

### 毕业设计外文资料译文（含原文）

（外文资料出处：*arXiv:1909.03751v2 [cs.CV] 19 Nov 2019*）

学 院：软件学院

专业名称：软件工程

班级学号：16206117

学生姓名：黄凯

指导教师：储珺

2020年03月

# 基于深度立体匹配的自适应单峰成本量滤波

张佑民1陈一敏1\*白晓1余隋汉金1 余昆2李志伟2元奎远2

1计算机科学与工程学院软件开发环境国家重点实验室，北京大数据与脑计算高级创新中心，

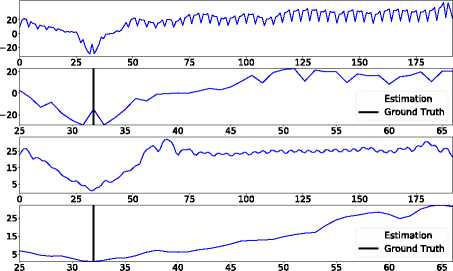
北京航空航天大学江西研究所，北京，中国2

(youmi, minwellcym, baixiao, fakecoderemail} @buaa.edu.cn {kunyu, zhiweili, kuiyuanyang} @deepmotion.a)i

**摘要**

基于最新深度学习的立体匹配方法将视差估计视为回归问题，其中损失函数直接根据真实视差及其估计的视差定义。但是，差异只是按成本量建模的匹配过程的副产品，而由于成本量受到限制，因此间接学习由差异回归驱动的成本量容易过拟合。在本文中，我们建议通过过滤在真实差异上达到峰值的单峰分布的成本量e,直接对成本量添加约束。此外，估计每个像素的单峰分布的方差，以明确地建模不同上下文下的匹配不确定性。所提出的体系结构在场景流和两个KITTI立体基准上达到了最先进的性能。特别值得一提的是，我们的方法结果成为名列第一KITTI 2012评估的结果以及KITTI 2015评估的第4位（记录于2019.8.20）。AcfNet的代码位于：https://github.com/DeepMotionAIResearch/DenseMatchingBenchmarko

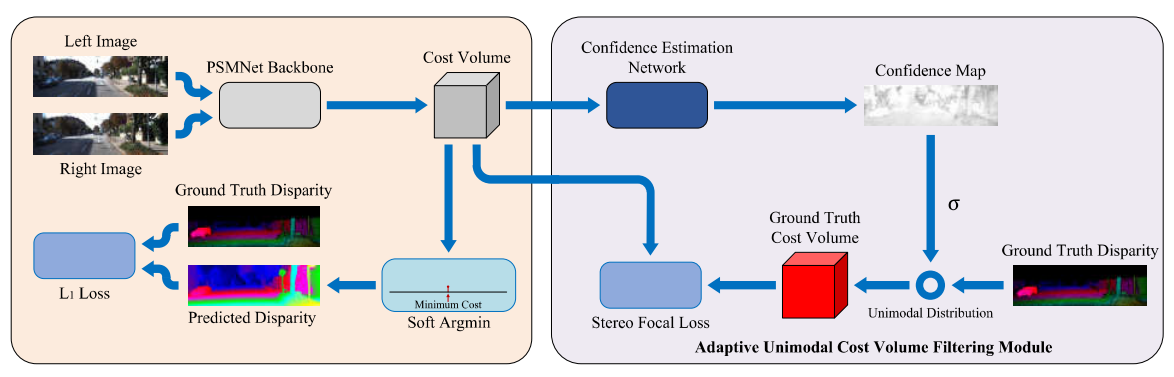
**介绍**

立体匹配是计算机视觉的核心技术之一，可以从2D图像中恢复现实世界的3D结构。它已广泛用于自动驾驶（Sivaraman和Trivedi，2013），增强现实（Zenati和Zerhouni ，2007）和机器人导航（Schmid等人，2013; Luo，Yu和Ren，2017; Luo等人，2019）。给定一对校正后的立体图像，立体匹配的目标是计算参考图像（通常是指左侧图像）中每个像素的视差，其中视差定义为像素中一对对应像素之间的水平位移左右图像。

图表 1 沿成本差异缩小的成本分布示例。第一行是经过soft-argmin训练的PSMNet的输出（Chang 和 Chen 2018）。第三行是模型的输出。为了更好地可视化，我们还放大了第二行和第四行的视差间隔[25、66]，其中“估计” 和“地面实况”分别是估计值和“地面实况”视差。我们的方法产生了一个更合理的成本分配，达到了真正的差异。

根据研讨会的探讨得出（Scharstein和Szeliski 2002）， 立体匹配算法通常包括四个步骤：匹配成本计算，成本合计，视差回归和视差细化。其中，匹配成本计算即获得成本量为可以说是最关键的第一步。成本量通常由张量表示，其中是参考图像的高度，宽度和最大视差。在传统方法中，成本量是通过人工设计的图像特征的预定义成本函数来计算的，例如，图像块的平方或绝对差。

在深度学习时代，图像功能和成本功能都被建模为网络层（Kendall等人2017; Chang和Chen 2018）。为了使所有层可区分并实现子像素的视差估计，使用了soft-argmin来根据它们的成本对权重进行细微地加权来估计视差，这与使用最小成本的指数作为估计视差的argmin相反。损失函数取决于估计的差异和端到端训练的背景实况。受益于大规模训练数据和端到端训练，基于深度学习的立体方法可实现最新的性能。



图表 2 提议的端到端AcfNet的体系结构。输入的立体图像被馈送到具有堆叠沙漏架构的PSMNet（Chang and Chen 2018） 主干，以在聚合后获得三个成本量。对于每个成本量，我们通过置信度估计网络（CENet）生成置信度图，并使用置信度值对地面真实成本量进行调制，以生成像素级单峰分布作为训练标签。使用生成的训练标签将拟议的立体焦损添加到成本量中。最后，通过soft-argmin函数估算亚像素视差图，然后将其作为PSMNet进行回归损失。

在深度学习模型中，成本量是作为中间层进行正确的监督，这使成本数量受到的约束较少，因为无限多的成本分布都可以产生相同的差异，其中只有在实际差异上达到峰值的成本分布才是合理的。因此，我们建议用单峰地面实况分布直接监督成本量。为了揭示不同像素的网络匹配不确定性（Kendall 和Gal 2017; Ilg等2018），我们设计了一个置信估计网络来估计每个像素的置信度和相应地控制单峰地面真实分布的控制清晰度。图1比较了PSMNet和我们的方法在同一像素处的成本分布，其中我们的方法围绕真实差异生成了正确的最小成本，而PSMNet却从真实差异生成了两个局部最小成本。

我们在包括场景流，KITTI 2012和KITTI 2015在内的三个立体声基准上评估了拟议的自适应单峰成本量过滤网络（AcfNet）。消融研究和对场景流的详细分析证明了AcfNet的有效性。我们还将立体声匹配结果提交给KITTI 2012和2015评估服务器，以及在KITTI 2012评估中名列第1位，而在KITTI 2015评估中排第4名（记录于2019.8.20）。

**相关工作**

立体匹配的深度学习从学习经典方法的图像特征开始（ Zbontar 和 LeCun 2016; Luo， Schwing 和Urtasun 2016）。DispNetC（Mayer等人，2016）是立体声匹配的第一个突破，它提出了一个端到端可训练网络，其中成本函数被预先定义为网络中的一个相关层以生成成本量，然后是一组卷积层被添加到成本量以回归差异图。基于DispNetC，提出了栈优化子网络来提高性能（Pang等人2017; Liang等人2018），并且该性能可能会进一步提高。通过使用附加信息来证明这些优势（Song等人2018）和语义（Yang等人2018）。为了增加网络学习成本函数的能力，（Guo 等人.2019）建议使用逐组相关层并生成多个成本量进行后期聚合。

GC-Net（Kendall 等人2017）通过在级联特征量上使用3D卷积层，为网络提供了更多的学习成本函数的灵活性，并通过学习的成本函数产生了成本量，根据成本通过soft-argmin估计差异分配。后续工作通过使用更好的图像功能（Chang和Chen，2018）以及以传统方法为灵感的成本汇总层（Cheng， Wang和Yang，2018; Zhang等，2019）来改善结果。在这些端到端立体声匹配网络中，成本量是中间层的输出而没有直接的监督，这使得学习不合理的成本分布成为可能，如图1所示。

在这项工作中，拟议的AcfNet使用在真实分布上达到顶峰的地面真实成本分布，直接在成本量估算中添加了超视野。另外，根据匹配的置信度来调整地面真实成本分配的清晰度。与我们的工作一致，稀疏LiDAR点用于通过加权高斯分布的加权估计成本分布来提高成本量，高斯分布的中心是由其对应的LiDAR点提供的视差（Poggi 等人2019），这是一种多传感器融合方法用于视差估计。相反，我们的方法在训练和测试期间仅将图像作为输入，并且以密集且自适应的方式将单峰监督添加到每个像素。

**AcfNet**

图2说明了总体框架，其中将建议的自适应单峰成本量过滤模块应用于成本量，并引入了额外的损失直接监督对所需财产的成本量学习。在这里，我们选择PSMNet（Chang和Chen，2018年）作为基础网络，以计算其立体匹配的最新性能的成本量。

**总览**

给定一对校正图像，对于每个像素左图是立体匹配，目的是找到它的余弦。右图像中的像素，即，其中视差通常由用于子像素匹配的浮点数表示。对于计算和可处理的内存，视差被分为一组可能的视差，即来构建成本量，其中是图像高度，宽度和最大视差。为了恢复子像素匹配，在加权插值中使用视差上的成本。整个过程是通过网络实现的，如图2左侧所示。

形式上，成本量包含每个像素的成本，用表示，并且子像素视差通过soft-argmin 估算（Kendall等人，2017）

|  |  |  |
| --- | --- | --- |
|  |  | (1) |

其中 和差距较小的成本在插值过程中贡献更多。给定每个像素的地面差异，定义了平滑的损失进行训练，即

其中

整个过程可以通过监督地面真实差异来区分，而成本量则通过提供差异插值的权重间接地进行监督。但是监督工作不够明确，可能存在无限可能的权重集以实现正确的插值结果。成本量的灵活性很容易过度拟合，因为许多不正确学习的成本量可能会插补接近地面实况的差异（即，培训损失很小）。

为了解决由不确定的损失函数间接监督成本量引起的问题，我们建议根据其单峰性质直接监督成本量。

**单峰分布**

定义成本量以反映候选匹配像素对之间的相似性，其中真正匹配的像素对应具有最低的成本（即，最高的相似性），并且成本应随与真正匹配的像素的距离而增加。此属性要求单峰分布在成本中每个位置的真实差异处达到峰值体积。给定地面真实视差，将单峰分布定义为

|  |  |
| --- | --- |
|  | (4) |
|  |

其中 是控制真实视差附近峰的锐度的方差（文献中又称为温度）。

由构造的地面真实成本量在不同像素上具有相同的峰值锐度，无法反映不同像素之间的相似性分布差异。例如，桌面上的像素应具有非常尖锐的峰，而均匀区域中的像素应具有相对平坦的峰。为了为成本量构建更多合理的标签，我们添加了一个置信度估算网络来自适应地预测每个像素的。

**置信度估算网络**

考虑将匹配属性嵌入到估计的成本量中（Fu和Fard 2018; Park和Yoon 2018; Kim等人2018），那么置信度估计网络将估计的成本量作为输入，并使用一些通过检查每个像素周围小的邻域中的匹配状态来确定每个像素的匹配置信度。具体来说，该网络采用3 x 3卷积层，然后进行批量归一化和ReLU激活，再通过Sigmoid激活后再使用1 x 1卷积层，以产生置信度图，其中像素具有大置信值，表示可以确信地找到此像素的唯一匹配，而小的置信度值表示存在匹配项产生歧义。然后，用于产生地面真实成本根据估计的置信度缩放分布

|  |  |  |
| --- | --- | --- |
|  |  | (5) |

其中是一个比例因子，反映了对置信度变化的敏感性，定义了的下界，避免了除以0的数值问题。因此，。我们的实验表明，两种像素可能具有较大的，即无纹理像素和遮挡像素，其中无纹理的像素往往具有多个匹配项，而遮挡像素则没有正确的匹配项。使用每个像素的自适应估计时，等式中定义的地面真实成本量。（4）根据需要进行修改。

**立体焦点损失**

在像素位置，我们现在同时具有估计的成本分布和地面真实性。容易通过交叉熵定义分布损失。 然而，存在一个严重的样本失衡问题，因为与数百个负像素相比，每个像素只有一个真实的视差（正）（Zbontar和LeCun 2016）。为了解决一阶段物体检测中的样本不平衡问题（Lin等人，2017），我们与聚焦损耗类似，我们设计了一个立体焦距损耗集中在正视差上，以避免由负视差主导的总损耗，

|  |
| --- |
|  |

其中是一个聚焦参数，当时损耗减小为交叉熵损耗，而则给正视差的权重与它们的成正比。因此，可以用很小的权重进一步明确地抑制容易产生的负差异，而使正差异仅能与少数硬差异竞争。

**总损失函数**

总而言之，我们的最终损失函数包含三个部分，分别为

|  |  |
| --- | --- |
|  | (7) |

其中， 是两个权衡的超参数。是监督成本量，而是监督差距。添加了作为正则化器，以刺激更多像素具有较高的置信度值，

|  |  |
| --- | --- |
|  | (8) |

**实验与分析**

**实施细节**

我们的网络是使用PyTorch（Paszke等人.2017）框架实现的，并且所有模型都使用RMSprop和标准设置进行了端到端训练。我们的数据处理与PSMNet相同（Chang 和 Chen 2018）。我们使用Scene Flow数据集从头开始训练模型，其中10个时期的学习率为0.001。对于Scene Flow，训练后的模型直接用于测试。对于KITTI，在对600个纪元的KITTI训练集进行微调后，我们使用通过场景流数据训练的模型。这种微调的学习率从0.001开始，分别在100和300个时代被引起。为了服从KITTI测试基准，我们以20个时期的0.001恒定学习率延长了“场景流” 的训练过程，以获得更好的预训练模型。批量大小设置为3，可以在3个NVIDIA GTX 1080Ti GPU上进行训练。我们的实验排除了范围之外的所有地面真实差异，其中。

**数据集**

我们在三个具有挑战性的立体声基准上对AcfNet进行定性和定量评估，即Scene Flow（Mayer等人,2016）， KITTI 2012（Geiger， Lenz和Urtasun 2012）和KITTI 2015（Menze和Geiger，2015）。

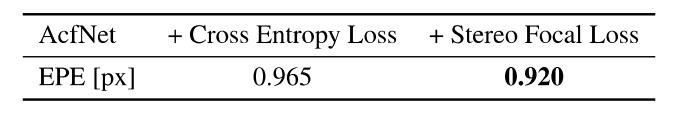
**场景流：**“ 场景流” 是一个大型的综合数据集，包含35,454个训练图像对和4,370个测试图像对，其中密集提供了地面真实差异图，足够用于直接训练深度学习模型。设置为GC-Net（Kendall等人，2017年）后，我们主要将此数据集用于消融研究。

**KITTI：**KITTI2015和KITTI2012是两个真实的数据集，具有从驾驶汽车捕获的街道视图。KITTI2015包含200幅使用LiDAR获得的稀疏地面视差的训练立体图像对，以及200颗由评估服务器持有的地面真实视差的测试图像对，仅用于提交评估。KITTI2012包含194个具有稀疏地面真实差异的训练图像对和195个具有地面真实差异的测试图像对，它们由评估服务器保存，仅用于提交评估。这两个数据集的规模较小，因此具有挑战性。指标：使用两种标准指标来衡量性能：（1）3-Pixel- Error（3PE），即，其预测的视差与真实像素相差超过3个像素的像素的百分比，以及（2）端点错误（EPE），即预测差异的平均差异及其真实值棒性些。3PE对于具有较大差异误差的离群值具有较强的鲁而EPE会将误差测量到亚像素级别。进一步评估处理遮挡物的能力各个方面，我们通过左右一致性检查将“场景流”的测试图像分为遮挡区域（OCC）和非遮挡区域（NOC）。总共，所有像素中有16％的被遮挡像素。如果没有前缀，则对所有像素测量性能。例如OCC，NOC和ALL， 在3PE或EPE之前添加。

**消融研究**

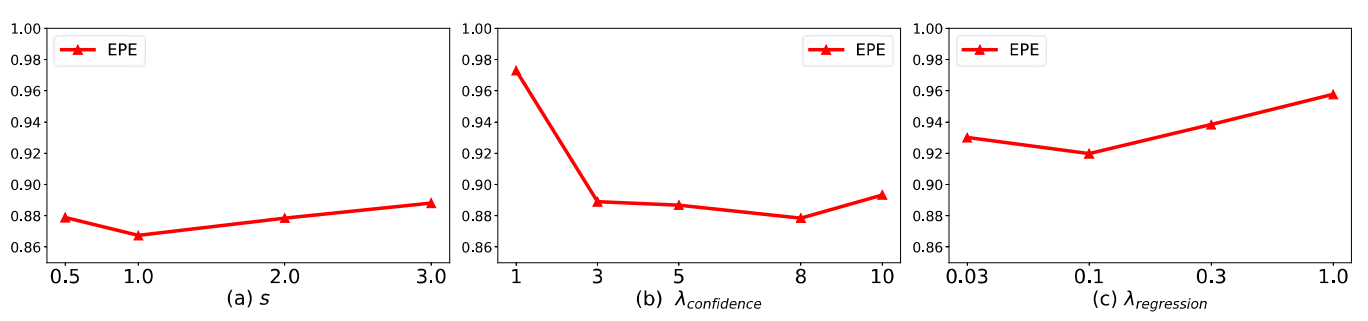
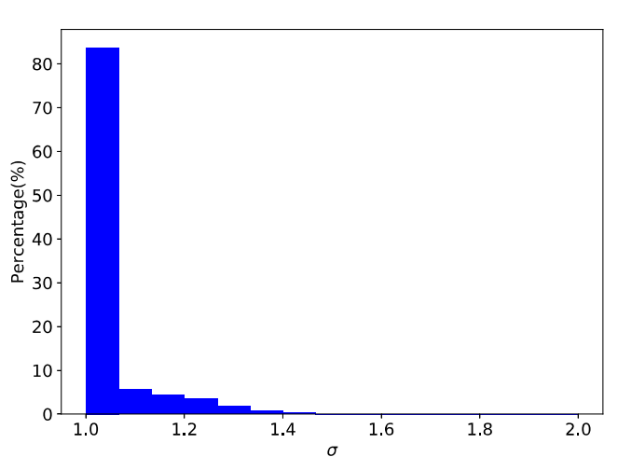
我们对场景流进行消融研究（Mayer等人，2016年），因为它具有足够大的训练数据，可以从头开始进行端到端训练。在所有实验中，将a设置为5.0立体焦点损失，以平衡正负样本。考虑到大多数像素的视差具有亚像素误差（即小于1个像素的误差），而3PE不能揭示3个像素以内的误差，因此我们使用EPE研究不同超参数设置的性能差异。

表格 1 在我们的模型AcfNet中，立体焦点损失和交叉熵损失之间的比较结果。



**单峰分布的方差**

方差调整单峰分布的形状，这在AcfNet中起着重要作用。在我们的方法中，由和界定。

首先，我们研究了所有像素方差均固定的情况，即。通过网格搜索，我们发现可获得最佳结果，这表明大多数像素都赞成用于建立单峰分布分布。因此，我们为自适应方差研究将的下限设置为1.0。此外，我们在这种情况下（即）比较了立体焦距损耗和交叉熵损耗。如表1所示，为AcfNet配备立体声焦距损失比交叉熵损失要好得多，这表明立体声焦距损失在平衡正负视差造成的损失中的有效性。

图表 4整体方差的直方图分布在AcfNet收敛后测试场景流的数据集。

图表 3在我们的方法中，不同的超参数的消融研究结果，其中控制方差的上界 。约束和 分别是置信度损失和视差回归损失的平衡权重。

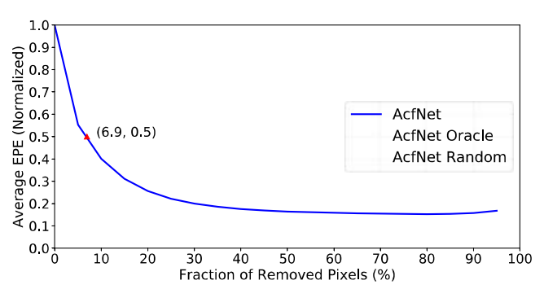
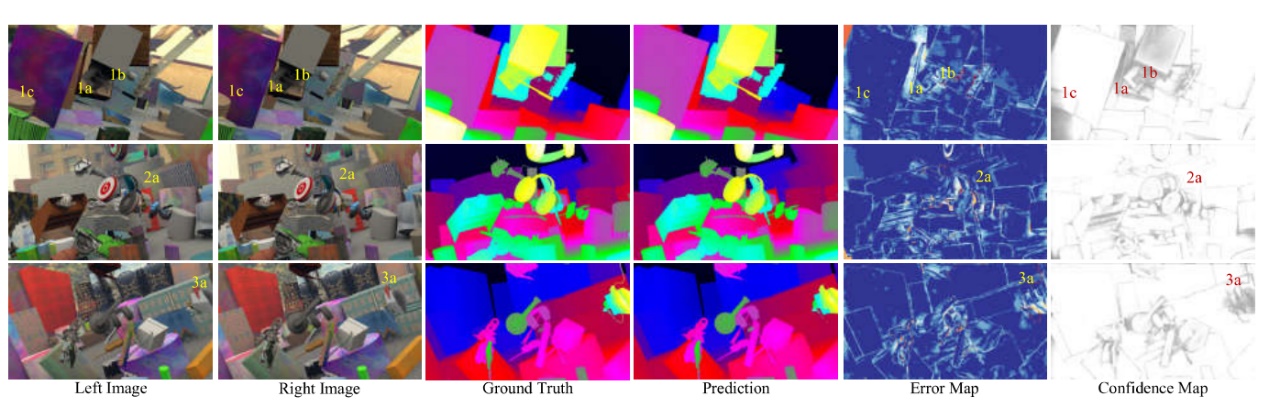
其次，我们研究控制上限的灵敏度。图3（a）显示了通过改变的性能，其中表现最佳性能，通过将从0.5更改为3.0可以相当稳定。图4显示了当（即）时的直方图，其中大多数像素偏爱较小的视觉效果（即锐利的分布），而证明了立体焦点损失在平衡正负差异损失方面的模具有效性。

**损失平衡权重**

财务上的超参数可平衡总方差和其他损失数字。图3（b）显示了通过改变信任度的性能曲线，其中过 大的信任信任度学习和信心不足-使用小征服导致性能较差，而表现最佳。

超参数可平衡最近使用的最新模型中广泛使用的回归损失,而的较大值将消除本文提出的其他两个损失的影响。图3（c）显示了性能曲线，可以观察到通过适当地权衡建议的两个损耗可以改善回归损耗。

**方差分析**

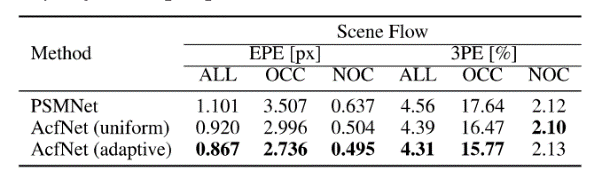
方差估计是我们成本过滤方案的重要组成部分，该方案 会根据匹配的不确定性自动调整单峰分布的平坦度。为了评估估计方差的质量，采用了晶格大小分布图（Ilg等 人2018），通过绘制评估结果，通过逐渐消除像素的方 差来绘制估计结果，从而揭示具有真实误差的估计方差 的相关性。为了进行比较，我们还在图5中绘制了随机分 配的方差（AcfNetRandom）和EPE主体（AcfNetOracle）分配的方差的曲线，其中估计方差与EPE错误高度相关，并证明了AcfNet解释具有估计方差的离群点像素的能力。

图表 6 来自“场景流”测试集中的三个样本的定性结果。从左到右的列是：左立体声输入图像，右立体声输入图像，视差地面真相，视差预测，误差图和置信度图。误差图中的冷色表示较小的预测误差，而暖色表示较大的预测误差。在置信度图中，亮色表示较小的方差，而暗色表示较高的方差。

图表 5图表 5 场景流测试数据集上AcfNet的稀疏度图。该图显示了具有最高有效值的像素的每个像素部分的归一化平均端点或（EPE）。曲线“ AcfNet Oracle”通过删除像素的每个部分显示了理想的情况由地面实况EPE排名。曲线“ AcfNet Ran dom”显示了最坏的情况，即随机删除了每个像素。通过AcfNet仅删除6.9％的像素可将平均EPE减半。

图6显示了“场景流”的每个像素的几个结果，其中硬区域主要出现在遮挡（la，le和2a），重复图案（lb，3a）和薄结构（3a）上。在这些困难地区，AcfNet提供了很大的差异，可以平分十个相应的成本分布。AcfNet 可以平衡不同像素的学习，并向高置信度（即低方差） 推动信息像素，而降低具有高方差的硬无信息像素，以避免过度拟合。

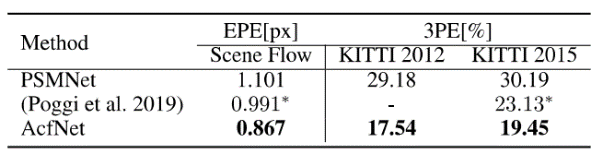
**自适应单峰成本量过滤**

AcfNet在PSMNet中增加了直接的成本量监管。表2比较了AcfNet和PSM Net的两个版本，其中AcfNet的统一版本比PSMNet明显更好，而AcfNet的自适应版本则进一步提高了性能。结果证明了单峰监督和自适应每像素空位估计的有效性。与AcfNet（统一）相比，AcfNet（自适应） 在OCC（即被遮挡区域）上有更多改进，这与方差分析的结论是一致的。

表格 2 对自适应单峰成本量过滤结果的评估，其中重新实现了PSMNet。AcfNet（uniform）表示为所有像素设置统一的单峰分布，而AcfNet（adaptive）表示自适应地调整每个像素的方差。

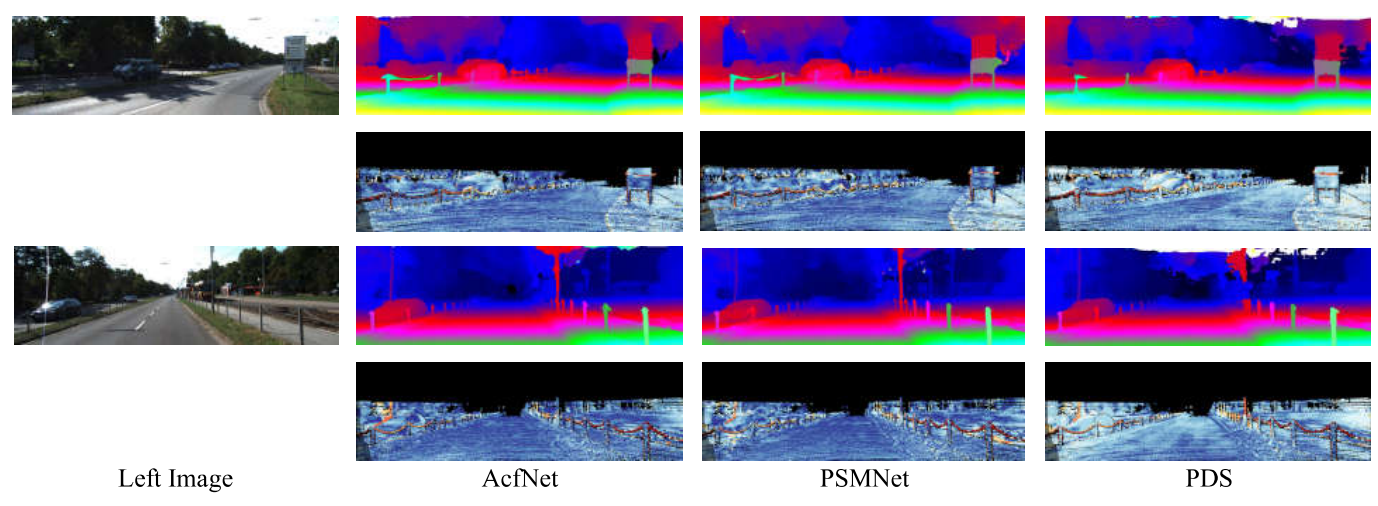
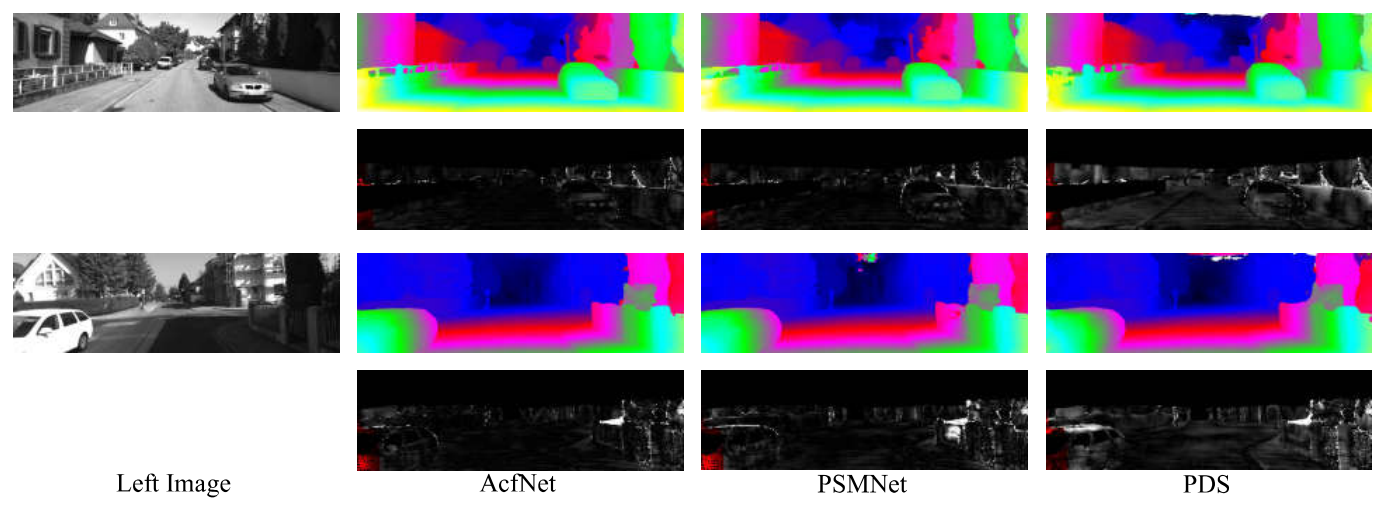
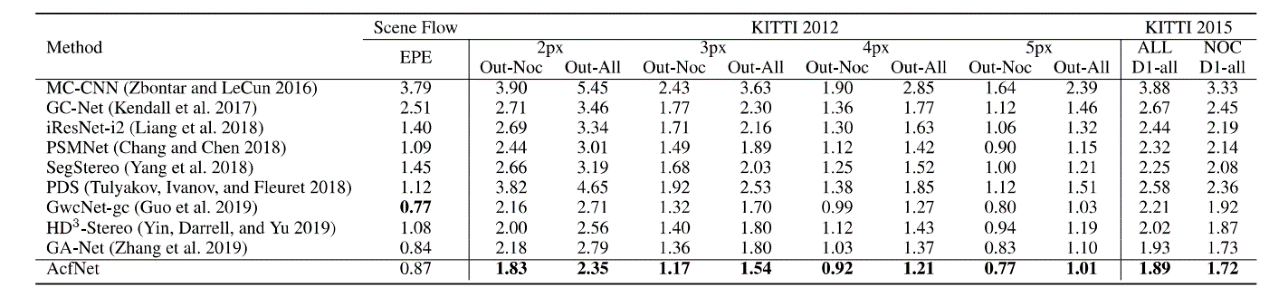
**成本量过滤比较**

为了进一步验证提议的成本量过滤的优越性，设计了实验与本研究进行比较（Poggi 等.2019）。与我们的工作相反，（Poggi 等.2019）通过稀疏LiDAR点使用视差来过滤训练和测试期间的成本量。从头开始对AcfNet和（Poggi 等.2019）的方法都进行了场景流的训练，并且自（Poggi 2019）需要稀疏的LiDAR点作为输入。表3报告了比较结果，即使不使用LiDAR点作为输入，AcfNet在所有性能指标上的表现也要好于大幅度（Poggi 等.2019）。另外，与PSMNet相比，AcfNet从Scene Flow到KITTI具有更好的生成能力，这进一步证明了AcfNet防止过度拟合的能力。



表格 3 成本量过滤比较的结果，其中所有方法都使用相同的基本模型PSMNet从头开始在Scene Flow上进行训练，并直接在KITTI 2012、2015训练数据集上进行测试。\*表示稀疏LiDAR点的差异，在测试时也用作模型输入。

**与最新方法的比较**

为了进一步验证提议的AcfNet，表4将AcfNet与KITTI 2012和2015上的最新方法进行了比较，其中AcfNet在所有评估指标上的显着优势均优于其他方法。需要注意的是，考虑到KITTI训练数据的规模小，场景流用于所有方法的预训练。图7和图8通过将AcfNet与PSMNet（Chang and Chen 2018）和PDS（Tulyakov，Ivanov 和Fleuret 2018）进行了比较，显示了KITTI 2015和2012的一些典型结果，其中用虚线框标出了明显改善的区域。不出所料，AcfNet的大多数改进都来自挑战性领域，例如薄型结构，天空边界和图像边界。

图表 8 KITTI 2015数据集上的可视化结果。显着改善的区域以虚线框突出显示。对于每个示例，第一行显示视差图，第二行显示误差图。较暖的颜色表示较大的预测误差。

图表 7 在KITTI 2012数据集上的可视化结果。显着改善的区域以虚线框突出显示。对于每个示例，第一行显示视差图，第二行显示误差图，明亮的颜色表示不准确的预测。

表格 4 场景流和KITTI基准测试的结果。遵循标准设置，在KITTI 2012上报告了非遮挡（Out-Noc）像素和所有（Out All）像素的错误像素百分比，在KITTI 2015上，报告了所有地面真实情况的平均视差离群值D1的百分比报告非遮挡像。

**结论**

在本文中，我们解决了成本约束不足的问题。

现有基于深度学习的立体声匹配方法的数量。拟议的AcfNet会以真实的视差达到峰值的地面实况单峰分布来监督成本，并根据每个像素的形成性适当估计每个像素分布的方差，以调制学习。AcfNet在相同数据集上显示出更好的测试性能，甚至在跨数据集评估上也表现出卓越的性能。

**致谢**

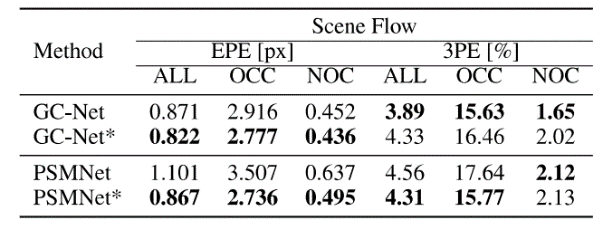
这项工作得到了国家自然科学基金项目No.61772057和国家重点实验室的支持资金。北京航空航天大学软件开发环境与江西青岛研究所。

**附录**

**A. 对不同骨干网的有效性**

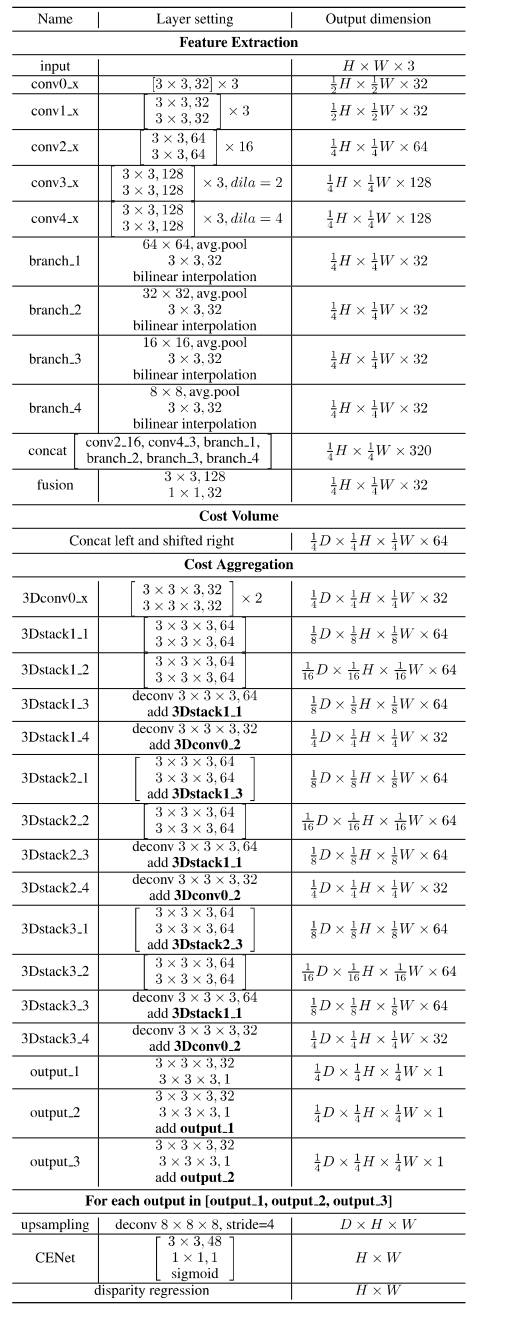
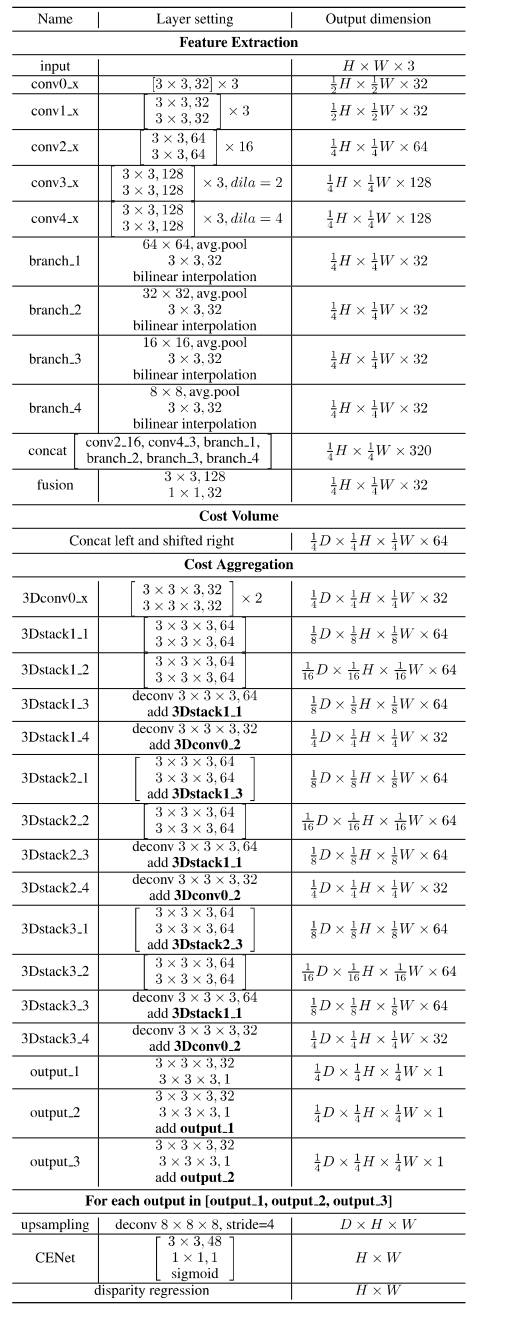
我们评估了不同骨干网之间的自适应单峰成本量过滤方案的有效性，即PSMNet的沙漏版本（Chang and Chen 2018）和GC-Net（Kendall 等2017）。我们使用“**实施详细信息**”中详细介绍的培训协议来重新实施所

有方法。具体而言，GC-Net的批量大小设置为24，以便在8 Tesla V10上进行训练。表5报告了结果，我们的方法在不同的主干网之间都提供了更好的性能。

表格 5 在不同的立体声匹配模型之间评估我们的方法其中\*表示为模型配备了自适应单峰成本量过滤方案

**B. 建筑细节**

表6列出了AcfNet的详细信息，该AcfNet用于实验以在Scene Flow数据集（Mayer等人，2016）和KITTI基准（Geiger，Lenz和Urtasun 2012； Menze和Geiger）上产生最先进的准确性2015）。它基于具有堆叠沙漏架构的PSMNet，可产生三个成本量，并向每个成本量添加Confidence Estimation网络（CENet）。

表格 6 AcfNet的网络体系结构参数。

**参考文献**

Chang, J.-R., and Chen, Y-S. 2018. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition^* 5410-5418.

Cheng, X.; Wang, P.; and Yang, R. 2018. Learning depth with convolution^ spatial propagation network. arXiv preprint arXiv:1810.02695.

Geiger, A.; Lenz, R; and Urtasun, R. 2012. Are we ready for au­tonomous driving? the kitti vision benchmark suite. In 2072 IEEE Conference on Computer Vision and Pattern Recognition, 3354— 3361. IEEE.

Guo, X.; Yang, K.; Yang, W.; Wang, X.; and Li, H. 2019. Group- wise correlation stereo network. In Proceedings of the IEEE Con­ference on Computer Vision and Pattern Recognition, 3273-3282.

Ilg, E.; Cicek, O.; Galesso, S.; Klein, A.; Makansi, O.; Hutter, F.; and Brox, T. 2018. Uncertainty estimates and multi-hypotheses networks for optical flow. In Proceedings of the European Confer­ence on Computer Vision (ECCV), 652-667.

Kendall, A., and Gal, Y 2017. What uncertainties do we need in bayesian deep learning for computer vision? In Advances in Neural Information Processing Systems, 5574-5584.

Kendall, A.; Martirosyan, H.; Dasgupta, S.; Henry, R; Kennedy, R.; Bachrach, A.; and Bry, A. 2017. End-to-end learning of geometry and context for deep stereo regression. In Proceedings of the IEEE International Conference on Computer Vision, 66-75.

Kim, S.; Min, D.; Kim, S.; and Sohn, K. 2018. Unified confidence estimation networks for robust stereo matching. IEEE Transactions on Image Processing 28(3):1299-1313.

Liang, Z.; Feng, Y; Guo, Y; Liu, H.; Chen, W.; Qiao, L.; Zhou, L.; and Zhang, J. 2018. Learning for disparity estimation through feature constancy. In Proceedings of the IEEE Conference on Com­puter Vision and Pattern Recognition, 2811-2820.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollar, P. 2017. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, 2980-2988.

Luo, C.; Yu, L.; Yang, E.; Zhou, H.; and Ren, P. 2019. A bench­mark image dataset for industrial tools. Pattern Recognition Letters 125:341-348.

Luo, W.; Schwing, A. G.; and Urtasun, R. 2016. Efficient deep learning for stereo matching. In Proceedings of the IEEE Coher­ence on Computer Vision and Pattern Recognition, 5695-5703.

Luo, C.; Yu, L.; and Ren, P 2017. A vision-aided approach to perching a bioinspired unmanned aerial vehicle. IEEE Transac­tions on Industrial Electronics 65(5):3976-3984.

Mayer, N.; Ilg, E.; Hausser, P; Fischer, R; Cremers, D.; Dosovit- skiy, A.; and Brox, T. 2016. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In Proceedings of the IEEE Conference on Computer Vision and Pat­tern Recognition, 4040-4048.

Menze, M., and Geiger, A. 2015. Object scene flow for au­tonomous vehicles. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3061-3070.

Pang, J.; Sun, W.; Ren, J. S.; Yang, C.; and Yan, Q. 2017. Cas­cade residual learning: A two-stage convolutional neural network for stereo matching, hi Proceedings of the IEEE International Con­ference on Computer Vision, 887-895.

Park, M.-G., and Yoon, K.-J. 2018. Learning and selecting confi­dence measures for robust stereo matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 41(6):1397-1411. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; De Wo, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Auto­matic differentiation in pytorch.

Poggi, M.; Pallotti, D.; Tosi, F.; and Mattoccia, S. 2019. Guided stereo matching. In Proceedings of the IEEE Conference on Com­puter Vision and Pattern Recognition^ 979-988.

Scharstein, D., and Szeliski, R. 2002. A taxonomy and evalua­tion of dense two-frame stereo correspondence algorithms. Inter­national Journal of Computer Vision 47(1-3):7-42.

Schmid, K.; Tomic, T.; Ruess, F.; Hirschmiiller, H.; and Suppa, M. 2013. Stereo vision based indooi/outdoor navigation for flying robots. In 2013IEEE/RSJ International Conference on Intelligent Robots and Systems, 3955-3962. IEEE.

Sivaraman, S., and Trivedi, M. M. 2013. A review of recent de­velopments in vision-based vehicle detection. In 2013 IEEE Intel­ligent Vehicles Symposium (IV), 310-315. IEEE.

Song, X.; Zhao, X.; Hu, H.; and Fang, L. 2018. Edgestereo: A context integrated residual pyramid network for stereo matching. In Asian Conference on Computer Vision, 20-35. Springer.

Tulyakov, S.; Ivanov, A.; and Fleuret, F. 2018. Practical deep stereo (pds): Toward applications-friendly deep stereo matching. In Ad­vances in Neural Information Processing Systems, 5871-5881.

Yang, G.; Zhao, H.; Shi, J.; Deng, Z.; and Jia, J. 2018. Segstereo: Exploiting semantic information for disparity estimation. In Pro­ceedings of the European Conference on Computer Vision (ECCV), 636-651.

Yin, Z.; Darrell, T.; and Yu, F. 2019. Hierarchical discrete distribu­tion decomposition for match density estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 6044-6053.

Zbontar, J., and LeCun, Y 2016. Stereo matching by training a convolutional neural network to compare image patches. Journal of Machine Learning Research 17(1-32):2.

Zenati, N., and Zerhouni, N. 2007. Dense stereo matching with application to augmented reality. In 2007 IEEE International Con­ference on Signal Processing and Communications, 1503—1506. IEEE.

Zhang, F.; Prisacariu, V; Yang, R.; and Torr, P. H. 2019. Ga-net: Guided aggregation net for end-to-end stereo matching. In Pro­ceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 185-194.

Adaptive Unimodal Cost Volume Filtering for Deep Stereo Matching

Ybumin Zhang1[[1]](#footnote-1) Yimin Chen1\* Xiao Bai1^ Suihanjin Yu1 Kim Yu2 Zhiwei Li2 Kuiyuan Yang2

1 State Key Laboratory of Software Development Environment, School of Computer Science and Engineering,  
Beijing Advanced Innovation Center for Big Data and Brain Computing,  
Jiangxi Research Institute, Beihang University, Beijing, China 2DeepMotion

(youmi, minwellcym, baixiao, fakecoderemail} @buaa.edu.cn {kunyu, zhiweili, kuiyuanyang} @deepmotion.ai

**Abstract**

State-of-the-art deep learning based stereo matching ap­proaches treat disparity estimation as a regression problem, where loss function is directly defined on true disparities and their estimated ones. However, disparity is just a byprod­uct of a matching process modeled by cost volume, while indirectly learning cost volume driven by disparity regres­sion is prone to overfitting since the cost volume is under constrained. In this paper, we propose to directly add con­straints to the cost volume by filtering cost volume with uni- modal distribution peaked at true disparities. In addition, vari­ances of the unimodal distributions for each pixel are esti­mated to explicitly model matching uncertainty under dif1 ferent contexts. The proposed architecture achieves state-of- the-art performance on Scene Flow and two KITTI stereo benchmarks. In particular, our method ranked the *1st* place of KTITI 2012 evaluation and the 4th place of KITTI 2015 evaluation (recorded on 2019.8.20). *The* codes of AcfNet are available at: <https://github.com/DeepMotionAIResearch/> DenseMatchingBenchmark.

Introduction

Stereo matching is one of the core technologies in com­puter vision, which recovers 3D structures of real world from 2D images. It has been widely used in areas such as autonomous driving (Sivaraman and Trivedi 2013), aug­mented reality (Zenati and Zerhouni 2007) and robotics navigation (Schmid et al. 2013; Luo, Yu, and Ren 2017; Luo et al. 2019). Given a pair of rectified stereo images, the goal of stereo matching is to compute the disparity d for each pixel in the reference image (usually refers to the left image), where disparity is defined as the horizontal displace­ment between a pair of corresponding pixels in the left and right images.

According to the seminar work (Scharstein and Szeliski 2002), a stereo matching algorithm typically consists of four steps: matching cost computation,cost aggregation, dis­parity regression and disparity refinement. Among them, matching cost computation, i.e., obtaining cost volume, is

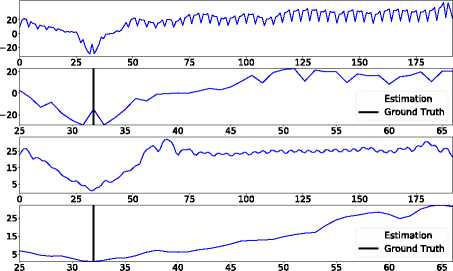
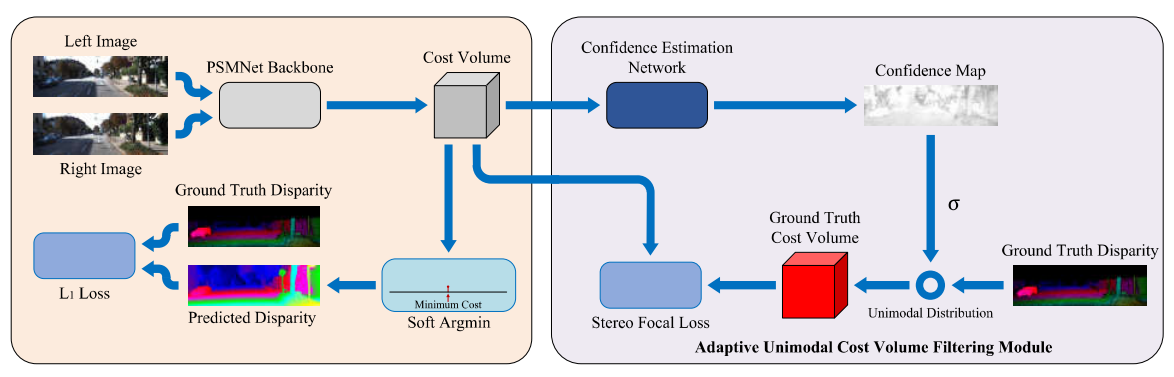


Figure 1: An example of cost distribution along the dispar­ity dimension of cost volume. The first row is the output of PSMNet (Chang and Chen 2018) trained with soft argmin. The third row is output of our model. For better visualiza­tion, we also zoom into the disparity interval [25,66] in the second and fourth row, where Estimation and Ground Truth are the estimated and the groundtruth disparity respectively. Our method generates a more reasonable cost distribution peaked at the true disparity.

arguably the most crucial first step. A cost volume is usually denoted by a H x PT x Z> tensor, where *H, W9 D* are the height, width and maximal disparity of the reference image. In traditional methods, cost volume is computed by a pre­defined cost function of manually designed image features, e.g., squared or absolute difference of image patches.

In the deep learning era, both image feature and cost func­tion are modeled as network layers (Kendall et al. 2017; Chang and Chen 2018). To make all layers differentiable and achieve sub-pixel estimation of disparity, *soft argmin* is used to estimate disparity by softly weighting indices according to their costs, which is in contrast to *argmin* that takes the in­dex with minimal cost as estimated disparity. The loss func­tion is defined on the estimated disparity and the ground truth for end-to-end training. Benefit from large-scale train­ing data and end-to-end training, deep learning based stereo approaches achieve state-of-the-art performance.

In the deep learning models, the cost volume is indi-

rectly supervised as an intermediate layer, which leaves cost volume less constrained since infinitely many cost distri­butions can generate the same disparity, where only cost distributions peaked at the true disparity are reasonable ones. Accordingly, we propose to directly supervise cost volume with unimodal ground truth distributions. To re­veal network matching uncertainties (Kendall and Gal 2017; Ilg et al. 2018) of different pixels, we design a confidence es­timation network to estimate per-pixel confidence and con­trol sharpness of the unimodal ground truth distributions ac­cordingly. Figure 1 compares the cost distributions at the same pixel by PSMNet and our method, where our method generates the correct minimal cost around the true disparity, while PSMNet generates two local minimal costs away from the true disparity.

**a**

Figure 2: Architecture of the proposed end-to-end AcfNet. The input stereo images are fed to PSMNet (Chang and Chen 2018) backbone with stacked hourglass architecture to get three cost volumes after aggregation. For each cost volume, we generate the confidence map by a Confidence Estimation Network (CENet), and modulate the ground truth cost volume with confidence values to generate pixel-wise unimodal distribution as training labels. The proposed *Stereo Focal Loss* is added to the cost volume using the generated training labels. Finally, a sub-pixel disparity map is estimated by the *soft argmin* function followed by regression loss as PSMNet.

We evaluate the proposed Adaptive unimodal cost volume filtering Network (AcfNet) on three stereo benchmarks in­cluding Scene Flow, KITTI2012 and KITTI2015. Ablation studies and detailed analysis on Scene Flow demonstrate the effectiveness of AcfNet. We also submit our stereo match­ing results to KITTI 2012 and 2015 evaluation server, and ranked the 1" place on KITTI 2012 evaluation and the 4\* place on KITTI 2015 evaluation (recorded on 2019.8.20).

Related Work

Deep learning for stereo matching starts from learning im­age features for classical methods (Zbontar and LeCun 2016; Luo, Schwing, and Urtasun 2016). DispNetC (Mayer et al. 2016) is the first breakthrough fbr stereo matching by proposing an end-to-end trainable network, where cost function is predefined as a correlation layer in the network to generate the cost volume, then a set of convolutional layers are added to the cost volume to regress disparity map. Based on DispNetC, stack refinement sub-networks are proposed to improve the performance (Pang et al. 2017; Liang et al. 2018), and the performance could be further im­proved by using additional information such edges (Song et al. 2018) and semantics (Yang et al. 2018). To add more capacity for network to learn the cost function, (Guo et al. 2019) propose to use group-wise correlation layer and gen­erate multiple cost volumes for latter aggregation.

GC-Net (Kendall et al. 2017) gives more flexibility for network to learn cost function by using 3D convolutional layers on concatenated feature volume, with cost volume produced by the learned cost function, disparity is esti­mated by *soft argmin* according to the cost distribution. Follow-up works improve results by using better image fea­tures (Chang and Chen 2018) and cost aggregation layers in­spired by classical methods (Cheng, Wang, and Yang 2018; Zhang et al. 2019). In these end-to-end stereo matching net­works, cost volume is the output of an intemiediate layer without direct supervision, which leaves the possibilities to learn unreasonable cost distributions as illustrated in Fig­ure 1.

In this work, the proposed AcfNet directly adds super­vision to the cost volume estimation using ground truth cost distributions peaked at true disparities. In addition, the sharpness of ground truth cost distribution is adjusted ac­cording to matching confidence. Concurrent to our work, sparse LiDAR points are used to enhance cost volume by weighting estimated cost distribution with a Gaussian distri­bution centered at the disparity provided by its correspond­ing LiDAR point (Poggi et al. 2019), which serves as a multi-sensor fusion method for disparity estimation. In con­trast, our method only takes images as input during both training and testing, and unimodal supervision is added to each pixel in a dense and adaptive way.

AcfNet

Figure 2 illustrates the overall framework, where the pro­posed adaptive unimodal cost volume filtering module is ap­plied to the cost volume, and an additional loss is introduced to directly supervise the learning of cost volume towards de­sired property. Here, we choose PSMNet (Chang and Chen 2018) as the basic network to calculate cost volume for its state-of-the-art performance on stereo matching.

**Overview**

Given a pair of rectified images, for each pixel *p =* in the left image, stereo matching aims to find its correspond­ing pixel in the right image, i.e.,, where disparity *d* is often represented by a floating-point number for sub-pixel matching. For both computation and memory tractable, disparity is discreted into a set of possi­ble disparities, i.e., to build cost volume, where *H, W* and *D* are the image height, width and maximum disparity respectively. To recover sub-pixel matching, costs over disparities are used in a weighted inter­polation. The whole process is implemented through a net­work as illustrated in the left part of Figure 2.

Formally, the cost volume contains *D* costs for each pixel denoted by , and the sub-pixel disparity is estimated through *soft argmin* (Kendall et d. 2017)

(1)

where, and disparities with

small cost contribute more during interpolation. Given the groundtruth disparity *dp* for each pixel , smoothloss is defined for training, i.e.,

(2)

where

⑶

The whole process is differentiable by supervising with the groundtruth disparity, while cost volume is indirectly su­pervised through providing weights for disparity interpola­tion. However, the supervision is underdetermined and there could be infinitely possible sets of weights to achieve correct interpolation results. The flexibility of cost volume is prone to overfitting since many improperly learned cost volumes could interpolate disparities close to ground truth (i.e., small training loss).

To address this problem raised from indirectly supervising cost volume with underdetermined loss function, we pro­pose to directly supervise the cost volume according to its unimodal property.

**Unimodal distribution**

Cost volume is defined to reflect the similarities between candidate matching pixel pairs, where the true matched pair should have the lowest cost (i.e., the highest similarity), and the costs should increase with the distance to the truly matched pixel. This property requires unimodal distribution be peaked at the true disparity at each position in the cost volume. Given the ground truth disparity *d9t,* the unimodal distribution is defined as

|  |  |
| --- | --- |
|  | (4) |
|  |

where is the variance (a.k.a temper­ature in literature) that controls the sharpness of the peak around the true disparity.

The ground truth cost volume constructed from *P(d)* has the same sharpness of peaks across different pixels, which cannot reflect similarity distribution differences across dif­ferent pixels. For example, a pixel on the table comer should have a very sharp peak while pixels in uniform regions should have relative flat peaks, lb build such more reason­able labels for cost volume, we add a confidence estimation network to adaptively predict *ap* for each pixel.

**Confidence estimation network**

Considering matching properties are embedded in the esti­mated cost volume (Fu and Fard 2018; Park and Yoon 2018; Kim et al. 2018), then the confidence estimation network takes the estimated cost volume as input, and uses a few layers to determine the matching confidence of each pixel by checking the matching states in a small neighborhood around each pixel. Specifically, the network employs a3 x 3 convolutional layer followed by batch nonndization and ReLU activation, and another lxl convolutional layer fol­lowed by sigmoid activation to produce a confidence map *f* € [0, where a pixel *p* with large confidence *fp* means a unique matching can be confidently found for this pixel, while small confidence values denote there are match­ing ambiguities. Then, *ap* for generating ground truth cost distribution is scaled from the estimated confidence,

(5)

where s > 0 is a scale factor that reflects the sensitivity of *a* to the change of confidence *fp,e>Q* defines the lower bound for *a* and avoids numerical issue of dividing 0. Ac­cordingly, . Our experiments show that two kinds of pixels are likely to have large *a,* i.e., texture-less pixels and occluded pixels, where the texture-less pixels tend to have multiple matches, while occluded pixels have no cor­rect matches. With the per-pixel adpatively estimated *api* the ground truth cost volume defined in Eq. (4) is modified ac­cordingly.

**Stereo focal loss**

At pixel position p, we now have both estimated cost dis­tribution *Pp(d)* and the ground truth *Pp{d).* It is straightfor­ward to define a distribution loss via cross entropy. However, there is a severe sample imbalance problem since each pixel has only one true disparity (positive) comparing with hun­dreds of negative ones (Zbontar and LeCun 2016). Similar to focal loss designed to solve the sample imbalance problem in one-stage object detection (Lin et al. 2017), we design a stereo focal loss to focus on positive disparities to avoid the total loss dominated by negative disparities,

is *^focusing* parameter, and the loss is reduced to cross entropy loss when a = 0, while a > 0 gives more weights to positive disparities in proportion to their *Pp(d),* Thus easy negative disparities are further suppressed explic­itly with quite small weights and let the positive disparity only compete with a few hard ones.

Total loss function

In sum, our final loss function contains three parts defined

(7)

as

|  |
| --- |
|  |

where Me two trade-off hyper- parameters. *Csf* supervises the cost volume while supervises the disparity, *^confidence* is added as a regularizer to encourage more pixels to have high confi­dence values,

(8)

**Experiments and Analysis Implementation details**

Our network is implemented using PyTorch (Paszke et al. 2017) framework, and all models are end-to-end trained us­ing RMSprop with standard settings. Our data processing is the same as PSMNet (Chang and Chen 2018). We train our models from scratch using the Scene Flow dataset with a constant learning rate of 0.001 for 10 epochs. For Scene Flow, the trained model is directly used for testing. For KITTI, we use the model trained with Scene Flow data after fine-tuning on the KITTI training set for 600 epochs. The learning rate of this fine-tuning begins at 0.001 and is de­cayed by I at 100 and 300 epochs. For submission to the KITTI test benchmark, we prolong the training process on Scene Flow with a constant learning rate of 0.001 for 20 epochs to obtain a better pre-training model. The batch size is set to 3 for training on 3 NVIDIA GTX 1080Ti GPUs. All ground truth disparities out of range of are excluded in our experiments, where *D* = 192.

Datasets

We evaluate AcfNet qualitatively and quantitatively on three challenging stereo benchmarks, i.e., Scene Flow (Mayer et al. 2016), KITTI 2012 (Geiger, Lenz, and Urtasun 2012) and KTTTI2015 (Menze and Geiger 2015).

**Scene Flow:** Scene Flow is a large synthetic dataset con­taining 35,454 training image pairs and 4,370 testing image pairs, where the ground truth disparity maps are densely pro­vided, which is large enough for directly training deep learn­ing models. Following the setting as GC-Net (Kendall et al. 2017), we mainly use this dataset for ablation study.

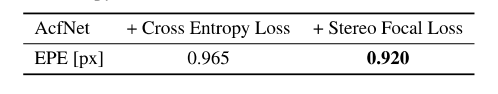
**KITTI:** KITTI 2015 and KITTI 2012 are two real-world datasets with street views captured from a driving car. KTTTI 2015 contains 200 training stereo image pairs with sparse groundtruth disparities obtained using LiDAR and 200 test­ing image pairs with ground truth disparities held by eval­uation server for submission evaluation only. KITTI 2012 contains 194 training image pairs with sparse ground truth disparities and 195 testing image pairs witii ground truth dis­parities held by evaluation server for submission evaluation only. These two datasets are challenging due their small size. **Metrics:** The performance is measured using two standard metrics: (1) 3-Pixel-Error (3PE), i.e., the percentage of pix­els for which the predicted disparity is off the true one by more than 3 pixels, and (2) End-Point-Error (EPE), i.e., the average difference of the predicted disparities and their true ones. 3PE is robust to outliers with large disparity errors, while EPE measures errors to sub-pixel level.

To further evaluate the ability on handling occluded re­gions, we divide the testing images of Scene Flow into occluded region (OCC) and non-occluded regions (NOC) through left-right consistency check. In total, tiiere are 16% occluded pixels in all pixels. The performance is measured on all pixels if no prefix such as OCC, NOC and ALL are added before 3PE or EPE.

Ablation studies

We conduct ablation studies on Scene Flow (Mayer et al. 2016) considering it has large enough training data for end- to-end training from scratch. In all experiments, *a* is set to 5.0 in stereo focal loss to balance positive and negative sam­ples. Considering disparities of most pixels are with sub­pixel errors (i.e., error smaller than one pixel) while 3PE cannot reveal errors within 3 pixels, we use EPE to study the performance variance for (Afferent hyper-parameter set­tings.

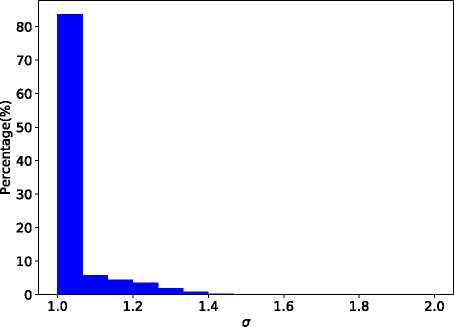
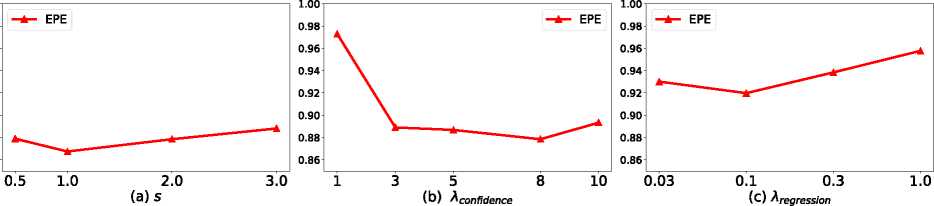
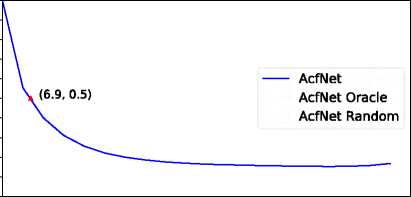
Table 1: Results of comparison between stereo focal loss and cross entropy loss in our model AcfNet.



**The variance** *a* **of unimodal distribution**

The variance *a* adjusts the shape of unimodal distribution, which plays an important role in AcfNet. In our method, *a* e [c, s + e] is bounded by *s* and e.

Firstly, we study the case when the variance *a* is fixed for all pixels, i.e. s = 0, a = e. By grid search, we find that *a* = 1.2 achieves the best result, which indicates most pixels favor *a* = 1.2 for building unimodal distributions. Thus, we set the lower bound c of cr to 1.0 for adaptive vari­ance study. Furthennore, we compare the stereo focal loss with cross entropy loss under this condition, i.e. *a* = 1.2. As shown in Table 1, equipping

Secondly, we study the sensitivity *s* which controls the upper bound of *a.* Figure 3(a) shows the performance by varying s, where s = 1 performs best and the performance is rather stable by varying *s* from 0.5 to 3.0. Figure 4 shows the histogram of *a* when *s = 1* (i.e., *a* e [1.0,2.0]), where most pixels favor small variances, i.e., sharp distributions, and a long tail of pixels require larger variances for flatten distributions.

1.00

0.98

0.96

0.94

0.92

0.90

0.88

0.86

Figure 3: Ablation study results for different hyper-parameters in our method, where *s* controls the upper bound of variance *a. ^confidence* and *Xregression* are balance weights for confidence loss and disparity regression loss respectively.

Figure 4: Histogram distribution of vanance *a* on the whole test dataset of Scene Flow after AcfNet has been converged.

**0987654321**

~pa,z\_-",lu」ON) wdw

10 20 30 40 50 60 70 80 90 100

Fraction of Removed Pixels (%)

Figure 5: Sparsincatioii plot of our AciNet on the Scene Flow test dataset. The plot shows the normalized average end-point-error (EPE) for each fraction of pixels with high­est variances has been removed. The curve 'AcfNet Oracle, shows the ideal case by removing each fraction of pixels ranked by the ground truth EPE. The curve 'AcfNet Ran­dom5 shows the worst case by removing each fraction of pix­els randomly. Removing only 6.9% of the pixels by AcfNet results in halving the average EPE.

**Loss balance weights**

Hyperparameter *XConfidence* balances the total variance and other losses. Figure. 3(b) shows the performance curve by varying *Xconfidence^* where both overconfident learn­ing with large *Aconfidence* and underconfident learning with small *Xconfidence* lead to inferior performance while *^confidence* = 8.0 performs the best.

Hyperparameter *Xregression* balances the regression loss that is widely used in recent state-of-the-art models, and large value for *Xregression* will eliminate effects of the other two losses proposed in this paper. Figure 3(c) shows the per­formance curve, it could be observed that regression loss can be improved through proper tradeoff with the proposed two

Variance analysis

A^riance estimation is an important component of our cost filtering scheme, which automatically adjusts the flatness of the unimodal distribution according to the matching uncer­tainty. To assess the quality of the estimated variances, spar- sification plot (Ilg et al. 2018) is adopted to reveal the rele­vance of the estimated variances with the true errors through plotting evaluation results by gradually removing pixels ac­cording their variances. For comparison, we also plot the curves of randomly assigned variances (AcfNet Random) and variances assigned by EPE errors (AcfNet Oracle) in Figure 5, where the estimated variances are highly relevant to EPE errors and demonstrates the ability of AcfNet in ex­plaining outlier pixels with estimated variances.

Figure 6 shows several per-pixel results from Scene Flow, where hard regions mainly appear at occlusions (la, 1c and 2a), repeated patterns (lb, 3a) and thin structures (3a). In these hard regions, AcfNet provides high variances to flat­ten the corresponding cost distributions. AcfNet can balance the learning for different pixels, and pushes informative pix­els towards high confidences (i.e, low variances), while al-

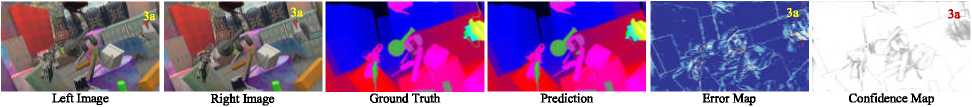
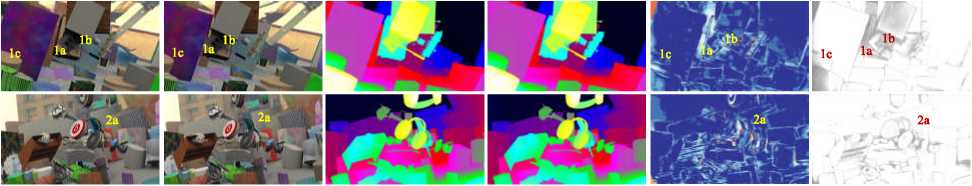
lows hard uninformative pixels with high variances to avoid overfitting.

Figure 6: Qualitative results on three samples from Scene Flow test set. Columns from left to right are: left stereo input image, right stereo input image, disparity ground truth, disparity prediction, error map and confidence map. Cold colors in the error map denote small prediction errors while warm colors denote large prediction errors. In confidence map, bright colors mean small variances while dark colors denote high variances.

Table 2: Evaluation of adaptive unimodal cost vol­ume filtering results, where PSMNet is re-implemented. AcfNet(uniform) denotes setting a uniform unimodal dis­tribution for all pixels, and AcfNet(adaptive) denotes adap­tively adjust the per-pixel variances.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Method | Scene Flow | | | | | |
| EPE [px] | | | 3PE [%] | | |
| ALL | OCC | NOC | ALL | OCC | NOC |
| PSMNet | 1.101 | 3.507 | 0.637 | 4.56 | 17.64 | 2.12 |
| AcfNet (unifonn) | 0.920 | 2.996 | 0.504 | 4.39 | 16.47 | 2.10 |
| AcfNet (adaptive) | 0.867 | 2.736 | 0.495 | 4.31 | 15.77 | 2.13 |

Adaptive unimodal cost volume filtering

AcfNet adds direct cost volume supervision to PSMNet. Table 2 compares two versions of AcfNet with PSM­Net, where uniform version of AcfNet is significantly bet­ter than PSMNet and adaptive version of AcfNet fur­ther improves the performance significantly. The results demonstrate the effectiveness of unimodal supervision and adaptive per-pixel variance estimation. Comparing with AcfNet(uniform), AcfNet(adaptive) improves more on OCC (i.e., occluded regions), which is consistent with conclusion in variance analysis.

Cost volume filtering comparisons

To further validate the superiority of the proposed cost vol­ume filtering, experiments are designed to compare with the concurrent work (Poggi et al, 2019). In contrast to our work, (Poggi et al. 2019) uses disparities by sparse LiDAR points to filter cost volume during both training and test­ing. Both AcfNet and the method of (Poggi et al. 2019) are trained on Scene Flow from scratch, and directly evaluated on training sets of KITTI 2012 and 2015 since (Poggi et

Table 3: Results of cost volume filtering comparison, where all methods are trained on Scene Flow from scratch using the same base model PSMNet, and directly test on KITTI 2012, 2015 training datasets. \* denotes disparities of sparse LiDAR points are also used as model input when testing.

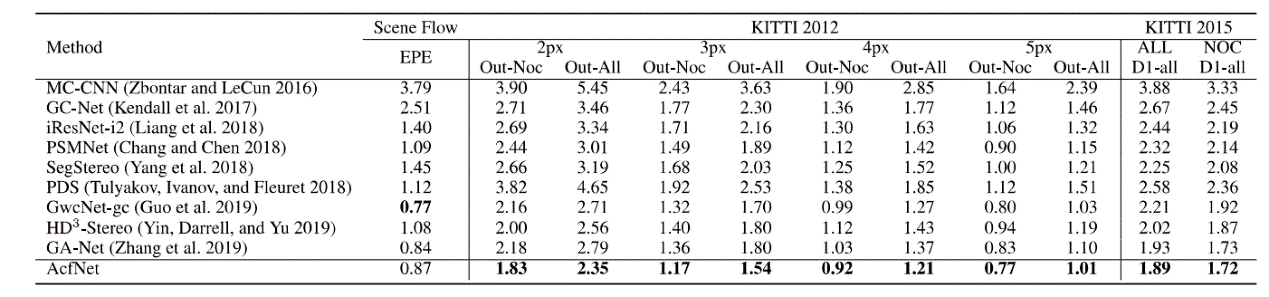
|  |  |  |  |
| --- | --- | --- | --- |
| Method | EPE[px] | 3PE[%] | |
| Scene Flow | KITTI 2012 | Kirn 2015 |
| PSMNet  (Poggi et al. 2019)  AcfNet | 1.101  0.991\*  **0.867** | 29.18  **17.54** | 30.19  23.13\*  **19.45** |

al. 2019) requires sparse LiDAR points as inputs. Table 3 reports the comparison results, where AcfNet outperforms (Poggi et al. 2019) on all performance metrics by large mar­gins even without using LiDAR points as inputs. In addition, comparing with PSMNet, AcfNet shows much better gen­eralization performance from Scene Flow to KITTI, which further proves the ability of AcfNet in preventing overfitting.

Comparisons with the state-of-the-art methods

To further validate the proposed AcfNet, Table 4 compares AcfNet with state-of-the-art methods on both KITTI 2012 and 2015, where AcfNet outperforms others by notable mar­gins on all evaluation metrics. To be noted, Scene Flow is used for pretraining in all methods considering the small size of KITTI training data. Figure 7 and 8 show sev­eral exemplar results from KITTI 2015 and 2012 by com­paring AcfNet with PSMNet (Chang and Chen 2018) and PDS (Tulyakov, Ivanov, and Fleuret 2018), where signifi­cantly improved regions are marked out with dash boxes. As expected, most improvements of AcfNet come from chal­lenging areas such as thin structures, sky boundaries and im­age borders.

Table 4: Results on Scene Flow and KITTI Benchmarks. Following standard setting, on KITTI 2012, percentages of erroneous pixels for both Non-occluded (Out-Noc) and all (Out-All) pixels are reported, on KITTI 2015, percentages of disparity outliers *Di* averaged over all ground truth pixels (DI-all) for both Non-occluded and All pixels are reported. The outliers are defined as those pixels whose disparity errors are larger than max(3px, 0.05#'), where *d9t* is the ground-truth disparity.



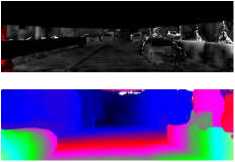
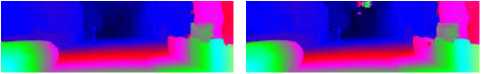
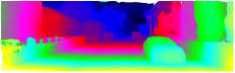
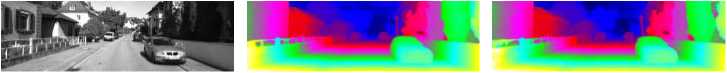
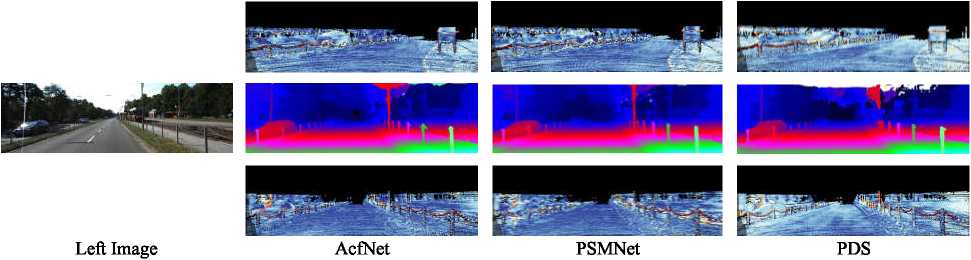
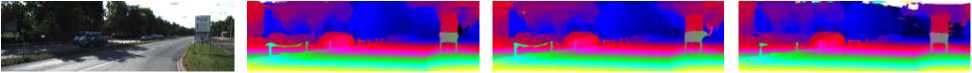


Figure 7: Visualization results on the KITTI2015 dataset. Significantly improved regions are highlighted with dash boxes. For each example, the first row shows the disparity map, and the second row shows the error map. Warmer color indicate larger prediction errors.

Left Image

PSMNet

Figure 8: Visualization results on the KITTI 2012 dataset. Significantly improved regions are highlighted with dash boxes. For each example, the first row shows the disparity map, and the second row shows the error map, bright colors indicate inaccurate predictions.

Conclusions

In this paper, we solve the under-constrain problem of cost volume in existing deep learning based stereo matching ap­proaches. The proposed AcfNet supervises the cost volume with ground truth unimodal distributions peaked at true dis­parities, and variances for per-pixel distributions are adap­tively estimated to modulate the learning according the in­formativeness of each pixel. AcfNet shows better testing performance on the same dataset and even superior perfor­mance on cross-dataset evaluation.

Acknowledgements

This work was supported by the National Natural Science Foundation of China project no. 61772057 and the support funding from State Key Lab. of Software Development Envi­ronment and Jiangxi and Qingdao Research Institute of Bei- hang University.

Appendices

**A. Effectiveness on different backbones**

We evaluate the effectiveness of our adaptive unimodal cost volume filtering scheme among different backbones, namely, the stack-hourglass version of PSMNet (Chang and Chen 2018) and GC-Net (Kendall et al. 2017). We re­implement all methods with the training protocol detailed in **Implementation details.** Specifically, the batch size of GC-Net is set to 24 for training on 8 Tesla V100. Table 5 reports the results, our method delvers better performance across different backbones.

Table 5: Evaluation our method among different stereo matching models, where \* denotes equipping the model with our adaptive unimodal cost volume filtering scheme.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Method** | **Scene Flow** | | | | | |
| EPE [px] | | | 3PE [%] | | |
| ALL | occ | **NOC** | ALL | **OCC** | **NOC** |
| **GC-Net** | 0.871 | 2.916 | 0.452 | **3.89** | **15.63** | **1.65** |
| **GC-Net\*** | **0.822** | **2.777** | **0.436** | 4.33 | 16.46 | 2.02 |
| **PSMNet** | 1.101 | 3.507 | 0.637 | 4.56 | 17.64 | **2.12** |
| **PSMNet\*** | **0.867** | **2.736** | **0.495** | **4.31** | **15.77** | 2.13 |

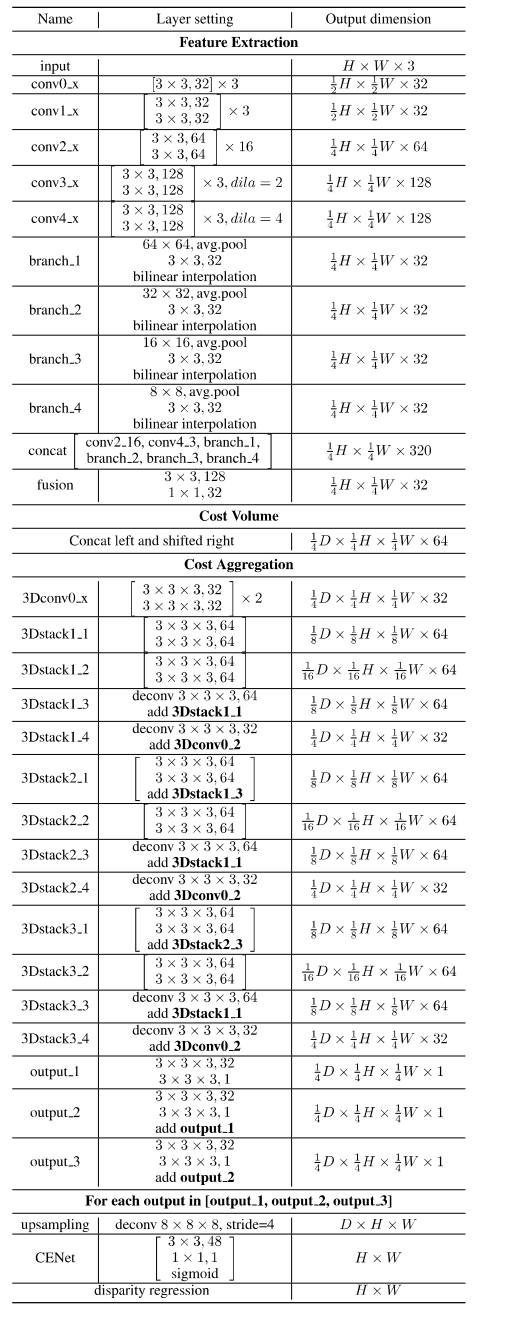
B. Architecture details

Table 6 presents the details of the AcfNet which is used in experiments to produce state-of-the-art accuracy on Scene Flow dataset (Mayer et al. 2016) and KITTI bench­marks (Geiger, Lenz, and Urtasun 2012; Menze and Geiger 2015). It is based on PSMNet with stacked hourglass archi­tecture, which produces three cost volumes, and Confidence Estimation network(CENet) is added to each of the cost vol­ume.

References

Chang, J.-R., and Chen, Y-S. 2018. Pyramid stereo matching network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition^* 5410-5418.

Table 6: Parameters of the network architecture of AcfNet.



Cheng, X.; Wang, P.; and Yang, R. 2018. Learning depth with convolution^ spatial propagation network. *arXiv preprint arXiv:1810.02695.*

Fu, Z., and Fard, M. A. 2018. Learning confidence measures by multi-modal convolutional neural networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)9* 1321- 1330. IEEE.

Geiger, A.; Lenz, R; and Urtasun, R. 2012. Are we ready for au­tonomous driving? the kitti vision benchmark suite. In 2072 *IEEE Conference on Computer Vision and Pattern Recognition,* 3354— 3361. IEEE.

Guo, X.; Yang, K.; Yang, W.; Wang, X.; and Li, H. 2019. Group- wise correlation stereo network. In *Proceedings of the IEEE Con­ference on Computer Vision and Pattern Recognition,* 3273-3282.

Ilg, E.; Cicek, O.; Galesso, S.; Klein, A.; Makansi, O.; Hutter, F.; and Brox, T. 2018. Uncertainty estimates and multi-hypotheses networks for optical flow. In *Proceedings of the European Confer­ence on Computer Vision (ECCV),* 652-667.

Kendall, A., and Gal, Y 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems,* 5574-5584.

Kendall, A.; Martirosyan, H.; Dasgupta, S.; Henry, R; Kennedy, R.; Bachrach, A.; and Bry, A. 2017. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE International Conference on Computer Vision,* 66-75.

Kim, S.; Min, D.; Kim, S.; and Sohn, K. 2018. Unified confidence estimation networks for robust stereo matching. *IEEE Transactions on Image Processing* 28(3):1299-1313.

Liang, Z.; Feng, Y; Guo, Y; Liu, H.; Chen, W.; Qiao, L.; Zhou, L.; and Zhang, J. 2018. Learning for disparity estimation through feature constancy. In *Proceedings of the IEEE Conference on Com­puter Vision and Pattern Recognition,* 2811-2820.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollar, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision,* 2980-2988.

Luo, C.; Yu, L.; Yang, E.; Zhou, H.; and Ren, P. 2019. A bench­mark image dataset for industrial tools. *Pattern Recognition Letters* 125:341-348.

Luo, W.; Schwing, A. G.; and Urtasun, R. 2016. Efficient deep learning for stereo matching. In *Proceedings of the IEEE Coher­ence on Computer Vision and Pattern Recognition,* 5695-5703.

Luo, C.; Yu, L.; and Ren, P 2017. A vision-aided approach to perching a bioinspired unmanned aerial vehicle. *IEEE Transac­tions on Industrial Electronics* 65(5):3976-3984.

Mayer, N.; Ilg, E.; Hausser, P; Fischer, R; Cremers, D.; Dosovit- skiy, A.; and Brox, T. 2016. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pat­tern Recognition,* 4040-4048.

Menze, M., and Geiger, A. 2015. Object scene flow for au­tonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 3061-3070.

Pang, J.; Sun, W.; Ren, J. S.; Yang, C.; and Yan, Q. 2017. Cas­cade residual learning: A two-stage convolutional neural network for stereo matching, hi *Proceedings of the IEEE International Con­ference on Computer Vision,* 887-895.

Park, M.-G., and Yoon, K.-J. 2018. Learning and selecting confi­dence measures for robust stereo matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41(6):1397-1411. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; De Wo, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; and Lerer, A. 2017. Auto­matic differentiation in pytorch.

Poggi, M.; Pallotti, D.; Tosi, F.; and Mattoccia, S. 2019. Guided stereo matching. In *Proceedings of the IEEE Conference on Com­puter Vision and Pattern Recognition^* 979-988.

Scharstein, D., and Szeliski, R. 2002. A taxonomy and evalua­tion of dense two-frame stereo correspondence algorithms. *Inter­national Journal of Computer Vision* 47(1-3):7-42.

Schmid, K.; Tomic, T.; Ruess, F.; Hirschmiiller, H.; and Suppa, M. 2013. Stereo vision based indooi/outdoor navigation for flying robots. In *2013IEEE/RSJ International Conference on Intelligent Robots and Systems,* 3955-3962. IEEE.

Sivaraman, S., and Trivedi, M. M. 2013. A review of recent de­velopments in vision-based vehicle detection. In *2013 IEEE Intel­ligent Vehicles Symposium (IV),* 310-315. IEEE.

Song, X.; Zhao, X.; Hu, H.; and Fang, L. 2018. Edgestereo: A context integrated residual pyramid network for stereo matching. In *Asian Conference on Computer Vision,* 20-35. Springer.

Tulyakov, S.; Ivanov, A.; and Fleuret, F. 2018. Practical deep stereo (pds): Toward applications-friendly deep stereo matching. In *Ad­vances in Neural Information Processing Systems,* 5871-5881.

Yang, G.; Zhao, H.; Shi, J.; Deng, Z.; and Jia, J. 2018. Segstereo: Exploiting semantic information for disparity estimation. In *Pro­ceedings of the European Conference on Computer Vision (ECCV),* 636-651.

Yin, Z.; Darrell, T.; and Yu, F. 2019. Hierarchical discrete distribu­tion decomposition for match density estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 6044-6053.

Zbontar, J., and LeCun, Y 2016. Stereo matching by training a convolutional neural network to compare image patches. *Journal of Machine Learning Research* 17(1-32):2.

Zenati, N., and Zerhouni, N. 2007. Dense stereo matching with application to augmented reality. In *2007 IEEE International Con­ference on Signal Processing and Communications,* 1503—1506. IEEE.

Zhang, F.; Prisacariu, V; Yang, R.; and Torr, P. H. 2019. Ga-net: Guided aggregation net for end-to-end stereo matching. In *Pro­ceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* 185-194.

1. Joint first authorship. (youmi,minwellcym} @ buaa.edu.cn. Corresponding author, baixiao @buaa.edu.cn.

   Copyright © 2020, Association for the Advancement of Artificial Intelligence ([www.aaai.org](http://www.aaai.org)). All rights reserved. [↑](#footnote-ref-1)