al. End-to-end learning of geometry and context for deep stereo regression[C]// 2017 IEEE International Conference on Computer Vision (ICCV), October 22-29, 2017, Venice, Italy. New York: IEEE Press, 2017: 66-75.

[9] Chang J R, Chen Y S. Pyramid stereo matching network[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 5410-5418.

[10] Zhang F H, Prisacariu V, Yang R G, et al. GA-net: Guided aggregation net for end-to-end stereo matching[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 15-20, 2019, Long Beach, CA, USA. New York: IEEE Press, 2019: 185-194.

[11] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 770-778.

[12] Khamis S, Fanello S, Rhemann C, et al. StereoNet: guided hierarchical refinement for real-time edge-aware depth prediction[M]// Ferrari V, Hebert M, Sminchisescu C, et al. Computer Vision-ECCV 2018. Lecture notes in computer science. Cham: Springer, 2018, 11219: 596-613.

[13] He K M, Zhang X Y, Ren S Q, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(9): 1904-1916.

[14] Woo S, Park J, Lee J Y, et al. CBAM: convolutional block attention module[M]//Ferrari V, Hebert M, Sminchisescu C, et al. Computer vision-ECCV 2018. Lecture notes in computer science. Cham: Springer, 2018, 11211: 3-19.

[15] Cheng M Y, Gai S Y, Da F P. A stereo-matching neural network based on attention mechanism[J].

Acta Optica Sinica, 2020, 40(14): 1415001
程鸣洋, 盖绍彦, 达飞鹏. 基于注意力机制的立体匹配网络研究[J]. 光学学报, 2020, 40(14): 1415001.

[16] Ren S Q, He K M, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[17] Menze M, Geiger A. Object scene flow for autonomous vehicles[C]//2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 7-12, 2015, Boston, MA, USA. New York: IEEE Press, 2015: 3061-3070.

[18] Kingma D P, Ba J. Adam: a method for stochastic optimization[EB/OL]. (2014-12-22)[2020-12-10]. https://arxiv.org/abs/1412.6980.

[19] Pang J H, Sun W X, Ren J S, et al. Cascade residual learning: a two-stage convolutional neural network for stereo matching[C]//2017 IEEE International Conference on Computer Vision Workshops (ICCVW), October 22-29, 2017, Venice, Italy. New York: IEEE Press, 2017: 878-886.