第 58 卷 第 4 期/2021 年 2 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

基于多任务学习的立体匹配算法

王玉锋[1,2]，王宏伟[2,3**]，刘宇[2]，杨明权[2]，全吉成[1,2*]

[1]海军航空大学，山东 烟台 264001；

[2]空军航空大学，吉林 长春 130022；

[3]信息工程大学，河南 郑州 450001

摘要 引入辅助任务信息有助于立体匹配模型理解相关知识，但也会增加模型训练的复杂度。为解决模型训练对额外标签数据的依赖问题，提出了一种利用双目图像的自相关性进行多任务学习的立体匹配算法。该算法在多层级渐进细化过程中引入了边缘和特征一致性信息，并采用循环迭代的方式更新视差图。根据双目图像中视差的局部平滑性和左右特征一致性构建了损失函数，在不依赖额外标签数据的情况下就可以引导模型学习边缘和特征一致性信息。提出了一种尺度注意的空间金字塔池化，使模型能够根据局部图像特征来确定不同区域中不同尺度特征的重要性。实验结果表明：辅助任务的引入提高了视差图精度，为视差图的可信区域提供了重要依据，在无监督学习中可用于确定单视角可见区域；在 KITTI2015 测试集上，所提算法的精度和运行效率均具有一定的竞争力。

关键词 机器视觉；立体匹配；深度学习；多任务学习；双目视觉

中图分类号 TP391.4 **文献标志码** A **doi**：10.3788/LOP202158.0415010

Algorithm for Stereo Matching Based on Multi-Task Learning

Wang Yufeng[1,2], Wang Hongwei[2,3**], Liu Yu[2], Yang Mingquan[2], Quan Jicheng[1,2*]

[1]*University of Naval Aviation, Yantai, Shandong* 264001, *China*;

[2]*Aviation University of Air Force, Changchun, Jilin* 130022, *China*;

[3]*Information Engineering University, Zhengzhou, Henan* 450001, *China*

Abstract The introduction of auxiliary task information is helpful for the stereo matching model to understand the related knowledge, but the complexity of model training increases. In order to solve the problem of dependence on extra label data during model training, we proposed an algorithm based on multi-task learning for stereo matching by using the autocorrelation of binocular images. This algorithm introduces the edge and feature consistency information in the multi-level progressive refinement process and updates the disparity map in a cyclic and iterative manner. According to the local smoothness of disparity and the consistency of left and right features of binocular images, a loss function is constructed to guide the model to learn the edge and feature consistency information without relying on additional label data. A spatial pyramid pooling with scale attention is proposed to enable the model to determine the importance of different scale features based on the local image features in different areas. The experimental results show that the introduction of auxiliary tasks not only improves the accuracy of disparity maps, but also provides a significant basis for the trusted regions of disparity maps. It can also be used to determine the single-view visible areas in unsupervised learning. The proposed algorithm has certain competitiveness in terms of accuracy and operating efficiency on the KITTI2015 test dataset.

Key words machine vision; stereo matching; deep learning; multi-task learning; binocular vision

OCIS codes 150.6910; 150.5670; 150.1135

收稿日期：2020-04-16；**修回日期**：2020-04-23；**录用日期**：2020-04-25

* **E-mail**：jicheng_quan@126.com ** **E-mail**：alex19820911@126.com

1 引 言

密集视差图的预测是计算机视觉领域中的一个基本问题，被广泛应用于机器人、三维重构和自动驾驶等领域。近年来，随着深度学习的快速发展，立体匹配作为一个学习任务，利用卷积神经网络（CNN）对部分或全部立体匹配过程进行建模。最初，基于CNN的立体匹配算法主要聚焦于学习图像的深度特征表示。Žbontar 等[1-2]训练了一个共享权重的特征提取网络，该网络为图像窗口提供了稳健的特征表示，并通过全连接层来计算图像特征之间的相似度。以该网络为基础，算法特征稳健性得到改善，运行效率和预测置信度等方面得到了进一步提高[3-6]。与传统的立体匹配算法相比，这些方法取得了一定的精度增益，但算法的计算量和内存负载较大，仍需要传统的后处理步骤进一步修正异常值。

利用CNN对立体匹配的整个过程进行建模，并对整个模型进行联合优化，往往能得到更高的性能。为了打破人工设计函数的局限，Mayer 等[7]设计了一个端到端的立体匹配模型，直接以双目图像为输入来预测密集视差图，并制作了一个大型的合成数据集对模型进行有监督训练。为了改善算法精度，Pang 等[8]级联了一个预测视差图残差的子模块，Liang 等[9]结合贝叶斯推理充分利用了先验和后验的图像特征，Jie 等[10]以循环迭代的方式来逐渐更新视差图。为了实现对立体匹配过程的更好建模，Kendall 等[11]使用3DCNN来处理特征比对过程，通过在三个维度上理解场景信息，能更好地处理病态区域（如被遮挡和无/弱纹理区域）。这种直观的建模机制加快了模型收敛速度，表现出较大的潜力，在多个方面得到了进一步研究，如融合多尺度特征[12-13]、改进损失体构建模式[14]、压缩三维卷积来提高算法运行效率[15-17]及学习局部相似性关系[18]等。

根据场景中物体的边界和语义知识，人眼能很好地对齐双目图像，从而形成精细的立体感。因此，综合边缘和语义信息有助于立体匹配模型理解几何知识，尤其是无纹理区域和单视角可见区域。Yang 等[19]在立体匹配模型中嵌入了语义特征，并根据语义线索构建了损失项以改善视差图的局部细节。Song 等[20-21]通过边缘特征的嵌入来改善视差图，并采用边缘抑制的视差平滑性损失来改善视差图的局部平滑性。语义和边缘信息的引入使模型预测的视差图含有更好的局部细节，但辅助任务模块的训练需要额外的标签数据，增加了模型训练的难度和对额外标签数据的依赖性。

为了充分发挥多任务学习的优势，同时解决模型对额外标签数据的依赖问题，本文在先前工作[22]的基础上引入了辅助任务来改善算法精度，并采用双目图像的自相关性来引导模型学习辅助任务信息。本文算法的主要思路包括三个方面：1）在多层级渐进细化过程中引入边缘和特征一致性信息，使模型在视差细化中能够包含更明确的边缘和遮挡信息；2）根据双目图像的自相关性构建损失函数，在不依赖额外标签数据的情况下引导模型学习边缘和特征一致性信息；3）采用尺度注意的特征融合，使模型能够根据局部图像特征来确定不同尺度特征的重要性。

2 所提算法

与文献[22]相似，本文所提算法首先在较低的空间分辨率层级预测初始视差图，然后在多个分辨率层级通过特征的对比来恢复视差图细节，包括三个主要模块：特征提取模块（FEM）、视差初始化模块（DIM）和视差细化模块（DRM），整体结构如图1所示。其中，FEM共包含5组卷积，在6个分辨率

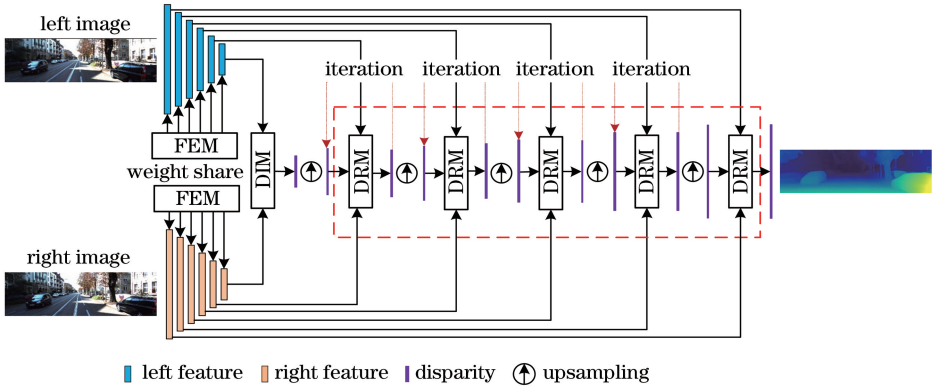

图1 所提算法的整体结构

Fig. 1 Overall architecture of proposed algorithm

层级为 DIM 和 DRM 提供稳健的特征表示；在原始图像 1/32 的分辨率层级，DIM 使用 3D CNN 预测初始视差图；在其他 5 个分辨率层级，DRM 通过对比左右视角的图像特征，使用 CNN 同步学习视差图残差、边缘图和特征一致性图，并以循环迭代的方式逐渐改善视差图。

模型以双目图像为输入，同步预测视差图、边缘图和特征一致性图的基本流程如下。

1）以双目图像为输入，使用权重共享的 FEM 在多个分辨率层级进行特征表示的学习，并通过邻域尺度特征的反向融合来增强特征的稳健性，输出 6 个空间尺度上的图像特征。

2）在原始图像 1/32 的空间分辨率层级，通过聚合左图像特征和平移的右图像特征来构建三维特征体，并使用 3D CNN 学习计算匹配代价，再使用 softmax 函数将其转化为视差概率分布，并通过可差分的视差回归函数输出粗略的视差图，上采样后作为 DRM 模块的初始视差图。

3）以左右图像特征、初始视差图、全 0 值的边缘图和全 1 值的特征一致性图为输入，使用 DRM 模块同步预测视差图残差、边缘图和特征一致性图，将初始视差图和视差图残差相加来修正初始视差图，并将修正的初始视差图、边缘图和特征一致性图作为下一次迭代的初始值。

4）重复步骤 3）直至达到设定的迭代次数，对输出的视差图、边缘图和特征一致性图进行上采样并将采样结果作为下一层级 DRM 的初始值。

5）采用步骤 3）和步骤 4）的方法调整视差图，直到预测的视差图的尺寸与原始图像的空间分辨率相同。

与文献[22]相比，所提算法采用相同的 DIM，主要对 FEM 和 DRM 进行了改进，以下对改进的 FEM 和 DRM 进行详细的介绍。

2.1　尺度注意的特征提取模块

通常不同区域的纹理丰富程度具有较大差异，在纹理丰富的区域，局部纹理就具有较好的区分度，而对于无纹理区域或简单重复纹理区域，需要更宽的视场范围进行特征描述。为了使模型能够提取更加稳健的特征表示，以空间金字塔池化（spatial pyramid pooling，SPP）[23]为基础，提出一种尺度注意的空间金字塔池化（spatial pyramid pooling with scale attention，SPPSA），通过学习不同区域中不同尺度的权重因子来控制相应尺度特征的重要性程度，从而实现不同区域对不同尺度上的特征具有不同的关注度。SPPSA 与 SPP 的结构对比如图 2 所示，其中，conv(3，1)表示核大小为 3、步长为 1 的卷积层，avgpool(4)表示核大小和步长均为 4 的平均池化层，$\boldsymbol{w}^s$ 和 $\boldsymbol{F}^s$ 分别为不同尺度的权重和特征，尺度 $s=1,2,3,4,5$。

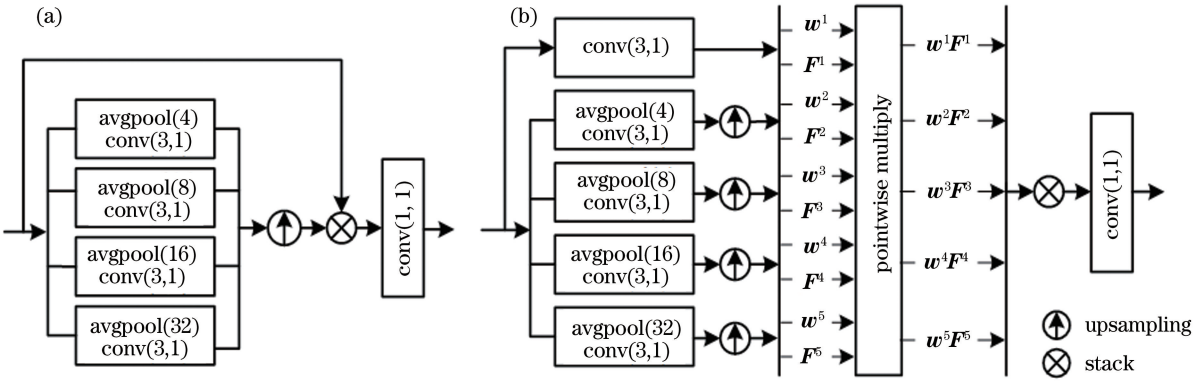

图 2　SPP 和 SPPSA 的结构对比。(a) SPP；(b) SPPSA

Fig. 2　Architecture comparison of SPP and SPPSA. (a) SPP; (b) SPPSA

结合 SPPSA 的 FEM 采用 5 组卷积来降低空间分辨率，分别记为 FEM-l_1（卷积组编号 $l_1=1,2,\cdots,5$），其均包含一个核大小为 3、步长为 2 的卷积层和一个基本残差块[24]，并在 FEM-2 中使用 SPPSA 模块代替基本残差块，其整体结构如图 3 所示。其中，$\boldsymbol{F}_l$（$l=0,1,2,3,4,5$）为不同分辨率层级 l 的图像特征，每个卷积层都跟随一个正则化层[25]和 Leaky ReLU 激活函数[26]（负值斜率取 0.1）。与文献[22]相似，对邻域尺度的特征进行反向融合（如图 3 中实线框所示），先将低分辨率的图像特征进行线性上采样，再将两个尺度的特征聚合并使用一个卷积层进行融合。

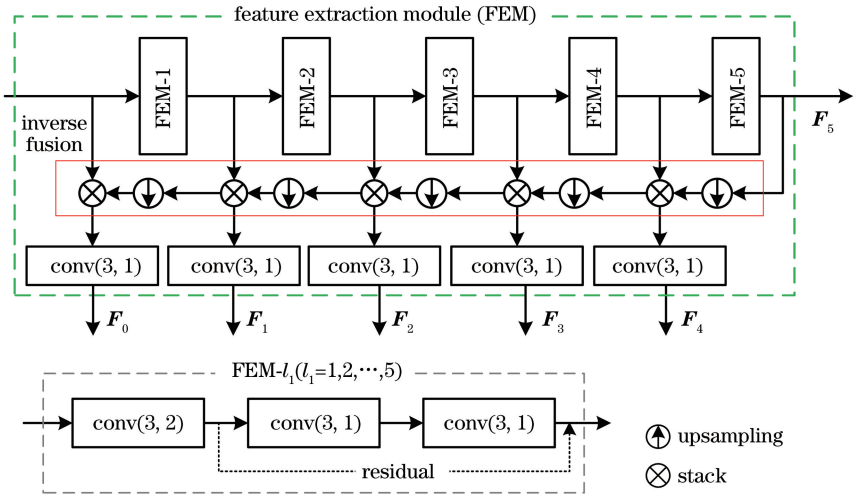

图 3　FEM 的结构

Fig. 3　Architecture of FEM

2.2　多任务学习的视差细化模块

在渐进的视差图细化过程中，图像边缘和单视角可视区域是非常重要的线索。通常，视差不连续边界往往含有明显的边缘信息，可以通过图像边缘来确定视差图的平滑性区域。视差平滑性可以引导模型预测局部平滑的视差图，从而改善无纹理区域的视差图精度，边缘信息也有助于模型根据局部特征关系来恢复视差图的局部结构细节。由于单视角可视区域不具有左右图像特征的一致性，无法通过对比左右图像特征来修正视差图，只能通过局部图像纹理信息来推理视差图。相应区域的检测可以使模型能够更好地区分具有不同特点的区域并进行分情况处理，进而有助于模型更好地识别需要学习的多种特征模式。

通过在 DRM 中引入辅助的图像边缘检测和特征一致性检测任务，可以使模型学习更精细的边缘信息和对象之间的遮挡关系信息，从而引导模型学习视差图的精细结构并识别单视角可见区域，且通过共享基本图像特征可以引导模型学习到更具泛化能力的特征表示。因此，图 1 中 DRM 的结构如图 4 所示，其中 MBF(multi-branch fusion)模块的结构请参见文献[22]。由于图 1 中的 DIM 并不预测边缘图和特征一致性图，因此在与 DIM 连接的 DRM 的初始边缘图中填充 0 及在初始特征一致性图中填充 1。

在 DRM 的结构中，边缘信息的引入包含两个部分，一是仅以左图像特征为输入学习图像的纹理边缘(图 4 中的 edge 1)，二是以左图像特征及其重构和边缘特征为输入学习视差不连续边缘(图 4 中的 edge 2)。引入的特征一致性信息与视差不连续边缘共享特征，分别使用两个卷积层作为相应任务的输出层。

2.3　损失函数的改进

通常辅助任务的学习需要额外的标签任务，或是在相关任务中训练模型权重，这无疑增加了模型训练的难度。在模型训练阶段，所提算法根据特征图的左右一致性和视差图的局部平滑性来构建惩罚项，从而引导模型预测合理的边缘图和特征一致性图，可以在不需要任何辅助任务标签的情况下训练模型。

首先对模型预测的中间层输出进行说明。模型提取的多尺度特征共 6 组，分别记为图 3 中的 $\boldsymbol{F}_l(l=0,1,2,3,4,5)$，每组均包含左右图像特征，在每个尺度上均有预测的视差图，预测的视差图记为 $\hat{\boldsymbol{d}}_l(l=0,1,\cdots,5)$。除视差初始化层级之外，每个尺度层级也会预测图像纹理边缘、视差不连续边缘和特征一致性图，分别记为 $\boldsymbol{B}_{l_2}^{(1)}(l_2=0,1,\cdots,4)$、$\boldsymbol{B}_{l_2}^{(2)}(l_2=0,1,\cdots,4)$ 和 $\boldsymbol{C}_l(l=0,1,\cdots,5)$。对于任意分辨率层级 l，使用右图像特征 $\boldsymbol{F}_l^{\text{right}}$ 和预测视差图 $\hat{\boldsymbol{d}}_l$ 构建的左图像特征记为 $\boldsymbol{F}_l^{\text{wraped}}$，若预测的视差图准确，在共同可视区域内左图像特征及其重构应满足一致性，两者的差值可表示其不一致性，记为 $\Delta\boldsymbol{F}_l=\left|\boldsymbol{F}_l^{\text{left}}-\boldsymbol{F}_l^{\text{wraped}}\right|$。预测的特征一致性图 $\boldsymbol{C}_l$ 是对共同可视区域的标识，因此可定义特征一致性损失 $L_{\boldsymbol{F}}$ 为

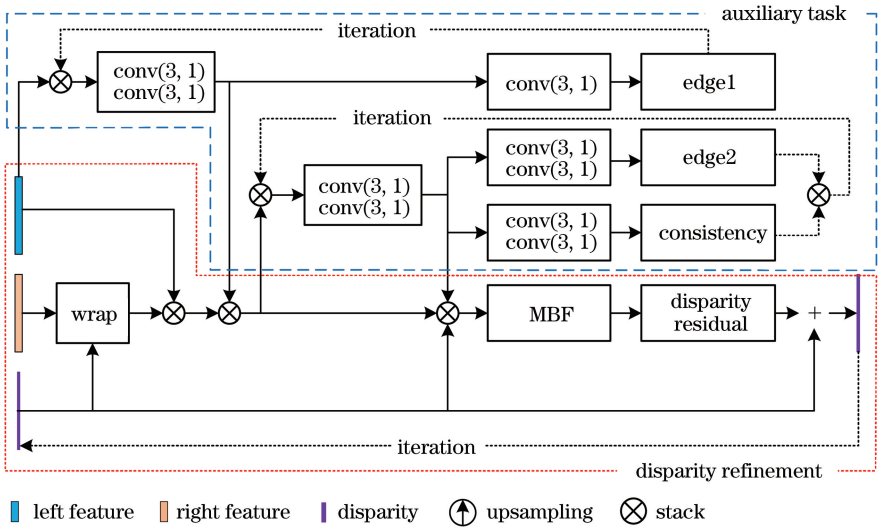

图 4　DRM 的结构

Fig. 4　Architecture of DRM

$$L_{\boldsymbol{F}}=\sum_{l=5}^{0}\mathrm{mean}(\Delta\boldsymbol{F}_l\cdot\boldsymbol{C}_l)+\mathrm{mean}(1-\boldsymbol{C}_l)\mathrm{mean}(\Delta\boldsymbol{F}_l),\tag{1}$$

式中：mean(·)表示对矩阵求均值。若预测的视差图准确，当共同可视区域(即 $\Delta\boldsymbol{F}_l$ 中值较小的区域) $\boldsymbol{C}_l$ 具有较大的值，单视角可见区域(即 $\Delta\boldsymbol{F}_l$ 中值较大的区域)$\boldsymbol{C}_l$ 具有较小的值时，$L_{\boldsymbol{F}}$ 值则较小，因此可以通过优化模型来使 $L_{\boldsymbol{F}}$ 减小，从而引导模型学习预测 $\boldsymbol{C}_l$。同时，若模型预测的 $\boldsymbol{C}_l$ 准确，则视差图准确，共同视域内 $L_{\boldsymbol{F}}$ 的值也就较小，因此通过使 $L_{\boldsymbol{F}}$ 减小也能引导模型预测更准确的视差图。

视差图局部具有平滑性且不连续边缘通常具有明显的纹理边缘，预测的边缘图 $\boldsymbol{B}_{l_2}^{(1)}$ 和 $\boldsymbol{B}_{l_2}^{(2)}$ 是对图像纹理边缘和视差图不连续边缘的标识，两种边缘均表征了视差的不连续性。边缘图 $\boldsymbol{B}_{l_2}^{(1)}$ 仅以左图像特征为依据，很难排除局部平面内的纹理信息；边缘图 $\boldsymbol{B}_{l_2}^{(2)}$ 以左图像特征及其重构为输入，可以通过局部特征一致性来排除局部平面内的纹理信息，并有充足的信息可以预测视差的不连续边缘。尽管边缘图 $\boldsymbol{B}_{l_2}^{(1)}$ 中包含的边缘通常多于视差不连续边缘，但边缘占整幅图像的比例仍较小，仍然可用于无纹理边缘区域的平滑性约束。因此，将两种边缘进行综合处理，取 $\boldsymbol{B}_{l_2}=0.5(\boldsymbol{B}_{l_2}^{(1)}+\boldsymbol{B}_{l_2}^{(2)})$，定义边缘抑制的平滑性损失 L_{ds} 为

$$L_{\mathrm{ds}}=\sum_{l_2=4}^{0}\mathrm{mean}[\partial\boldsymbol{D}_{l_2}\cdot(1-\boldsymbol{B}_{l_2})]+\mathrm{mean}(\boldsymbol{B}_{l_2})\mathrm{mean}(\partial\boldsymbol{D}_{l_2}),\tag{2}$$

式中：$\partial\boldsymbol{D}_{l_2}$ 表示对视差图 $\boldsymbol{D}_{l_2}$ 沿某个方向求偏导数。当视差图的局部平滑区域(即 $\partial\boldsymbol{D}_{l_2}$ 中值较小的区域)$\boldsymbol{B}_{l_2}$ 具有较小的值，视差不连续边缘(即 $\partial\boldsymbol{D}_{l_2}$ 中值较大的区域)$\boldsymbol{B}_{l_2}$ 具有较大的值时，L_{ds} 值则较小，因此可以通过优化模型使 L_{ds} 减小，从而引导模型学习预测 $\boldsymbol{B}_{l_2}$。同时，若模型的预测 $\boldsymbol{B}_{l_2}$ 准确，则可以引导模型预测局部平滑的视差图来改善其精度。

所提算法既可以采用有监督训练方式训练模型，也可以采样无监督训练方式训练模型。模型设计预测了边缘图和特征一致性图，且不需要使用相应的标签数据，两种训练方式均需要通过引入特征一致性损失和视差平滑性损失来引导模型预测合理的边缘图和特征一致性图，同时，边缘图和特征一致性图也有助于模型预测更准确的视差图。

有监督的训练方式仅使用视差图标签数据(记为 $\boldsymbol{d}_{\mathrm{gt}}$)，使用平均池化生成 6 个分辨率层级的标签，分辨率每降一级则将视差值减半，各层级的视差图记为 $\boldsymbol{d}_{\mathrm{gt}}^{(l)}$($l=0,1,\cdots,5$)，且 $\boldsymbol{d}_{\mathrm{gt}}^{(0)}=\boldsymbol{d}_{\mathrm{gt}}$。按照中间层的输出顺序进行渐进式训练，只有当误差均值满足一定阈值(记为 T)后再训练后续的中间层。

对于有监督训练方式，最终的损失函数为

$$L_{\text{supervised}}=L_{\boldsymbol{d}}+w_{\boldsymbol{F}}L_{\boldsymbol{F}}+w_{\text{ds}}L_{\text{ds}},\tag{3}$$

式中：$w_{\boldsymbol{F}}$ 和 w_{ds} 为权衡特征一致性损失和局部平滑性损失的重要性权重，实验中分别取 1 和 0.1；$L_{\boldsymbol{d}}$ 为基于视差值的损失，计算公式为

$$L_{\boldsymbol{d}}=\sum_{l=5}^{0}F_{\text{smoothL1}}(\hat{\boldsymbol{d}}_l,\boldsymbol{d}_{\text{gt}}^{(l)}),\tag{4}$$

式中：$F_{\text{smoothL1}}(\cdot)$ 为平滑的 L1 损失函数[27]，且当满足 $F_{\text{smoothL1}}(\hat{\boldsymbol{d}}_l,\boldsymbol{d}_{\text{gt}}^{(l)})>T$ 时 $L_{\boldsymbol{d}}$ 停止累加，T 为设定的阈值，实验中 T 值取 1。

对于无监督的训练方式，结合特征一致性 $\boldsymbol{C}_l$ 来定义颜色一致性损失：

$$L_{\text{ap}}^{(\boldsymbol{C})}=\sum_{l=4}^{0}\text{mean}(L_{\text{ap}}^{(l)}\cdot\boldsymbol{C}_l)+\text{mean}(1-\boldsymbol{C}_l)\text{mean}(L_{\text{ap}}^{(l)}),\tag{5}$$

式中：$L_{\text{ap}}^{(l)}$ 表示颜色一致性损失，计算公式为

$$L_{\text{ap}}^{(l)}=\text{mean}\left[\alpha\frac{1-F_{\text{SSIM}}(\boldsymbol{I}^{\text{wrap}},\boldsymbol{I})}{2}+(1-\alpha)(\|\boldsymbol{I}^{\text{wrap}}-\boldsymbol{I}\|_1)\right],\tag{6}$$

式中：$\boldsymbol{I}$ 为实际记录的图像；$\boldsymbol{I}^{\text{wrap}}$ 为使用相反视角图像和相应视差图对 $\boldsymbol{I}$ 进行重构的图像；$F_{\text{SSIM}}(\cdot)$ 函数为图像的结构相似度函数[28]；$\|\cdot\|_1$ 为 L1 范数；α 为权衡两种损失的权重因子，实验中取 $\alpha=0.85$。

最终的损失函数为

$$L_{\text{unsupervised}}=L_{\text{ap}}^{(\boldsymbol{C})}+w_{\boldsymbol{F}}^{1}L_{\boldsymbol{F}}+w_{\text{ds}}^{1}L_{\text{ds}},\tag{7}$$

式中：$w_{\boldsymbol{F}}^{1}$ 和 w_{ds}^{1} 为权衡特征一致性损失和局部平滑性损失的重要性权重，实验中分别取 1 和 0.1。

3　实验与分析

实验采用 SceneFlow 数据集[7]、KITTI 数据集[29-30]、Middlebury 数据集[31]和 ETH3d 数据集[32]对所提算法的性能进行评价和分析。首先，在 SceneFlow 数据集上对模型进行消融实验，分析模型的主要模块对算法性能的影响；然后，在 KITTI 训练集上分析损失函数对算法性能的影响，并在 KITTI 测试集上与几种典型的基于深度学习的立体匹配算法进行对比分析；最后，使用 KITTI 数据集、Middlebury 数据集和 ETH3d 数据集来分析模型的泛化性能。

对算法性能的评价指标主要包括 E_{ep}、E_{D1}、R_n 和 f_{run}，其中，E_{ep} 表示预测视差值与真值之间的差值绝对值；E_{D1} 表示每组图像对中评价区域的错误像素百分比，其中 E_{ep} 小于 3 或 E_{ep} 小于真值的 5% 则认为是正确像素，否则为错误像素；R_n 表示评价区域内 E_{ep} 大于 n 的像素百分比，n 为选择的误差阈值，实验中取 1、3、5；f_{run} 为算法每秒可处理的帧数。

3.1　实验细节

使用 PyTorch 编写算法（源代码见 https://github.com/wyf2017/DBSM），所有实验使用 Adam 优化器[33]以小批量随机梯度下降的方式进行训练，Adam 优化器的延迟率参数为 (0.9,0.999)，单次迭代的样本数为 6，最大视差 D 设为 192。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。其中，颜色增强包括色调增强、对比度增强、亮度增强、随机灰度化和随机高斯噪声添加；为保持双目图像的核线几何特性，空间变换增强只包括随机裁剪和随机翻转，随机裁剪的像素大小为 256 pixel×768 pixel。

为了方便实验描述，将所用数据集进一步分为 8 个子集：1) SceneFlow 测试集，共 4000 多组图像，记为 SF-val；2) SceneFlow 数据集中除 SF-val 之外的图像，共 35000 多组，记为 SF-train；3) KITTI2015 训练集的前 160 对图像和 KITTI2012 训练集的前 160 对图像，记为 K-train；4) KITTI2015 训练集和 KITTI2012 训练集中除 K-train 之外的 76 对图像，记为 K-Val；5) Middlebury 训练集中按场景名称排序后每隔 5 组取 1 组的图像集合，记为 MQ-val；6) Middlebury 训练集 1/4 分辨率的图像中除 MQ-val 之外的图像，记为 MQ-train；7) ETH3d 训练集中按场景名称排序后每隔 5 组取 1 组的图像集合，记为 ETH-val；8) ETH3d 训练集中除 ETH-val 之外的图像，记为 ETH-train。

实验主要分为三组。第 1 组为模型的消融实验，对 DRM 循环迭代次数、FEM 的不同版本和多任务学习模块进行分析，训练实验均在 SF-train 上进行，在 SF-val 上对算法性能进行对比和分析。第 2 组以在 SF-train 上训练的模型为基础，首先在 K-val 上对不同损失函数进行对比分析，然后在 K-train 和 K-val 上再对模型进行微调模型，并提交

KITTI 评价集的数据以与其他典型算法进行对比。第 3 组仍以在 SF-train 上训练的模型为基础，分别在 K-train、MQ-train、ETH-train 及其三者的并集上进行微调，并在 K-val、MQ-val 和 ETH-val 上分析模型的泛化性能。其中，在 SF-train 上的训练，先以学习率为 0.001 训练 30 个 epoch，再每隔 10 个 epoch 将学习率减半，总共训练 60 个 epoch；在 K-train、MQ-train、ETH-train 及其三者的并集上对模型的微调，均先以学习率为 0.0001 训练 300 个 epoch，再以学习率为 0.00001 训练 100 个 epoch；在 K-train 和 K-val 的并集上对模型的微调，以学习率 0.00002 训练 100 个 epoch。

3.2 模型的消融分析

在 DRM 中以循环迭代的方式来更新视差图，随着循环迭代次数的增加，算法精度得到提高，算法效率随之降低。将原始分辨率层级的迭代次数固定为 1，其他分辨率层级采用相同的迭代次数，不同迭代次数下算法的性能对比如表 1 所示。可以看出，当迭代次数从 1 增加为 2 时，算法误差明显降低，E_{ep} 从 1.02 pixel 降低为 0.85 pixel（相对降低了约 17%），E_{D1} 从 4.12% 降低为 3.56%（相对降低了约 14%）；当迭代次数从 2 继续增大时，并无明显的精度提升，但效率的降低较为明显，因此之后实验中迭代次数均取 2。

表 1 不同 DRM 迭代次数下算法的性能对比

Table 1 Performance evaluation of algorithm under each number of DRM iterations

Number of iterations	E_{ep}/pixel	E_{D1}/%	R_1/%	R_3/%	R_5/%	f_{run}/(frame·s^{-1})
1	1.02	4.12	11.67	4.63	3.38	15.84
2	0.85	3.56	10.06	4.13	2.82	12.22
3	0.83	3.52	9.85	4.02	2.80	8.64
4	0.82	3.46	9.62	3.87	2.81	6.22

以模型的完整结构为基础，在 FEM 中第二组卷积的不同设置情况下，算法的性能对比如表 2 所示，其中，当 FEM-2 使用基本残差块、SPP 和 SPPSA 时，对应模型分别记为 ResBlock、SPP 和 SPPSA。可以看出：与在 FEM-2 中采用 ResBlock 相比，采用多尺度特征融合层（SPP 或 SPPSA）时算法性能得到改善，当在 FEM-2 中采用 SPP 和 SPPSA 时，E_{ep} 从 0.87 pixel 分别降低为 0.86 pixel 和 0.85 pixel（相对降低了约 1% 和 2%），E_{D1} 从 3.75% 分别降低为 3.62% 和 3.56%（相对降低了约 3% 和 5%）。

表 2 不同 FEM 设置下算法的性能对比

Table 2 Performance evaluation of algorithm under different FEM settings

Number of iterations	E_{ep}/pixel	E_{D1}/%	R_1/%	R_3/%	R_5/%	f_{run}/(frame·s^{-1})
ResBlock	0.87	3.75	10.48	4.32	2.93	13.37
SPP	0.86	3.62	10.00	4.22	2.88	12.64
SPPSA	0.85	3.56	10.06	4.13	2.82	12.22

以模型的完整结构为基础，不同辅助任务设置下算法的性能对比如表 3 所示。其中，DRM 是以图 4 为基础，None 表示不输出任何辅助信息（即剪掉辅助任务的所有输出层），Con 表示保留辅助任务输出的特征一致性，edge 1 表示保留辅助任务输出的 edge 1，edge 2 表示保留辅助任务输出的 edge 2，edge 1＋edge 2＋Con 表示完整的模型结构。可以看出：尽管仅使用视差图标签数据，辅助任务的引入也在一定程度上改善了算法性能，与不采用任何辅助任务（设置为 None）相比，当采用所提算法的完整模型结构（设置为 edge 1＋edge 2＋Con）时，E_{ep} 从 1.05 pixel 降低为 0.85 pixel（相对降低了约 19%），E_{D1} 从 4.48% 降低为 3.56%（相对降低了约 21%）。

表 3 不同辅助任务设置下算法的性能对比

Table 3 Performance evaluation of algorithm under different auxiliary task settings

Number of iterations	E_{ep}/pixel	E_{D1}/%	R_1/%	R_3/%	R_5/%	f_{run}/(frame·s^{-1})
None	1.05	4.48	12.36	4.94	3.52	14.65
Con	1.00	4.17	11.65	4.80	3.32	13.58
edge 1＋Con	0.88	3.84	10.71	4.35	2.76	12.46
edge 2＋Con	0.89	3.85	10.65	4.37	2.98	13.23
edge 1＋edge 2＋Con	0.85	3.56	10.06	4.13	2.82	12.22

为了能更加直观地分析模型中的辅助任务输出，以 SF-val 中的一组图像为例，模型在原始空间分辨率层级上时辅助任务的输出和预测的视差图及其误差如图 5 所示，其中误差图中大于 3 的值被设为 3。可以看出，辅助任务模块可以预测较好的边缘图，如图 5(b)和图 5 (c)所示；在辅助任务中，预测的边缘图 1 比边缘图 2 含有更多的纹理边缘，边缘图 2 通过相关特征的比对能够很好地排除非视差不连续的纹理边缘，但也使得模型忽略了视差不连续变换较小的边缘，如图 5(b)和图 5(c)椭圆框所示；预测的特征一致性图可以很好地表征单视角可视区域，如图 5(d)所示；视差图中误差较大的区域大部分都位于特征不一致的区域，如图 5(d)和图 5(e)椭圆框所示。

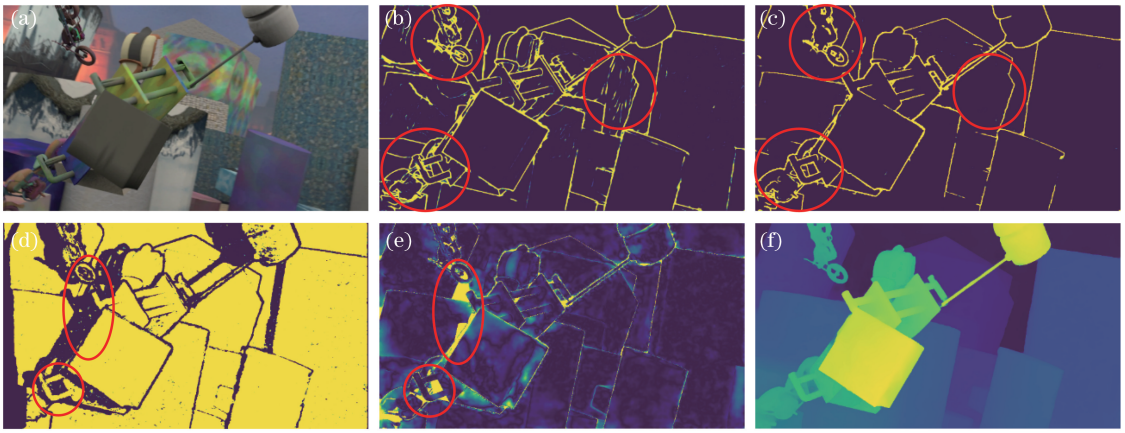

图 5　所提算法的可视化结果。(a)左图像；(b)边缘图 1；(c)边缘图 2；(d)特征一致性图；(e)误差图；(f)视差图

Fig. 5　Visual results of proposed algorithm. (a) Left image; (b) edge 1; (c) edge 2; (d) feature consistency map; (e) error map; (f) disparity map

3.3　KITTI 数据集上算法性能的对比

以在 SF-train 上训练的模型权重为基础，在 K-train 上微调模型，分析无监督学习方式中的特征一致性损失和平滑性损失对模型性能的影响，训练完成时在 K-val 上进行测试的评价，结果如表 4 所示。其中，SL1 表示采用(3)式作为损失函数训练模型(有监督学习方式)，其他设置均采用(7)式作为损失函数训练模型(无监督学习方式)；ap 表示(1)式中不考虑特征一致性图(即 $\boldsymbol{C}_l$ 的元素全为 1)，(2)式中不考虑边缘图(即 $\boldsymbol{B}_{l_2}$ 的元素全为 0)；ap+edge 表示(1)式中不考虑特征一致性图，(2)式中考虑边缘图(即 $\boldsymbol{B}_{l_2}$ 为模型输出)；ap+Con 表示(1)式中考虑特征一致性图(即 $\boldsymbol{C}_l$ 为模型输出)，(2)式中不考虑边缘图；ap+edge+Con 表示(1)式中考虑特征一致性图，(2)式中考虑边缘图。

表 4　不同训练损失下的性能评价

Table 4　Performance evaluation under different training losses

Training loss	E_{ep}/pixel	E_{D1}/%	R_1/%	R_3/%	R_5/%
ap	1.65	8.27	27.96	8.65	6.02
ap+edge	1.48	7.51	25.75	7.89	5.36
ap+Con	1.32	6.56	21.72	6.86	4.75
ap+edge+Con	1.28	6.04	19.95	6.23	4.09
SL1+edge+Con	0.71	2.56	13.25	2.87	1.69

从表 4 中可以看出，在无监督学习方式中，模型预测的特征一致性图和边缘图可以有效改善算法性能，与损失函数设置为 ap 相比，当设置为 ap+edge+Con 时，E_{ep} 从 1.65 pixel 降低为 1.28 pixel(相对降低了约 22%)，E_{D1} 从 8.27%降低为 6.04%(相对降低了约 27%)。与有监督学习方式相比，在无监督学习方式中，算法精度仍有较大差距，主要是因为：当特征不一致区域内(特别是被遮挡区域)存在不规则结构时，无监督学习方式无法得到有效的信息反馈，也就无法引导模型学习并预测准确的视差值。

为了能与其他典型算法进行直观的对比，取表 4 中损失函数设置为 SL1+edge+Con 时训练的模型，在 K-train 和 K-val 的并集上进行微调。使用微调完的模型预测 KITTI2015 测试集的视差图，并将其结果提交到 KITTI 数据集网站上进行在线评价，KITTI2015 测试集上的评价结果如表 5 所示，All

表 5 KITTI2015 测试集上不同算法的性能评价

Table 5 Performance evaluation of each algorithm on KITTI2015 test dataset

Algorithm	E_{D1}(All) /%			E_{D1}(Noc) /%			Runtime /s
	bg	fg	All area	bg	fg	All area	
M2S_CSPN[18]	1.51	2.88	1.74	1.40	2.67	1.61	0.50
EdgeStereo-V2[21]	1.84	3.30	2.08	1.69	2.94	1.89	0.32
WSMCnet[17]	1.72	4.19	2.13	1.51	3.57	1.85	0.39
SegStereo[19]	1.88	4.07	2.25	1.76	3.70	2.08	0.60
PSMNet[12]	1.86	4.62	2.32	1.71	4.31	2.14	0.41
iResNet-i2[9]	2.25	3.40	2.44	4.11	3.72	4.05	0.12
CRL[8]	2.48	3.59	2.67	2.32	3.12	2.45	0.47
GC-net[11]	2.21	6.16	2.87	2.02	5.58	2.61	0.90
MBFnet[22]	2.59	4.80	2.96	2.22	4.14	2.54	0.05
SGM-Net[6]	2.66	8.64	3.66	2.23	7.44	3.09	67.00
MC-CNN-arct[2]	2.89	8.88	3.89	2.48	7.64	3.33	67.00
DispNetC[7]	4.32	4.41	4.34	4.11	3.72	4.05	0.06
Proposed algorithm	2.07	4.01	2.39	1.89	3.69	2.19	0.09

表示评价时包含所有像素，Noc 表示只考虑非遮挡区域内的像素，bg 表示只考虑背景区域，fg 表示只考虑前景区域。其中，对所提算法 Runtime 的统计以 RTX2070 显卡为主要硬件。可以看出：所提算法在算法精度和运行效率上均具有一定的竞争力，与两种典型的基于多任务学习的立体匹配算法（SegStereo 算法[19] 和 EdgeStereo-V2 算法[21]）相比，虽然误差率略高，但在硬件性能较差的情况下运行效率仍具有明显优势，并且所提算法不依赖其他任务训练的模型权重，训练过程中也不需要相应辅助任务的标签数据，同时可以为遮挡区域的判断提供重要依据，这对后期场景几何的恢复具有积极作用。与文献[22]相比，所提算法通过引入辅助任务来提高精度，虽然增加了模型复杂度，但模型可以预测边缘信息和特征一致性信息，为模型的无监督学习提供了更多有益信息，并且可以为遮挡区域的检测提供重要依据。

3.4 泛化性能的分析

以在 SF-train 上训练的模型权重为基础，在不同数据集上微调后的算法性能对比如表 6 所示。其中，Pretrained 表示评价时使用 SF-train 上训练的模型权重，KME-train 表示对模型微调时使用 K-train、MQ-train 和 ETH-train 的并集。可以看出，在新场景中对模型权重的微调能够提高算法性能，如在 KME-train 上微调模型后，与预训练的模型相比，各个数据集上的评价指标均有一定提升；算法性能对数据集仍具有较强的依赖性，从同类型的数据上来看，微调模型能提高算法性能，但也会造成模型的过拟合，降低模型在其他场景数据上的精度。

表 6 不同数据集上的性能评价

Table 6 Performance evaluation on each dataset

Dataset	K-val		MQ-val		ETH-val	
	E_{ep}/pixel	E_{D1}/%	E_{ep}/pixel	E_{D1}/%	E_{ep}/pixel	$ED1$ /%
Pretrained	1.48	7.05	0.72	4.99	0.65	4.77
K-train	0.71	2.56	1.06	7.20	0.51	1.66
MQ-train	1.72	9.38	0.47	2.58	0.59	2.69
ETH-train	1.84	10.85	1.32	8.36	0.32	1.26
KME-train	0.78	3.01	0.49	2.61	0.36	1.33

4 结 论

所提算法在视差细化阶段引入了边缘和特征一致性信息，使模型能够学习到更明确的边缘和遮挡信息，辅助任务的引入使算法误差相对降低了约 20%。在模型训练的过程中，根据图像的自相关性构建损失函数，从而引导模型进行辅助任务的学习，模型可以在不需要任何额外数据的情况下有效学习边缘和特征一致性信息，解决了多任务学习对辅助标签数据的依赖。在无监督训练方式中，辅助任务

可以提供单视角可见区域的信息，从而排除特征不一致性产生的反馈噪声。在算法的泛化性能方面，模型权重的微调可以改善同类型数据集上的算法性能，同时也会使模型存在偏好性，降低其他类型数据的算法精度，同时采用多种类型数据训练模型可以使算法性能得到较好的均衡。

参 考 文 献

[1] 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, 2015: 1592-1599.

[2] Žbontar J, LeCun Y. Stereo matching by training a convolutional neural network to compare image patches[J]. Journal of Machine Learning Research, 2016, 17(1): 2287-2318.

[3] Žbontar J, LeCun Y. Stereo matching by training a convolutional neural network to compare image patches [EB/OL]. (2015-10-20) [2020-01-17]. https://arxiv.org/abs/1510.05970.

[4] Park H, Lee K M. Look wider to match image patches with convolutional neural networks[J]. IEEE Signal Processing Letters, 2017, 24(12): 1788-1792.

[5] 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, 2016: 5695-5703.

[5] Seki A, Pollefeys M. Patch based confidence prediction for dense disparity map[C]//Proceedings of the British Machine Vision Conference 2016, September 19-22, 2016, York, UK. London: British Machine Vision Association, 2016: 23.1-23.13.

[6] Seki A, Pollefeys M. SGM-nets: semi-global matching with neural networks [C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 6640-6649.

[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, 2016: 4040-4048.

[8] 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, 2017: 878-886.

[9] Liang Z F, Feng Y L, Guo Y L, et al. Learning for disparity estimation through feature constancy [C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE, 2018: 2811-2820.

[10] Jie Z Q, Wang P F, Ling Y G, et al. Left-right comparative recurrent model for stereo matching[C]// 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE, 2018: 3838-3846.

[11] 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, 2017: 66-75.

[12] 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, 2018: 5410-5418.

[13] Nie G Y, Cheng M M, Liu Y, et al. Multi-level context ultra-aggregation for 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, 2019: 3278-3286.

[14] Guo X Y, Yang K, Yang W K, et al. Group-wise correlation stereo network [C] // 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 15-20, 2019, Long Beach, CA, USA. New York: IEEE, 2019: 3268-3277.

[15] Tulyakov S, Ivanov A, Fleuret F. Practical deep stereo (PDS): toward applications-friendly deep stereo matching [EB/OL]. (2018-06-05) [2020-01-17]. http://cn.arxiv.org/abs/1806.01677.

[16] Duggal S, Wang S L, Ma W C, et al. DeepPruner: learning efficient stereo matching via differentiable PatchMatch [C] // 2019 IEEE/CVF International

Conference on Computer Vision (ICCV), October 27-November 2, 2019, Seoul, Korea (South). New York: IEEE, 2019: 4383-4392.

[17] Wang Y F, Wang H W, Yu G, et al. Stereo matching based on 3D convolutional neural network [J]. Acta Optica Sinica, 2019, 39(11): 1115001.
王玉锋, 王宏伟, 于光, 等. 基于三维卷积神经网络的立体匹配算法[J]. 光学学报, 2019, 39(11): 1115001.

[18] Cheng X J, Wang P, Yang R G, et al. Learning depth with convolutional spatial propagation network [EB/OL]. (2018-10-04)[2020-01-17]. http://cn.arxiv.org/abs/1810.02695.

[19] Yang G R, Zhao H S, Shi J P, et al. SegStereo: exploiting semantic information for disparity estimation[M]// Ferrari V, Hebert M, Sminchisescu C, et al. Computer vision — ECCV 2018. Lecture notes in computer science. Cham: Springer, 2018, 11211: 660-676.

[20] Song X, Zhao X, Hu H W, et al. EdgeStereo: a context integrated residual pyramid network for stereo matching [M]// Jawahar C, Li H, Mori G, et al. Computer vision—ACCV 2018. Lecture notes in computer science. Cham: Springer, 2019, 11365: 20-35.

[21] Song X, Zhao X, Fang L J, et al. EdgeStereo: an effective multi-task learning network for stereo matching and edge detection[EB/OL]. (2019-03-05)[2020-01-17]. http://cn.arxiv.org/abs/1903.01700.

[22] Wang Y F, Wang H W, Liu Y, et al. Real-time stereo matching with a hierarchical refinement [J]. Acta Optica Sinica, 2020, 40(9): 0915002.
王玉锋, 王宏伟, 刘宇, 等. 渐进细化的实时立体匹配算法[J]. 光学学报, 2020, 40(9): 0915002.

[23] Zhao H S, Shi J P, Qi X J, et al. Pyramid scene parsing network [C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 6230-6239.

[24] 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, 2016: 770-778.

[25] Ioffe S, Szegedy C. Batch normalization: accelerating deep network training by reducing internal covariate shift[EB/OL]. (2015-02-11)[2020-01-17]. https://arxiv.org/abs/1502.03167.

[26] Maas A L, Hannun A Y, Ng A Y. Rectifier nonlinearities improve neural network acoustic models [EB/OL]. (2013-06-16)[2020-01-17]. https://ai.stanford.edu/~amaas/papers/relu_hybrid_icml2013_final.pdf.

[27] Girshick R. Fast R-CNN [C]// 2015 IEEE International Conference on Computer Vision (ICCV), December 7-13, 2015, Santiago, Chile. New York: IEEE, 2015: 1440-1448.

[28] Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity [J]. IEEE Transactions on Image Processing, 2004, 13(4): 600-612.

[29] 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, June 16-21, 2012, Providence, RI, USA. New York: IEEE, 2012: 3354-3361.

[30] 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, 2015: 3061-3070.

[31] Scharstein D, Hirschmüller H, Kitajima Y, et al. High-resolution stereo datasets with subpixel-accurate ground truth[M]// Jiang X, Hornegger J, Koch R. GCPR 2014: pattern recognition. Lecture notes in computer science. Cham: Springer, 2014, 8753: 31-42.

[32] Schöps T, Schönberger J L, Galliani S, et al. A multi-view stereo benchmark with high-resolution images and multi-camera videos [C]// 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 2538-2547.

[33] Kingma D P, Ba J. Adam: a method for stochastic optimization[EB/OL]. (2014-12-22)[2020-01-17]. http://cn.arxiv.org/abs/1412.6980.