分类号_____ 学号_____

学校代码 10487 密级_____

华中科技大学

硕士学位论文

基于卷积神经网络的立体匹配研究

学位申请人：

学 科 专 业： 信息与通信工程

指 导 教 师：

答 辩 日 期：

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree for the Master of Engineering

Research on stereo matching based on convolutional neural network

Candidate :

Major : Information and Communication Engineering

Supervisor :

Huazhong University of Science & Technology

Wuhan 430074, P.R.China

May, 2019

独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除文中已经标明引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在文中以明确方式标明。本人完全意识到本声明的法律结果由本人承担。

学位论文作者签名：

日期：　　年　　月　　日

学位论文版权使用授权书

本学位论文作者完全了解学校有关保留、使用学位论文的规定，即：学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权华中科技大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本论文属　保密□，　在_____年解密后适用本授权书。

　　不保密□。

（请在以上方框内打“√”）

学位论文作者签名：　　指导教师签名：

日期：　　年　月　日　　日期：　　年　月　日

摘 要

双目立体视觉在获取深度信息上有着低成本、灵活和易实现等优点，因此在机器人导航、自动驾驶和增强现实等众多前沿方向中有着广泛应用。双目视觉的核心就是通过在两幅图像中寻找匹配点得到视差，从而利用三角测量原理恢复场景深度。近来，卷积神经网络凭借其强大的特征提取和模型表达能力，在诸多的计算机视觉任务中都取得了突破性进展，基于深度学习的立体匹配研究逐渐成为目前的热点。本文基于端到端的卷积神经网络进行立体匹配研究，着力于解决立体匹配中存在的遮挡、反射和弱纹理等易发生误匹配的难点区域。

对于立体匹配中存在的难点区域，利用像素的上下文信息进行视差推断是一个重要思路。为了在立体匹配中充分利用上下文信息，本文提出了利用层次化上下文信息的端到端立体匹配算法。在特征提取过程中，设计空间金字塔池化模块在不同尺度和位置上提取全局和局部的层次化上下文信息以适应不同的匹配区域。在优化学习匹配代价阶段，设计了编码-解码结构 3D 卷积模块，通过跳跃连接聚合多尺度的上下文信息。在不需要后处理的情况下，提出的立体匹配网络通过视差回归直接输出精细化的视差。

另外，本文分析了基于深度学习的方法中存在的遮挡区域与非遮挡区域训练样本不平衡问题，针对此问题，提出了回归聚焦损失函数来进行监督训练。回归聚焦损失函数可以在训练过程中自适应地调整样本损失，平衡样本在总损失中的贡献，使模型聚焦在难估计处理的样本上，防止模型退化。

我们在 Middlebury 2014 数据集上进行了实验分析，结果表明提出的立体匹配算法能够有效提高视差估计精度，尤其是在难处理的遮挡区域。

关键词：双目视觉，立体匹配，卷积神经网络，上下文信息，损失函数

ABSTRACT

Because of the advantages of low cost, flexibility and easy implementation in acquiring depth information, binocular vision has been widely used in many promising applications such as robot navigation, autonomous driving and augmented reality. The core of binocular vision is to obtain the disparity by finding corresponding points in two images, then the depth of the scene can be calculated using the principle of triangulation. Recently, convolutional neural networks have made breakthroughs in many computer vision tasks with their powerful capabilities of feature extraction and model representation. Stereo matching based on deep learning has gradually become a hot research topic. This thesis is based on an end-to-end convolutional neural network for stereo matching research, and we focus on the ill-posed regions that are prone to be mismatching such as occlusion areas, reflective surfaces and texture-less regions.

For the ill-posed regions that existing in stereo matching, it is an important idea to use the context information of pixels to perform disparity inference. In order to make full use of context information in stereo matching, this paper proposes an end-to-end stereo matching algorithm that utilizes hierarchical context information. In the feature extraction process, we design spatial pyramid pooling module to extract hierarchical context information at different scales and locations to accommodate different matching regions. In the stage of learning optimized matching cost, 3D convolution operation of the encoder-decoder architecture is designed, and the multi-scale context information is aggregated via the skip connection. The proposed end-to-end stereo matching network directly outputs the refined disparity map without any additional post-processing.

In addition, this thesis analyzes the imbalance of training samples between occlusion and non-occlusion regions based on deep learning methods. To solve this problem, a regression focal loss function is proposed for supervised training. The regression focal loss

function can adaptively adjust the sample loss during the training process, suppress the loss of well-estimated samples, make model focus on the samples that are difficult to estimate and prevent the model from degrading.

We evaluated our algorithm on the Middlebury 2014 dataset. The results show that the proposed stereo matching algorithm can effectively improve the disparity estimation accuracy, especially in the occlusion region.

Key words: binocular vision, stereo matching, convolutional neural network, context information, loss function

目 录

1 绪论

1.1 研究背景与意义

根据在成像过程中是否主动发出能量，目前计算机视觉领域获取深度信息的方式可以分为两类，即主动深度传感与被动深度传感。主动深度传感器主要包括飞行时间（time of flight，TOF）、结构光和激光雷达，如微软公司推出的 Kinect II 传感器和用于无人驾驶汽车上的 LiDAR 传感器。TOF 相机通过测量发出的调制脉冲信号在相机与场景之间的往返时间，从而计算出传感器到场景的深度[1]。TOF 相机通常会存在成像分辨率低、传感距离有限等不足。结构光传感器将调制的结构光投射到三维场景的表面上，由于结构光的模式图案会因为物体的形状发生形变，通过形变程度利用三角原理就可以计算得到场景中各点的深度信息。结构光传感器易受到外部光源的干扰，通常仅限在室内使用[2][3]，而激光雷达则由于高昂的价格限制了其普及度。被动深度传感中最常用的方法就是双目立体视觉。类似人类视觉系统的左右眼，双目立体视觉技术使用两台相机在不同位置对同一场景进行拍摄，从而物理世界的同一目标点在不同相机平面上成像，通过寻找两幅图像上的匹配点形成视差，最后利用三角测量原理将视差转化为深度。

相对于主动深度传感器，双目立体视觉有着低成本、灵活、易扩展和易实现的优势。因而双目立体视觉在诸多领域被广泛研究和使用，如自动驾驶[4]、机器人导航[5]、虚拟视点合成[6]和增强现实[7]等，存在巨大的实用价值和商业价值。

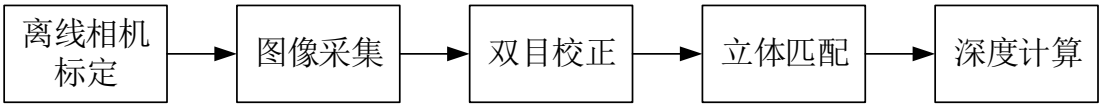


图 1.1 双目立体视觉系统。

如图 1.1 所示，双目立体视觉系统通常包含以下流程：离线相机标定、图像采集与校正、立体匹配和深度计算。上述五个步骤中最关键以及难度最大的环节就是

立体匹配，立体匹配的好坏直接决定了最终的深度图质量。

立体匹配技术主要是在校正后的左右视点图像对上寻找匹配点，然后根据匹配点在左右视点上的水平坐标差计算得到视差。寻找匹配点对的一个潜在假设就是同一表面在左右视点成像是相似的，而在实际成像过程中，一方面会存在如光照变化等成像噪声，左右视点图像存在遮挡区域、反射区域和透视畸变区域，造成左右视点成像的差异性，使得左视点像素在右视点图像中很难找到对应的匹配点。另一方面，左右视点图像存在弱纹理区域和纹理重复等区域，造成左右视点成像的歧义性，使得左视点像素在右视点图像中存在多个对应的匹配点。左右视点成像的差异性和歧义性容易造成误匹配，从而使得立体匹配成为十分具有挑战性的问题。

近来，由于深度学习的突破性进展，基于卷积神经网络的方法在图像识别[8]、目标检测[9]和语义分割[10][11]等诸多视觉任务上都取得了目前最好的效果，卷积神经网络展现了强大的特征提取和模型表达能力。传统方法将立体匹配建模为一个多阶段的优化问题，而各个阶段依赖于人工设计的优化准则。研究者们利用卷积神经网络将立体匹配转化一个学习问题，从大量带有 ground truth 的图像数据中自动学习优化表达，在视差估计精度和速度上较传统方法都有很大提升。卷积神经网络为立体匹配研究提供了一条新的思路。

1.2 双目立体视觉概述

1.2.1 双目立体视觉基本原理

在双目立体视觉中，利用两台相机从不同的视角对同一场景进行拍摄，通过匹配两个视角图像，就可以恢复出相应的场景深度信息。

如图 1.2 所示的双目立体对极几何示意图，考虑场景目标中的某点 P，P_l 和 P_r 分别表示 P 在左右两个相机成像平面的成像点。点 P、P_l 和 P_r 构成的平面称为对极平面，对极平面与左右两个相机平面的交线成为对极线，见图 1.2 中的绿线所示。以左视点的成像点 P_l 为基准，通过在右视点的对极线上寻找对应点 P_r，计算两点之间的

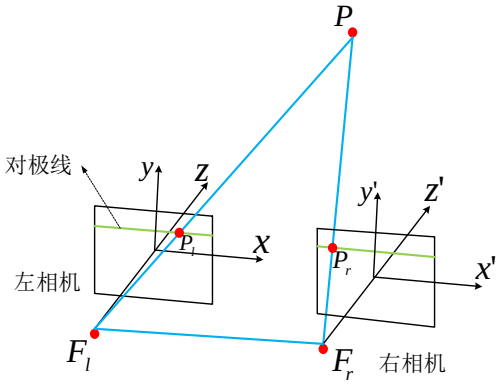


图 1.2 双目立体视觉极线几何示意图。

偏移，根据三角测量原理就可以得到场景目标的深度信息。因此，双目立体视觉的主要目的就是寻找左右视点图像之间相应的匹配像素。

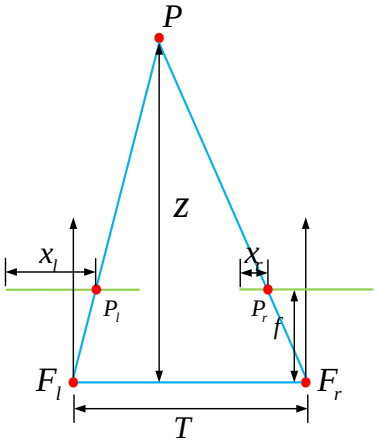


图 1.3 校正后的双目立体视觉几何模型。

为了缩小搜索对应点的区域，通常会进行左右相机的水平校正，使得左视点的像素在右视点只可能出现在对应的水平极线上。如图 1.3 所示为标准形式的双目立体视觉几何示意图，即左右视点经过了水平校正。其中 F_l 和 F_r 分别表示左右相机的光心。考虑场景目标中的某点 P，P_l 和 P_r 分别表示 P 在左右两个相机成像平面的成像点，x_l 和 x_r 分别为 P_l 和 P_r 在相机图像坐标下的水平坐标（以像素为单位）。f 为相机焦距，T 为左右光心之间的距离，即基线距离。P 到基线的距离 Z 可以通过以下三角测量公式推导得到，

$$\frac{T}{Z}=\frac{P_lP_r}{Z-f}=\frac{(T+x_l)-x_r}{Z-f} \tag{1-1}$$

$$Z=\frac{T\bullet f}{x_l-x_r}=\frac{T\bullet f}{d} \tag{1-2}$$

其中的$d=x_l-x_r$即为视差。由公式(1-2)可以发现，场景深度与视差成反比，通过视差我们就可以计算得到深度，从而恢复场景的三维信息。其中焦距f可以通过离线的相机标定得到；视差d通过立体匹配得到，这也是本文的研究对象。以上就是双目立体视觉的基本原理。

1.2.2 双目立体视觉系统

如图 1.1 所示，双目立体视觉系统通常主要包含以下五个步骤：

（1）离线相机标定。此步骤主要为了获得两个相机的内外参。（焦距、图像中心、透视畸变参数和两个相机之间的旋转、平移矩阵。）

（2）图像采集。图像采集是为了实现双目相机对于场景的同步捕获。

（3）双目校正。双目校正是为了消除透视畸变，将立体图像对转化为标准形式，使得左右视点图像水平极线对齐。

（4）立体匹配。寻找左右视点图像之间的匹配对应点，估算视差，得到稠密的视差图。

（5）深度计算。由前述四个步骤得到焦距、基线和视差，利用公式(1-2)计算得到深度，恢复场景三维信息。

1.2.3 立体匹配研究及其难点

本文主要研究对于校正后图像对间的立体匹配。如图 1.4 所示，对于给定的校正后的彩色图像对，以左视图为基准图像。以左视图中位于(x,y)处的像素为例，立体匹配可以理解为在右视图的水平极线上寻找匹配点的过程，搜寻范围为$[x-d_{\max},x]$，进而得到当前位置的视差。对左视图的所有像素进行相同的操作，立体匹配最终输出稠密的视差图。同样也可以以右视图为基准图像进行立体匹配，得到右视图所对应的

视差图。

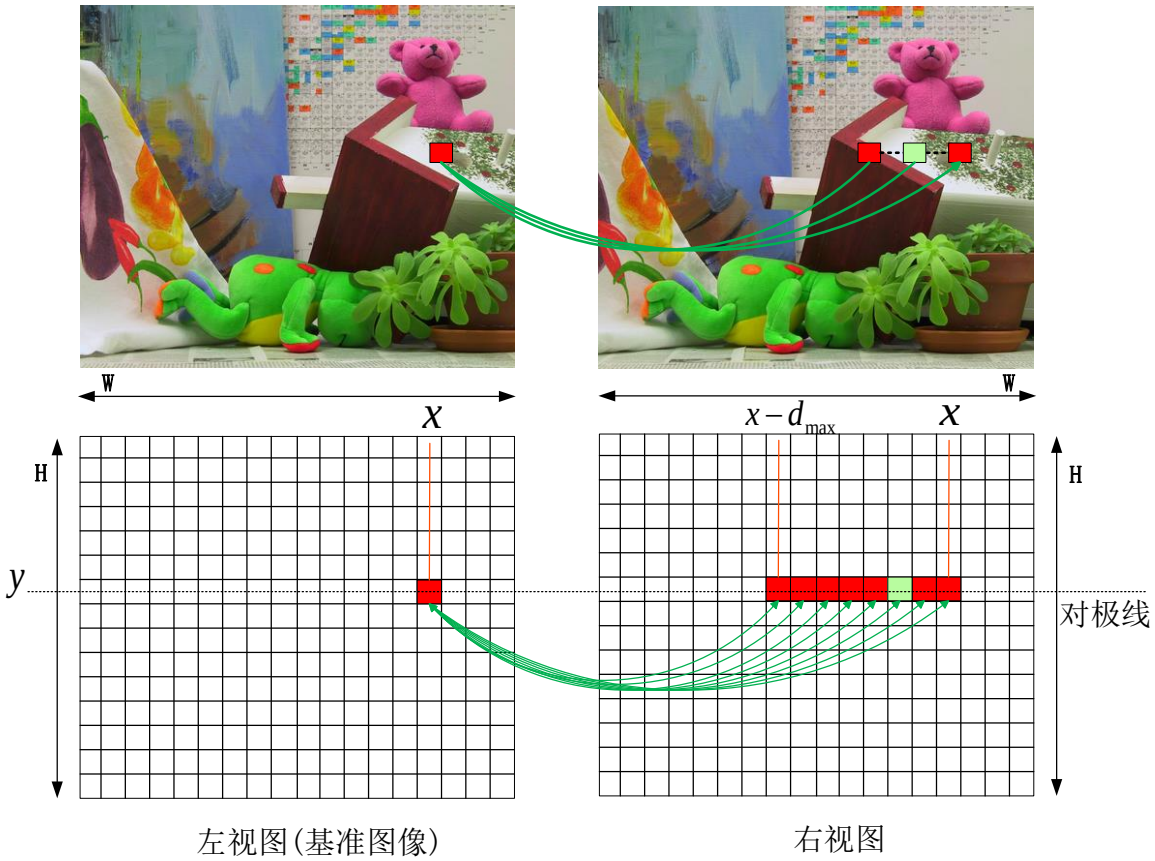
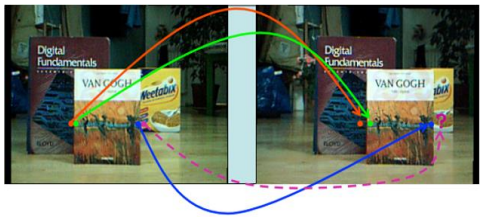


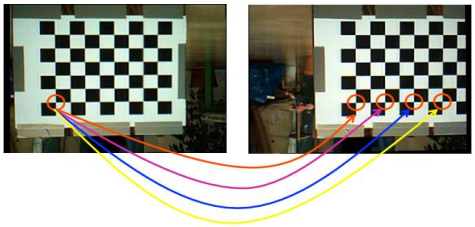
图 1.4 立体匹配示意图。

（a）反射

（b）弱纹理



（c）遮挡



（d）纹理重复

图 1.5 立体匹配中的部分难点。

在立体匹配的过程中，会遇到诸多的匹配难点问题。如图 1.5 所示，对于立体匹配中存在的反射和遮挡区域，左视图的待匹配点在右视图找不到对应的像素点。对于存在的弱纹理平坦区域和纹理重复区域，左视图的待匹配点在右视图中存在多个相似的对应点，增加了匹配的歧义性。此外，成像过程中可能还存在光照变化、成像噪声和光学失真等，这些因素都增加了立体匹配的难度。

针对立体匹配中存在的难点问题，经过多年的发展，来自全球的研究者们已经做出了大量的工作，在接下来的第二章中，我们会对立体匹配的国内外研究现状做出详细的介绍与分析。

1.3 本文主要研究工作及组织结构

本文旨在设计一种基于卷积神经网络的立体匹配算法，主要研究工作可以概括如下：

（1）本文为了在立体匹配中充分利用上下文信息，提出了一个利用层次化上下文信息的端到端立体匹配网络。考虑到成像场景的复杂性，不同区域的像素有着不同的上下文依赖关系，为了适应不同匹配区域像素的上下文信息特点，我们在特征提取过程中，设计了空间金字塔池化模块在不同尺度和位置上提取全局和局部的层次化上下文信息。同时，在优化学习匹配代价阶段，设计了编码-解码结构 3D 卷积模块，通过跳跃连接聚合多尺度的上下文信息，构建了一个端到端的卷积神经网络，实现精细化的视差估计。

（2）目前公开的立体图像数据集中存在遮挡区域与非遮挡区域像素不平衡现象，而基于卷积神经网络的方法会在原始图像中密集采样构建训练样本，从而导致训练样本失衡，就会导致模型训练在遮挡区域出现退化，这与我们希望针对难处理的遮挡区域优化的意图相驰。针对此问题，本文提出了回归聚焦损失函数。回归聚焦损失函数可以在训练过程中自适应地调整样本损失，平衡样本在总损失中的贡献，使模型聚焦在难估计处理的样本上，防止模型退化，从而针对遮挡区域达到更好的优化效果。

本文的组织结构如下：

第一章为绪论。本章主要介绍了本文研究内容的背景和意义、双目立体视觉的基本原理。并且描述立体匹配的概念及其研究难点，最后介绍了本文的主要研究内容和组织结构。

第二章为国内外研究现状。本章分别从传统方法下的立体匹配和深度学习下的立体匹配两方面对立体匹配的研究现状作了详细介绍。

第三章为利用层次化上下文信息的端到端立体匹配算法。本章首先详细介绍了本文提出的端到端立体匹配网络和回归聚焦损失函数。然后从回归聚焦损失函数实验、Middlebury stereo benchmark 实验和网络结构消融实验三个方面对提出的立体匹配算法进行了实验分析。

第四章为总结与展望。本章对本文的研究工作进行总结，并对未来的研究工作做出了展望。

2 国内外研究现状

2.1 引言

立体匹配作为双目立体视觉中最关键的环节，到目前，来自全球的研究者们已经做出了大量的工作。2002 年，Scharstein 和 Szeliski[12]对立体匹配算法进行了全面的综述调研，并将立体匹配归纳为了四个步骤：匹配代价计算、代价聚合、视差最优化计算和视差精细化。同时，这个综述报告也介绍了第一个 Middlebury 数据集和相应的评价指标，此数据集包含通过结构光获得的真实深度图。这样带有 ground truth 的数据集使得利用有监督的统计学习方法来进行立体匹配成为可能。另一方面，随着深度学习在计算视觉任务中的广泛应用，研究者们利用卷积神经网络来处理立体匹配问题，目前的结果也展现了此思路的潜力，逐渐成为研究热点。在本文中，我们根据是否利用了深度学习将现有的立体匹配研究分为两大类，将从传统方法下的立体匹配研究和深度学习下的立体匹配研究这两个大方面来介绍研究现状。

2.2 传统方法下的立体匹配研究

在 1.2.3 节中，介绍到本文研究基于水平校正后彩色图像对的立体匹配，也就是说对于基准图像的像素点(x,y)，对应在参考视图中匹配像素点(x',y')，满足以下约束：

$$x-d_{\max}\le x'\le x-d_{\min},\qquad y'=y \tag{2-1}$$

其中$d_{\max}$和$d_{\min}$为场景的最大视差和最小视差，通常设$d_{\min}=0$，$d_{\max}$为场景给定。

由此，我们可以引出一个重要概念，视差空间图像（disparity space image，DSI）。如图 2.1 所示，DSI 被定义为一个大小为$W\times H\times d_{\max}$的三阶矩阵$C(X,Y,D)$，其中W和H分别为图像的宽和高，每个元素$C(x,y,d)$表示左视图像素点(x,y)与右视图像素$(x-d,y)$之间的匹配代价（也可以视为匹配置信度）。每个切面$C(X,Y,d)$表示每个像素视差为d时的置信度。

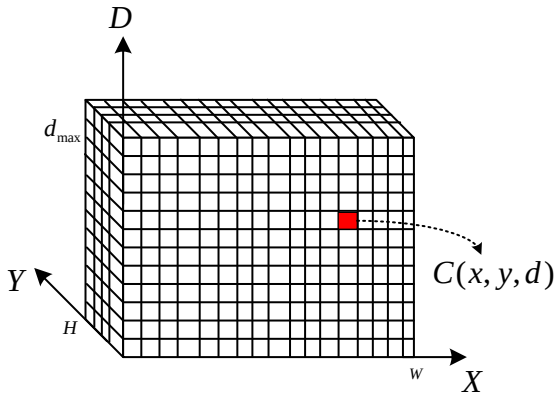


图 2.1 视差空间图像。

立体匹配算法对于每个像素通过在所有潜在视差位置遍历计算匹配代价构建DSI，从而视差估计就可以看成是在 DSI 中寻找一个内嵌的表面，使得 DSI 满足某些最优的性质，如代价最小和最佳平滑性等。

后续的立体匹配算法大都聚焦在如何构建和优化 DSI 上，因此我们根据不同的优化方式将传统方法下的立体匹配细分为了基于局部优化的立体匹配算法和基于全局优化的立体匹配算法。基于局部优化的立体匹配算法通常包含完整的匹配代价计算、代价聚合、视差最优化计算和视差精细化四个步骤，基于全局优化的立体匹配算法通常不包含代价聚合步骤。

2.2.1 基于局部优化的立体匹配研究

基于局部优化的立体匹配算法通过在 DSI 中利用“赢者通吃”（winner-take-all，WTA）的优化策略来实现视差估计。对于基准图像像素 $I_l(x,y)$ 利用 WTA 的策略得到的估计视差为：

$$d^*(x,y)=\underset{0\le d\le d_{\max}}{\arg\min}\,C(x,y,d) \tag{2-2}$$

基于局部优化的立体匹配算法的优化策略非常简单，主要的关注点在如何构建 DSI 上，即匹配代价计算和代价聚合。

（1）匹配代价计算

匹配代价计算以像素点的领域窗口为计算基元，度量与参考视图中对应窗口之

间的相似性。常见的代价计算方法有绝对误差和（sum of absolute differences，SAD），平方误差和（sum of squared differences，SSD），截断绝对误差和（sum of truncated absolute differences，STAD）和归一化互相关系数（normalized cross correlation，NCC）。以左视点中位于(x,y)处的像素为例，当视差为d时，其以 SAD、SSD、STAD 和 NCC 方式计算匹配代价依次表示如下：

$$C_{SAD}(x,y,d)=\sum_{(x,y)\in W}\left|I_l(x',y')-I_r(x'-d,y')\right| \tag{2-3}$$

$$C_{SSD}(x,y,d)=\sum_{(x',y')\in W}\left(I_l(x',y')-I_r(x'-d,y')\right)^2 \tag{2-4}$$

$$C_{STAD}(x,y,d)=\sum_{(x',y')\in W}\min\left\{\left|I_l(x',y')-I_r(x'-d,y')\right|,T\right\} \tag{2-5}$$

$$C_{NCC}(x,y,d)=\frac{\sum_{(x',y')\in W}I_l(x',y')\cdot I_r(x'-d,y')}{\sqrt{\sum_{(x',y')\in W}(I_l(x',y'))^2\cdot\sum_{(x',y')\in W}(I_r(x'-d,y'))^2}} \tag{2-6}$$

其中W为(x,y)的邻域计算窗口，(x',y')为W中的像素坐标，I_l和I_r为分别为左、右视点的灰度图（对于彩色图像，可以对三个颜色通道依次按照上述方式计算，最后对三个通道的代价求平均得到最终匹配代价。本文为了方便标记，均采用灰度图表述。）SAD 和 SSD 度量方式的潜在假设是视点之间成像的像素灰度值是一致的，但当计算窗口存在异常点时，如成像噪声和光学失真造成的像素值异常变化，SAD 和 SSD 就会很不稳定。对于 STAD，其中T为截断阈值。相对于 SAD 和 SSD，STAD 对于异常点有一定的抑制作用，但T需要人为设定。NCC 可以补偿成像增益变化，并且可以消除高斯噪声的影响，但是通常会造成视差不连续处模糊。

为了使得 NCC 对于成像增益和偏置变化具有更好的鲁棒性，同时保持像素值的空间仿射一致性，Martin 和 Crowley [13]提出了零均值归一化互相关系数（zero mean normalized cross correlation，ZNCC），计算方式如下：

$$C_{ZNCC}(x,y,d)=\frac{\sum_{(x',y')\in W}(I_l(x',y')-\bar{I}_l)\cdot(I_r(x'-d,y')-\bar{I}_r)}{\sqrt{\sum_{(x',y')\in W}(I_l(x',y')-\bar{I}_l)^2\cdot\sum_{(x',y')\in W}(I_r(x'-d,y')-\bar{I}_r)^2}} \tag{2-7}$$

其中 $\bar{I}_l$ 和 $\bar{I}_r$ 分别为左右视图中对应的窗口区域像素均值。

Zabih 和 Woodfill[14]将局部的无参变换引入到匹配代价计算，提出了 Rank 变换和 Census 变化。其主要思想就是利用局部区域像素值的相对次序性统计信息来实现匹配代价计算，而不是直接利用像素值大小本身来计算。Rank 变换和 Census 变换会首先对灰度图进行变换，按照像素值的相对大小进行编码。对于灰度图中一点 P，记变换后值为：

$$I_{Rank}(p)=\sum_{q\in N_p}T[I(q)<I(p)] \tag{2-8}$$

$$I_{Census}(p)=\bigcup_{q\in N_p}T[I(q)<I(p)] \tag{2-9}$$

其中 N_p 为 p 的邻域，$T[\bullet]$ 为判断函数，满足中括号中的条件则为 1，不满足则为 0。Rank 变换将灰度值变换为相对大小变化次数，Census 变换将灰度值变换为二值串。在此基础上计算得到匹配代价如下：

$$C_{Rank}(x,y,d)=\sum_{(x',y')\in W}\left|I^l_{Rank}(x',y')-I^r_{Rank}(x'-d,y')\right| \tag{2-10}$$

$$C_{Census}(x,y,d)=\sum_{(x',y')\in W}HAMMING\{I^l_{Census}(x',y'),I^r_{Census}(x'-d,y')\} \tag{2-11}$$

其中 I^l_{Rank} 和 I^r_{Rank} 分别为 Rank 变换后的左右视点图，I^l_{Census} 和 I^r_{Census} 分别为 Census 变换后左右视点图。$HAMMING\{\bullet\}$ 表示计算汉明距离，即计算两个二值串中相异位的个数。Census 变换不仅同 Rank 变换一样保存了像素值的相对次序性关系，而且还保存了局部邻域的空间结构。Rank 变换和 Census 变化由于使用的是像素的相对次序性来计算匹配代价，所以对成像中存在的异常点、相机增益和相机偏置有着不错的鲁棒性。

为了结合 SAD 对于像素值差异的敏感性和 Census 对于噪声的鲁棒性，Mei[15]等人提出了 AD-Census 的融合方式来计算匹配代价。另外，梯度信息[16]和互信息[17]也

可以用来计算匹配代价。本文主要列举了上述具有代表性的匹配代价计算方法，Hirschmller[18]对上述的各种匹配代价计算方式进行了光学失真鲁棒性评估，Census变换的表现最好。关于其他扩展的匹配代价计算方式，具体可以参考文献[18]。

（2）代价聚合

通过计算像素在各个潜在视差位置处的匹配代价，就可以得到除初始 DSI，这里记为$C_0(X,Y,D)$。若直接在$C_0(X,Y,D)$上运用 WTA 优化得到视差图，往往得不到好的结果，因为$C_0(X,Y,D)$是由各个像素独立计算匹配代价构造的，容易受到噪声的影响，且在视差不连续区域，容易造成视差图模糊。代价聚合就是在一个支撑区域内对于$C_0(X,Y,D)$进行加权计算的过程，其主要思想是因为单个像素匹配的不稳定，所以就考虑选择和领域像素一起进行支撑计算，支撑区域选择的假设是区域内的像素都有着相似的视差[12]。考虑视差为d时，取$C_0(X,Y,D)$的一个切面$C_0(X,Y,d)$，代价聚合可以看作是在支撑区域内对$C_0(X,Y,d)$进行 2D 卷积，

$$C(X,Y,d)=W(X,Y,d)*C_0(X,Y,d) \tag{2-12}$$

其中$W(X,Y,d)$为支撑区域卷积核，$C(X,Y,d)$为代价聚合后的 DSI。

支撑窗口的选择是代价聚合的关键，当支撑窗口区域覆盖的区域不具有相同的视差时，局部匹配的方法很容易失败。为了在视差不连续区域取得精确的视差估计，局部支撑窗口的选择期望能自适应图像区域的形状和大小，因此窗口仅选取具有相同深度的像素来进行支撑计算。我们可以将现有的局部代价聚合方法分为两大类。

第一类方法要么聚焦在从预定义的多个窗口中选择最优的支撑窗口，要么聚焦在对局部的支撑窗口的形状和大小进行像素自适应。预定义支撑窗口的这类方法的一个常见限制是局部支撑窗口的形状是矩形状的或限定的，因此不适用于任意形状深度不连续附近区域的像素。Zhang[19]等人提出了十字交叉的自适应支撑窗口的代价聚合方法，在每个像素位置通过像素颜色相似性和关联性约束自适应构建十字交叉区

域，可以准确地估计场景结构。Yang[20]提出一种非局部的代价聚合方法，将支撑窗口扩展到整幅图像，通过在由像素构成的图结构中计算最小生成树的方式进行代价聚合，可以保留视差图的边缘信息。Zhang[21]等人引入多尺度的代价聚合方式来模拟人类视觉机制，在多个图像分辨率尺度计算初始 DSI，再在多个 DSI 上进行代价聚合，由粗到细确定最终视差。

第二类方法在固定支撑窗口的形状和大小后，聚焦在考虑调整窗口内每个位置的权重。Xu[22]等人通过径向计算确定自适应支撑权值，但这种方法对初始的 DSI 估计非常敏感。Yoon 和 Kweon[23]根据颜色相似性和几何接近度为支撑窗口中像素分配支撑权重。尽管该方法的视差结果相对准确，但由于存储了中心像素相关的权重，因此消耗了大量的内存。Tombari[24]等人利用图像分割信息，提出了对一个大支撑窗口（51x51）中每个像素修改权重函数，此方法以极度增大计算量为代价来提高视差估计准确率。

（3）视差精细化

此步骤可以视为后处理。在对经过匹配代价计算和代价聚合步骤后构建 DSI，运用 WTA 优化就可以得到出初始视差图 D_0。此时得到的视差图每个像素位置的值都为整数。由于视差与深度成反比的关系，为了得到更精确的深度信息，就需要对初始视差图进行精细化处理。对于像素 $p(x,y)$，精细化视差 $D_{refinement}(x,y)$ 可以表示为：

$$D_{refinement}(x,y)=D_0(x,y)-\frac{C_+-C_-}{2[C_++C_--2C]} \tag{2-13}$$

其中 $C_+=C(x,y,D_0(x,y)+1)$ ，$C_-=C(x,y,D_0(x,y)-1)$，$C=C(x,y,D_0(x,y))$。此外，还可以对视差图进行左右一致性检验和中值滤波进一步提高精度。

总的来说，基于局部优化方法的方法主要考虑设计匹配代价计算和代价聚合的方法来提高匹配准确度，但这类方法仅依赖于局部窗口的像素信息，对于纹理丰富的区域有着不错的匹配效果，并且算法复杂度较低，但对于遮挡、无纹理和弱纹理等难点区域有着较高的误匹配率。

2.2.2 基于全局优化的立体匹配研究

WTA 的局部优化方法只是在 DSI 上对单个像素取最小值得到视差，而全局优化则是在视差优化计算阶段考虑全部的像素，加入相邻像素满足的先验约束，通常约束像素之间具有视差平滑性。全局优化的一般形式就是在整张图像上构建能量函数，通过最优化能量函数，求得最终的视差值。能量函数的一般形式如下：

$$E(D) = E_{data}(D) + E_{prior}(D) \tag{2-14}$$

其中 D 为每个像素所对应的视差，$E_{data}(D)$表示视差为D时，左右视图像素之间不一致性代价，$E_{prior}(D)$表示视差为D时的先验约束[31]。最终的视差图D^*可以通过最小化$E(D)$得到：

$$D^* = \arg\min_{D\in[0,d_{\max}]} E(D) \tag{2-15}$$

为了得到D^*，现有的方法可以分为两大类：第一类将能量函数构建在离散空间，即视差值是整数值，这样立体匹配问题就转变为了标注问题。第二类将能量函数构建在连续空间，这种能量函数通常利用全变分框架优化。

当最小化能量E视为标注问题时，能量函数E的形式如下：

$$E_{labeling}(D) = \sum_{p\in I_l} C(p, D(p)) + \sum_{i,j\in N(P)} V_{i,j}(D(i), D(j)) \tag{2-16}$$

其中p是像素，I_r是左视图，$C(p, D(p))$为匹配代价（计算方法同 2.2.1 节中匹配代价计算方式一样），对应着(2-14)式中的数据项。$V_{i,j}(D(i), D(j))$表示两个在p的邻域中$N(p)$的两个相邻像素i, j分别对应视差为$D(i)$和$D(j)$的代价，对应着(2-14)式中先验项。马尔科夫随机场（Markov random field，MRF）被广泛地应用于解决标注问题。尽管通过 MRF 模型求解能量最小化是一个 NP 难问题，但置信传播（belief propagation，BP）、图割（graph cuts，GC）和动态规划（dynamic programming，DP）等方法已经被验证可以取得很好的近似结果[25][26][27]。

Felzenszwalb 和 Huttenlocher[26]提出一种多尺度的 BP 公式来进行优化求解，只需要小量恒定的迭代，但复杂度还是相对较高。Psota[28]在最小生成树上进行有效的视差传递，从而获得最大后验视差估计。Zabih[29]等人第一次将利 GC 引入到立体匹配

中视差能量最小优化。Taniai[30]等人利用迭代的 GC 优化进行视差估计，在弱纹理区域取得了不错的效果，但是计算复杂度较高。在立体匹配中，DP 通常对每个扫描线独立地执行优化，Ohta 和 Kanade[31]提出了一种将多扫描线信息整合到优化中的方法，他们分两个阶段进行动态规划:首先对每个边缘像素进行扫描线内优化，然后利用边缘信息进行扫描线间的优化，以确保扫描线之间的一致性。

Hirschmuller[17]等人提出半全局匹配（Semi-Global Matching，SGM）方法以便仅使用局部信息来提供全局代价计算的近似。这种高效的方法可以在基于局部优化方法的运行时间内实现，同时减少了孔径问题的影响。SGM 代价函数与（2-6）式一致，核心思想是用多个一维空间的优化结果来替代二维空间的优化结果，从而将二维的 DP 问题的转化为多个一维 DP 问题。与之前表述的不包含代价聚合步骤的全局优化算法不同，SGM 的数据项会从多个方向上进行一维的代价聚合。Schonberger[32]等人在出一种基于学习的扫描方向优化方法 SGM-forest，提高了 SGM 算法的匹配精度。

对于构建在离散空间的全局优化算法，最优化计算后得到的初始视差图，依然可以利用在 2.2.1 节中介绍到的后处理方法得到精细化视差图，这里就不再赘述。

变分方法将能量函数表示为：

$$E_{TV}(u)=\int_{\Omega}\psi_D(u)dp+\lambda\int_{\Omega}\psi_S(u)dp \tag{2-17}$$

其中，Ω 表示整个图像域；u 对应着 $u(p)$，表示像素 p 所在的视差域；$\psi_D(u)$ 对应着的数据项；$\psi_S(u)$ 是的正则项或者平滑项，用来惩罚邻域内视差值的变化；λ 是平滑项相对数据项的权重。

Ranftl[33]等人首先提出了一种变分框架来解决立体匹配问题，他们设计了一种利用梯度信息的正则化器，以便只在均匀区域使用平滑性，而在其他情况下则使用较低的平滑性，取得了不错的效果。但是这种正则化倾向于正则平行表面。Kuschk 和 Cremers[34]通过合并边缘信息扩展了 Ranftl[33]等人提出的正则化器，以避免偏向于正平行表面，然后利用梯度下降法迭代地最小化能量函数，并且将算法利用 GPU 实现。与文献[33]相比，他们的错误率和运行效率的都得到了提升。

相较于局部优化方法，考虑所有像素信息的全局优化算法以高昂的计算复杂度为代价，可以取得更为精确的估计视差图，尤其是在弱纹理和无纹理区域，但对于遮挡和反射等难点区域依然存在较高的误匹配率。基于局部优化的立体匹配算法实现起来相对简单，但往往精度不高。基于半全局优化的立体匹配算法，如 SGM 算法，可以在计算精度与运算效率之间取得折中。

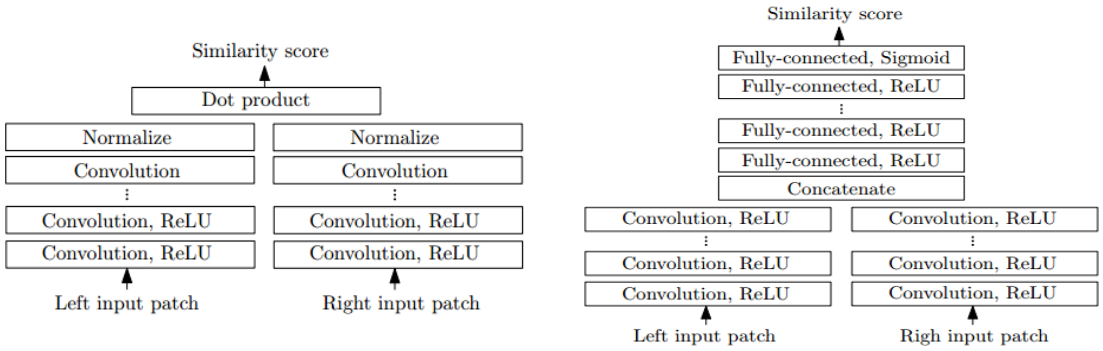
2.3 深度学习下的立体匹配研究

在传统立体匹配算法中一个非常重要的环节就是计算匹配代价，本质上是度量两个图像块的相似度。考虑到卷积神经网络强大的特征表达和学习能力，深度学习下的立体匹配研究的一个重要思路就是利用卷积神经网络来进行有监督地学习匹配代价，以替代传统方法中人工设计的代价计算准则。

相较于为了计算图像块之间的匹配代价而单独设计网络结构进行分步优化的方式，研究者们试图利用卷积神经网络直接实现像素级的视差输出，构建端到端的立体匹配网络进行有监督的学习训练则是另一个重要思路。

基于上述分析，深度学习下的立体匹配研究的又可以分为两大类：基于卷积神经网络的匹配代价计算研究和基于卷积神经网络的端到端立体匹配研究。

2.3.1 基于卷积神经网络的匹配代价计算研究



(a) MC-CNN-fst 网络　　(b) MC-CNN-acrt 网络

图 2.2 MC-CNN[35][36]的两种网络结构示意图。

Zbontar 和 LeCun[35][36]将卷积神经网络引入到立体匹配研究中来，其核心思想就是将传统的匹配代价计算转化为了一个二分类问题。作者首先根据 ground truth，在左右视点图像上对应位置上选取9×9的图像块组成训练数据，并打上标正负标签，正标签表示图相对相似，负标签表示不相似，然后设计了两种监督学习网络，快速网络 MC-CNN-fst 和精确网络 MC-CNN-acrt。如图 2.2 所示，MC-CNN-fast 和 MC-CNN-acrt 都通过网络提取图像块特征，在特征维度上计算相似性。不同的是，MC-CNN-acrt 直接将左右图像块的特征向量进行点积运算作为最后的相似性分数，采用合页损失函数训练；MC-CNN-acrt 引入全连接层，采用交叉熵损失函数训练。MC-CNN-acrt 由于引入了全连接层，所以计算速度较慢，但是相似性度量更为准确。在 MC-CNN 算法中，匹配代价的计算由传统方法中人工设计的准则换成了由训练的网络模型来预测，后续依然采用了传统匹配方法中的十字交叉代价聚合[19]、SGM[17]优化和视差精细化等步骤。MC-CNN 超过了传统的匹配算法，取得了当时最好的结果。研究者们在 MC-CNN 的思路上做了更深的研究，Luo[37]等人提出了一个快速的网络，网络结构与 MC-CNN-fst 相似，但是将完整的右视图作为输入提取特征，将计算匹配代价转化为一个多分类的问题，类别数为所有可能视差值，即为最大视差加一，可以直接输出各个潜在视差位置的匹配代价，在算法效率上有非常大的提高。Chen[38]等人设计了一个新的匹配代价计算网络，直接输入左右视点图像进行特征提取，并在不同尺度阶段，应用滑动窗口法在各个视差位置应用向量内积计算代价，最后融合不同尺度的代价作为最后输出。Shaked 和 Wolf[39]提出了一个高速通路的网络来计算匹配代价，通过多个层级的加权残差块来提取特征，并且在训练过程中利用了交叉熵损失和合页损失的混合损失。Seki[40]依然将匹配代价计算视为一个的二分类问题，采用与 MC-CNN-acrt 类似的网络结构，作者考虑到像素的邻域像素具有一致的视差时图像块更有可能匹配，从而设计了与 SGM 联合的匹配置信度融合方法。

2.3.2 基于卷积神经网络的端到端立体匹配研究

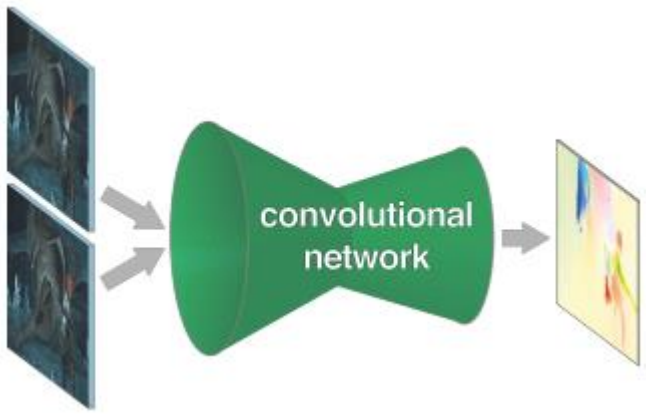


图 2.3 DispNet[42]网络结构示意图。

2015 年，Long[41]等人利用全卷积神经网络（Fully convolutional network，FCN）在语义分割中取得了非常好的效果。基于 FCN 的启发，Mayer[42]等人第一次将端到端的神经网络的引入到光流估计和视差估计中，针对视差估计提出了 DispNet。如图 2.3 所示的 DispNet 的结构示意图，其整体结构和 FCN 类似，不过考虑到立体匹配有左右视点图像两个输入，所以 DispNet 以共享权值的方式对左右视图并行地提取特征。通过卷积和池化操作，提取到多个尺度层级的特征图。与 FCN 不同的是，DispNet 并没有直接对不同尺度的特征图进行转置卷积得到原始分辨率的视差图，而是采用编码-解码的对称结构，逐步转置卷积恢复到原始分辨率，实现由细到粗，再由粗到细的视差推理过程。最后，对估计视差图中每个像素和真实视差值计算欧式距离构建损失来进行端到端的训练。Pang[43]等人提出了一个包含两个阶段的级联网络 CRL，第一阶段在 DispNet 的基础上增加额外的卷积上采样，从而得到具有更多细节的视差图像。第二阶段显式修正第一阶段的初始化视差，通过在多个尺度上与第一阶段耦合进行残差学习，最终将两个阶段的输出相加就得到了最终的视差图。Kendall[44]等人提出了一个端到端的网络 GC-Net，考虑在立体匹配中利用上下文信息和场景几何信息，并且 GC-Net 将立体匹配转化为回归问题，在不需要后处理的情况，直接实现精细化的视差输出。Liang[45]等人提出了一个整合传统方法中四个步骤的端到端网络 iResNet，通过卷积神经网络来提取图像特征，将代价计算、代价聚合和视差优化计

算操作整合到一个网络中，得到初始视差图，紧接着设计了一个子网络，通过与原图像空间中的特征一致性来反馈优化初始视差，最终输出优化后的视差图。

总的来说，基于端到端的立体匹配算法都试图避免人工设计的浅层表达，从而充分利用卷积神经网络的特征提取和模型表达能力，自发地从数据中学习约束表达。上述端到端的算法都利用了多尺度的特征来进行视差估计，主要思想就是在立体匹配中考虑上下文信息以减少误匹配。DispNet[42]和 GC-Net[44]利用 FCN[41]的思想，通过堆叠浅层特征和深层特征来反复利用层次信息。CRL[43]则在不同分辨率下进行层次化的监督训练。实际上，在传统方法中代价聚合和加入先验项的手段就是为了考虑像素的上下文信息。对于左右视点图像存在成像差异和歧义的难点区域，如遮挡和弱纹理区域，利用像素的上下文信息可以有效地进行视差推断。在接下来的章节将会介绍到本文提出的利用层次化上下文信息的端到端立体匹配算法。

2.4 本章小结

本章分别从传统方法下的立体匹配和深度学习下的立体匹配对立体匹配的国内外研究现状进行了全面的介绍与分析。传统方法下的立体匹配算法通常包含匹配代价计算、代价聚合、视差最优化计算和视差精细化等步骤。在视差最优化阶段又可以根据优化方法分为局部方法和全局方法。全局优化方法相对局部优化方法可以取得更高精度的估计视差，但是要以高昂的计算复杂度为代价。由于深度学习强大的特征提取和模型表达能力，利用卷积神经网络的立体匹配算法开始流行起来。相对于仅利用卷积神经网络来计算基于图像块的匹配代价，而将立体匹配转化为回归问题，设计端到端的立体匹配网络，成为目前基于深度学习的立体匹配研究的热点方向。

3 利用层次化上下文信息的端到端立体匹配算法

3.1 引言

随着带有 ground truth 的数据集公开，使得利用有监督的机器学习方法进行立体匹配成为可能，一些立体匹配算法利用深度神经网络来学习匹配代价，再将匹配代价与人工设计的优化函数和后处理方法相结合，从而得到最终的视差图。如此分步优化和依赖人工设计约束函数的方法，并不能充分发挥深度神经网络的模型表达能力。构造端到端的卷积神经网络实现从数据中自动学习匹配代价和代价优化的深层表达，成为目前基于深度学习研究的主流方向。

传统方法中基于局部优化的立体匹配方法，通过代价聚合步骤汇聚邻域像素的代价来表征当前像素匹配代价，以提高匹配鲁棒性；基于全局优化的立体匹配算法，通过最小化能量函数的全局优化策略在视差空间中求得视差值，能量函数通常包含数据项和先验项，先验项用来衡量相邻像素视差之间的差异性。无论是局部优化中的代价聚合步骤，还是全局优化中加入的先验项，都是在考虑邻域像素之间的相关性，也就像素的上下文信息。上下文信息对于立体匹配中难点区域的视差计算是十分有帮助的，如在遮挡、反射等容易发生误匹配的区域，利用像素的上下文信息就可以很好地进行视差推断。而传统方法中加入的上下文约束都是人工设计的浅层表达[44]，如何在基于卷积神经网络的端到端的在立体匹配中，通过深度学习的方式充分利用上下文信息是本章算法的主要出发点。

另外，目前公开的立体图像数据集中存在遮挡区域与非遮挡区域像素不平衡现象，而基于卷积神经网络的方法会在原始图像中密集采样构建训练样本，从而导致训练样本失衡，就会导致模型训练在遮挡区域出现退化，这与我们希望针对难处理的遮挡区域优化的意图相驰。针对此问题，本文提出了回归聚焦损失函数进行网络监督训练。

本章接下来将详细介绍提出的利用层次化上下文信息的端到端立体匹配算法。

3.2 利用层次化上下文信息的端到端立体匹配算法

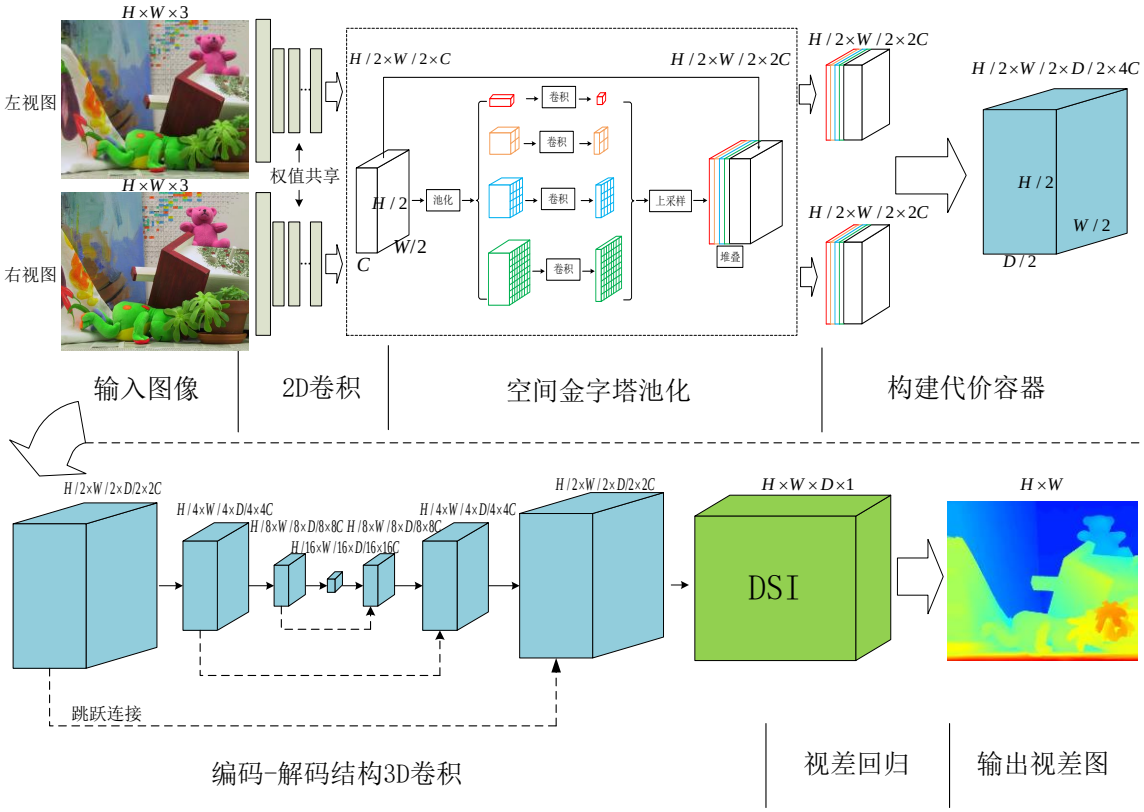


图 3.1 利用层次化上下文信息的端到端立体匹配算法框架图。

图 3.1 所示为本文提出的利用层次化上下文信息的端到端立体匹配算法框架图，可以分为以下几个模块：2D 卷积特征提取、空间金字塔池化、构建匹配代价容器、编码-解码结构 3D 卷积和视差回归。

本算法的主要出发点是在立体匹配中充分利用上下文信息进行视差推断，实现端到端的立体匹配。考虑到成像场景和物体的复杂性，不同匹配区域像素有着不同的上下文依赖关系，为了适应不同区域像素的上下文信息特点，本算法在 GC-Net[44]的基础上，在特提取阶段设计空间金字塔池化模块以提取不同尺度和位置上的层次化上下文信息；同时，在匹配代价学习优化阶段构建了编码-解码结构 3D 卷积模块。设计了一个端到端的全卷积神经网络，从而实现输入左右视点图像，直接回归输出精细化的视差图。

在图 3.1 所示的框架图中，由于构建代价容器和编码-解码 3D 卷积阶段涉及到 4 阶张量，故在示意图中省去了特征图数量这一维度以便于显示。网络的具体结构和每个阶段的输出张量维度可以见表 3-1，其中 H 和 W 为输入图像的高和宽，D 为最大视差 $D_{\max}$ 加 1，C 特征图数量。

表 3-1 网络结构

	网络层描述	输出张量维度
	输入图像	$H\times W\times 3$
2D 卷积特征提取（3.2.1 节）		
1	5×5 卷积，C 个卷积核，步长为 2	$H/2\times W/2\times C$
2	3×3 卷积，C 个卷积核，步长为 1	$H/2\times W/2\times C$
3	3×3 卷积，C 个卷积核，步长为 1	$H/2\times W/2\times C$
	将第 1 层输出与第 3 层输出逐层相加（跳跃连接）	$H/2\times W/2\times C$
4-17	重复 7 次第 2 层与第 3 层构成的残差块	$H/2\times W/2\times C$
18	3×3 卷积，C 个卷积核，步长为 1	$H/2\times W/2\times C$
空间金字塔池化（3.2.2 节）		
19	$H/2\times W/2$ 均值池化，步长为 $H/2\times W/2$	$1\times 1\times C$
20	1×1 卷积，$C/4$ 个卷积核，步长为 1	$1\times 1\times C/4$
21	$H/4\times W/4$ 均值池化，步长为 $H/4\times W/4$	$2\times 2\times C$
22	1×1 卷积，$C/4$ 个卷积核，步长为 1	$2\times 2\times C/4$
23	$H/8\times W/8$ 均值池化，步长为 $H/8\times W/8$	$4\times 4\times C$
24	1×1 卷积，$C/4$ 个卷积核，步长为 1	$4\times 4\times C/4$
25	$H/16\times W/16$ 均值池化，步长为 $H/16\times W/16$	$8\times 8\times C$
26	1×1 卷积，$C/4$ 个卷积核，步长为 1	$8\times 8\times C/4$
	将 20，22，24，26 层的输出双线性插值，并与 18 层输出堆叠	$H/2\times W/2\times 2C$

构建匹配代价容器		
	见 3.2.3 节	$H/2\times W/2\times D/2\times 4C$
编码-解码 3D 卷积（3.2.4 节）		
27	3×3×3卷积，$2C$ 个卷积核，步长为 1	$H/2\times W/2\times D/2\times 2C$
28	3×3×3卷积，$4C$ 个卷积核，步长为 2	$H/4\times W/4\times D/4\times 4C$
29	3×3×3卷积，$4C$ 个卷积核，步长为 1	$H/4\times W/4\times D/4\times 4C$
30	3×3×3卷积，$8C$ 个卷积核，步长为 2	$H/8\times W/8\times D/8\times 8C$
31	3×3×3卷积，$8C$ 个卷积核，步长为 1	$H/8\times W/8\times D/8\times 8C$
32	3×3×3卷积，$16C$ 个卷积核，步长为 2	$H/16\times W/16\times D/16\times 16C$
33	3×3×3卷积，$16C$ 个卷积核，步长为 1	$H/16\times W/16\times D/16\times 16C$
34	3×3×3卷积，$16C$ 个卷积核，步长为 1	$H/16\times W/16\times D/16\times 16C$
35	3×3×3转置卷积，$8C$ 个卷积核，步长为 2	$H/8\times W/8\times D/8\times 8C$
	将第 31 层输出与第 35 层输出逐层相加（跳跃连接）	$H/8\times W/8\times D/8\times 8C$
36	3×3×3转置卷积，$4C$ 个卷积核，步长为 2	$H/4\times W/4\times D/4\times 4C$
	将第 29 层输出与第 36 层输出逐层相加（跳跃连接）	$H/4\times W/4\times D/4\times 4C$
37	3×3×3转置卷积，$2C$ 个卷积核，步长为 2	$H/2\times W/2\times D/2\times 2C$
	将第 27 层输出与第 37 层输出逐层相加（跳跃连接）	$H/2\times W/2\times D/2\times 2C$
38	3×3×3转置卷积，1 个卷积核，步长为 2	$H\times W\times D\times 1$
视差回归		
	见 3.2.5 节	$H\times W$

3.2.1 2D 卷积特征提取

在的 2.2.1 节中，我们介绍了传统方法中比较有代表性的匹配代价计算方式，如 SAD、SSD、NCC 和 ZNCC 等方法直接依赖于原始像素值的强度，Rank 变换和 Census 变换利用局部像素大小的相对次序性来进行计算，计算从原始的像素空间转化为了次序性空间。同样，对原始图像进行 2D 卷积特征提取操作，目的是为了将原始图像

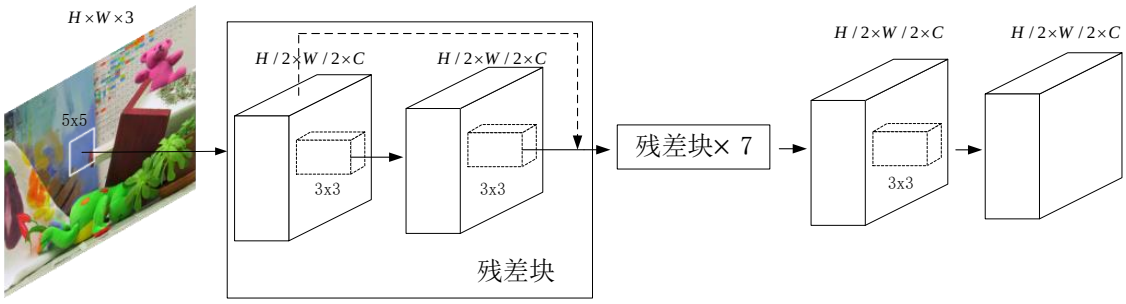


图 3.2 2D 卷积特征提取。

空间转化到深层特征空间，从而在深层特征空间中计算匹配代价。

如图 3.2 所示，我们利用残差网络来进行特征提取，我们将两个3×3的卷积通过跳跃连接构成了一个残差块[46]，其中每个 2D 卷积操作的实际组成为 2D 卷积+BN[47]+Relu。结合表 3-1 与图 3.2，我们首先会对输入图像以 2 为步长进行5×5的卷积，此时特征图的尺度为$H/2\times W/2$，然后经过 8 个残差组和3×3卷积完成特征提取，输出大小的为$H/2\times W/2\times C$的特征图。由于立体匹配度有左右视点图像两个输入，本文用共享权值的相同网络分别对左右视点进行特征提取，保证特征提取的一致性。

3.2.2 空间金字塔池化

在本章的引言介绍到，本章算法考虑在基于卷积神经网络的端到端立体匹配中充分利用上下文信息，空间金字塔池化就是提取层次化上下文信息的关键。

在卷积神经网络中，池化操作可以用来提取上下文信息。在图像分类中，全局池化被经常使用来提取上下文信息。考虑到成像场景的复杂性，对于不同区域，我们希望提取不同层次范围的上下文信息，如对于弱纹理的平坦区域，上下文范围就应大些，对于纹理丰富的区域，我们期望对应的上下文区域就较小些。仅仅使用全局池化提取最粗糙的上下文信息，就容易丢失掉图像的空间相关性。Zhao[48]等人在提出的PSPNet 中利用空间金字塔池化来提取上下信息进行语义分割，达到了当时最好的效果。空间金字塔池化的核心思想就是多尺度和层次化地提取上下文信息，以减轻不同子区域之间的上下文信息丢失。在本文算法中，我们利用空间金字塔池化在不同尺度

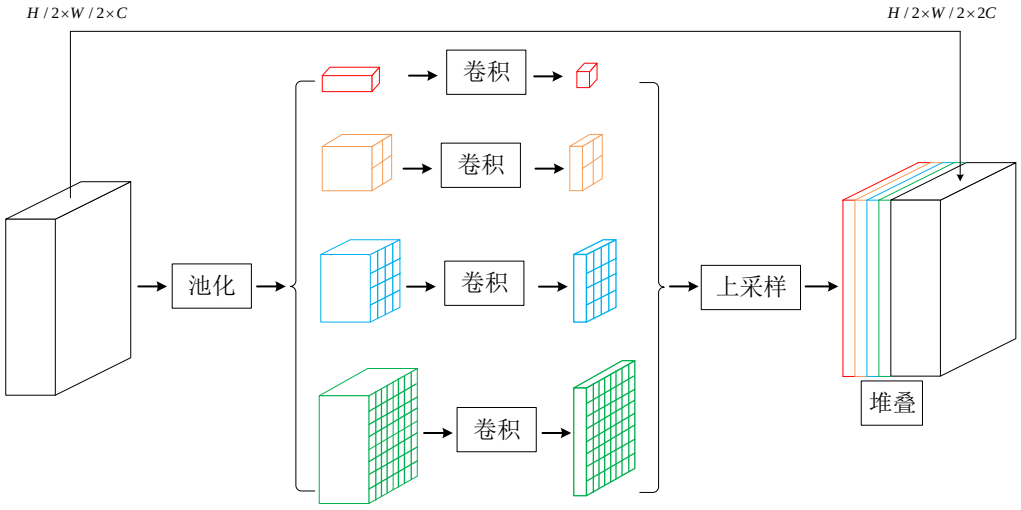


图 3.3 空间金字塔池化。

和位置上提取全局和局部的层次化上下文信息来处理立体匹配问题。

本文设计了如图 3.3 所示的空间金字塔池化模块，使用了 4 个不同层级的平均池化操作（平均池化相较最大池化计算涉及到子区域内的所有值，更利于提取子区域的全局信息），池化后特征图的大小分别为1×1、2×2、4×4和8×8，即将 2D 卷积特征图划分为1×1、2×2、4×4和8×8的子区域，在这些的子区域上提取上下文信息。紧接着使用1×1的卷积来减小特征维度，均为输入特征图数量的 1/4。利用双线性插值恢复到 2D 卷积提取的特征图尺寸，最后和 2D 卷积提取的特征图堆叠起来，最终得到包含层次化上下文信息，尺寸为 $H/2\times W/2\times 2C$ 特征图。

3.2.3 构建匹配代价容器

在经过 2D 卷积特征提取和空间金字塔池化后，我们可以得到左右视点融合深层图像特征和层次化上下文信息的特征图。构建匹配代价容器的目的就是组合左右视点图像的特征图，以便利用卷积网络来学习匹配代价构建 DSI。MC-CNN-fst[36]通过计算特征向量的点积来组合图像特征，MC-CNN-acrt[36]和 GC-Net[44]则通过堆叠的方式组合特征图。相对于向量点积等距离度量的方式，堆叠的方式能够维持特征图的维度，从而可以通过后续的优化手段完整地利用提取到的图像特征。本文利用和 GC-Net[44]同样的方式构建匹配代价容器，在堆叠过程中加入双目立体几何约束，即当视

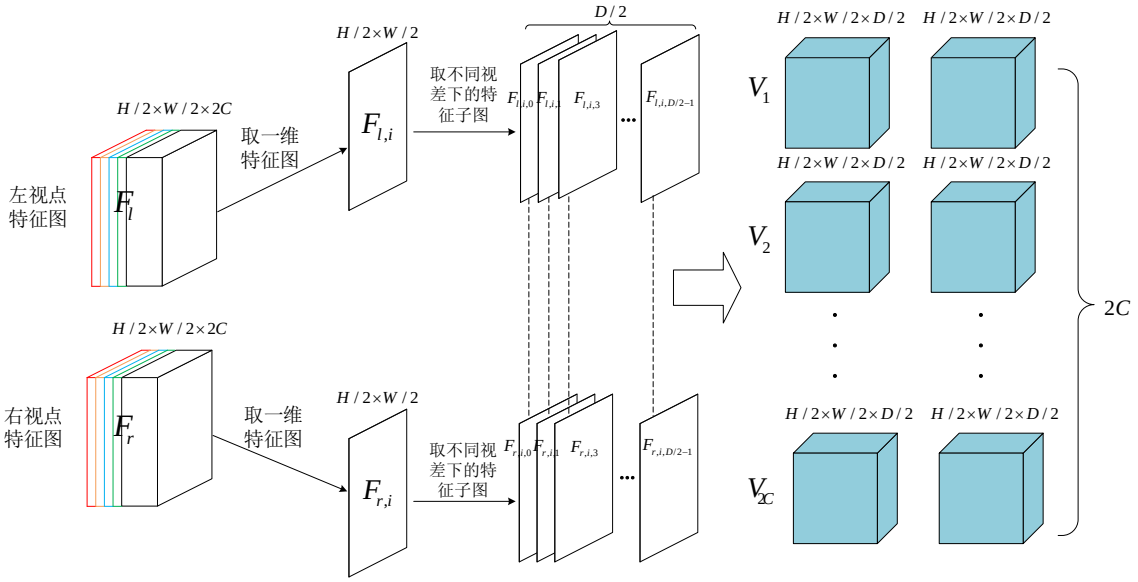


图 3.4 构建匹配代价容器。

差为 d 时，左视点像素 (x, y) 应与右视点像素 $(x-d, y)$ 相对应。

如图 3.4 所示，记左右视点特征点分别为 F_l 和 F_r，分别取其中的第 i 维特征图 $F_{l,i}$ 和 $F_{r,i}$。对 $F_{l,i}$ 和 $F_{r,i}$ 在不同视差值下提取特征图，并根据视差与之对应，构建代价容器在第 i 维大小为 $H/2\times W/2\times D$ 的分量 V_i，。具体操作为，当取视差 d 时，$F_{l,i}$ 提取到的特征图为第 d 列到第 $W/2$ 列的子图 $F_{l,i,d}$，$F_{r,i}$ 提取到的特征图为第 1 列到第 $W/2-d$ 列的子图 $F_{r,i,d}$。遍历所有可能的视差 $d\in[0, D/2-1]$，$F_{l,i}$ 和 $F_{r,i}$ 就能各得到 $D/2$ 张特征图，将这 D 张特征图堆叠在一起就构成了 V_i。最后遍历左右视点的所有特征图，就得到了大小为 $H/2\times W/2\times D/2\times 4C$ 的匹配代价容器。

3.2.4 编码-解码结构 3D 卷积

由前述步骤，我们得到了一个 4 阶的匹配代价容器，代价容器包含视差维度和空间维度上丰富的特征信息。我们希望利用卷积网络从代价容器中学习优化匹配代价，最后得到需要的视差空间图像 DSI。在 4 阶匹配代价容器上进行 3D 卷积操作，可以在高、宽和视差三个维度上学习丰富的特征约束关系。与 GC-Net 直接通过转置卷积的方式得到 DSI 的方式不同，本文设计了编码-解码结构 3D 卷积。为了学习到

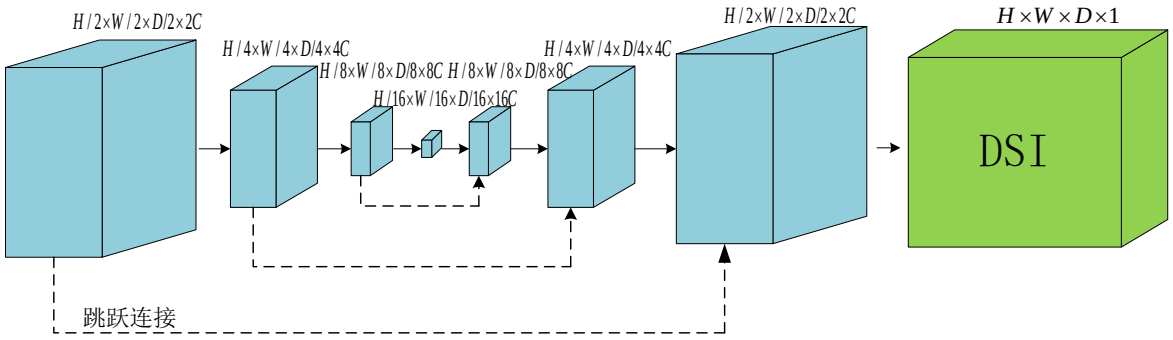


图 3.5 编码-解码结构 3D 卷积。

更多的上下文信息，我们使用编码-解码结构实现由细到粗再由粗到细的学习过程，通过跳跃连接聚合上下文信息。

图 3.5 是本文编码-解码结构 3D 卷积的示意图，省略了部分卷积层和特征图数量维度，具体的卷积过程可以参见表 3-1。匹配代价容器经过了 3 次 3D 卷积下采样和对应的 3 次的 3D 转置卷积，最后再通过一次 3D 转置卷积上采样，得到尺寸为 $H \times W \times D$ 的 DSI。

3.2.5 视差回归

通过对 DSI 运用 WTA 优化策略就可以得到最终的视差图。然而，这个操作会造成两个问题：1、无法达到精细化的视差估计；2、此最小化操作不可导，从而不能进行反向传播训练。

由 DSI 我们可以得到每个像素在视差为 d 时的匹配代价 C_d，从而可以利用的 soft argmin[49]操作来计算每个视差值的置信度，再通过置信度加权求和得到估计视差 $\hat{d}$。具体计算方式如下：

$$\hat{d} = \sum_{d=0}^{D\max} d \times \sigma(-C_d) \tag{3-1}$$

其中 $\sigma(\bullet)$ 对应的是 softmax 操作，

$$\sigma(-C_d) = \frac{e^{C_d}}{\sum_{i=0}^{D_{\max}} e^{C_i}} \tag{3-2}$$

因为匹配代价越低对应置信度越高，所以会在匹配代价前面加上负号，从而构成 soft argmin 操作。此方法不仅可导，而且可以通过回归方式得到的精细化视差估计。

3.2.6 回归聚焦损失函数

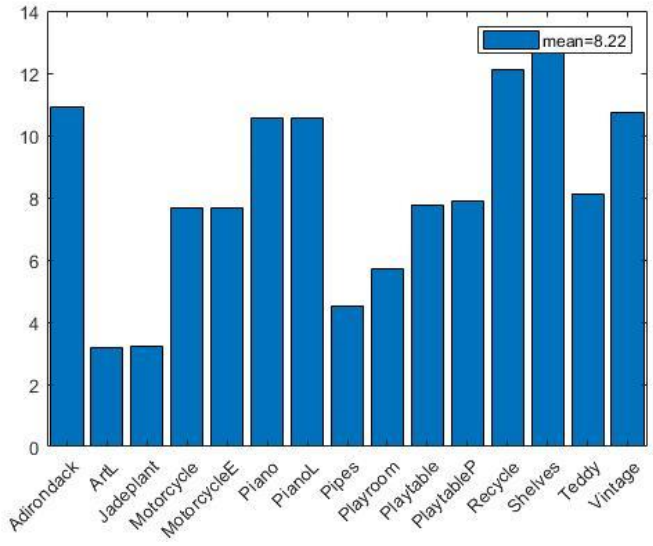


图 3.6 Middlebury2014 训练集非遮挡与遮挡区域比例统计。

基于卷积神经网络的立体匹配算法，需要充分地构建数据样本来训练深度神经网络庞大的模型参数。受限于公开的立体匹配数据集的图像数量，通常会在原始数据集图像上进行密集采样来构建足够的学习样本。例如，论文[35][36]中，在 Middlebury 数据集上（在后续实验部分用到的数据集）采样了 3800 万个9×9的图像块样本来进行匹配代价学习训练。但是在公开的数据集中，遮挡区域像素与非遮挡区域像素之间的比例是失衡的。如图 3.6 所示，我们统计了 Middlebury 2014 训练集图像中非遮挡区域与遮挡区域像素数量的比值，总的比例值为 8.22，非遮挡区域像素数量远多于遮挡区域的像素数量。

通过在原始图像对上进行密集采样构建训练样本，就会导致训练样本中遮挡区域像素与非遮挡区域像素数量之间存在不平衡。当训练像素样本失衡时，会导致两个问题:1、训练不高效，大多数都是非遮挡区域的像素，贡献很少的有用学习信息；2、非遮挡区域图像像素会主导训练，导致退化的模型[50]。而容易发生误匹配的遮挡区

域是我们希望在立体匹配中着力解决的。实际上，论文[35][36]并没有在遮挡区域上构建图像样本来进行训练，从而就很难在遮挡区域获得可靠的视差估计。而目前基于卷积神经网络的立体匹配的算法都没有考虑到在训练中存在的失衡问题。本文受到Focal Loss[50]的启发，针对存在的失衡问题，提出了回归聚焦损失函数来进行端到端的立体匹配训练。

Focal Loss[50]在处理分类问题中训练样本不平衡问题的主要思想是在训练过程中自适应地调整样本损失，降低分类较好样本的损失，使模型聚焦在难处理的样本上。基于 Focal Loss[50]的启发，利用相似的思想将其运用在回归问题上。我们在 L1 损失的基础上，提出了一种回归聚焦损失来实现困难样本发掘，使模型训练聚焦在难处理的样本上，避免模型退化。

在回归问题中，通常会使用 L1 损失作为损失函数，对应在立体匹配问题中，有以下形式，

$$L1=\left\| d-\hat{d} \right\|_1 \tag{3-3}$$

d 和 $\hat{d}$ 分别表示实际视差和预测视差。$L1$ 计算预测视差与实际视差之间的 L1 范数，刻画了视差估计的好坏程度。实际上，$L1$ 也是立体匹配算法的评价指标之一，即平均误差，$L1$ 越小表示视差估计越好。

在立体匹配中遮挡区域与非遮挡区域像素数量失衡时，非遮挡区域像素会主导 L1 损失和训练梯度，从而使模型倾向于非遮挡区域像素优化，使其转化为估计良好的样本，而难估计的遮挡区域像素并没有得到充分优化。本文提出的回归聚焦损失函数将降低估价良好的样本的损失权重，使得模型专注于训练难估计像素样本。由 $L1$ 改写后的回归聚焦损失函数记为 $RFL(L1)$，表达式如下：

$$RFL(L1)=(1-e^{-L1})^{\gamma}L1 \tag{3-4}$$

$(1-e^{-L1})^{\gamma}$ 作为调节因子乘到 L1 损失上，其中 $\gamma>0$。如图 3.7 所示，RFL 关于 $L1$ 在不同 γ 下的变化曲线，其中 γ=0 时就是 L1 损失。

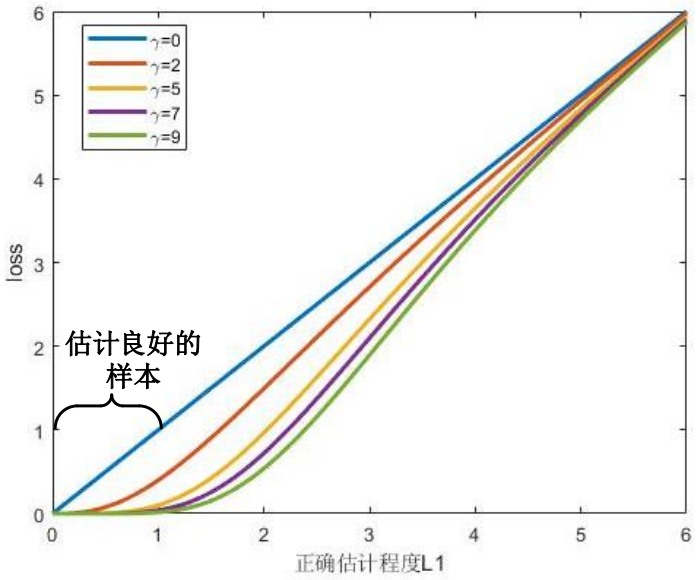


图 3.7 RFL 关于 $L1$ 在不同 γ 下的曲线。

从图 3.7 可以观察到，当一个像素样本被错误估计时，那么 $L1$ 很大，调制因子 $(1-e^{-L1})^{\gamma}$ 接近 1，损失不被影响。当 $L1$ 较小时，$(1-e^{-L_1})^{\gamma}$ 接近于 0，那么估计良好的像素样本的权值就被调低了，并且估计越好，损失抑制越大。其中 γ 的作用同 Focal Loss 中类似，用来调节估计良好的样本调低权值的比例。

回归聚焦损失函数可以根据与 ground truth 之间的估计差异，自适应地弱化估计良好像素样本在总损失中的贡献，使模型聚焦在那些难估计处理的像素样本上，从而防止模型退化。

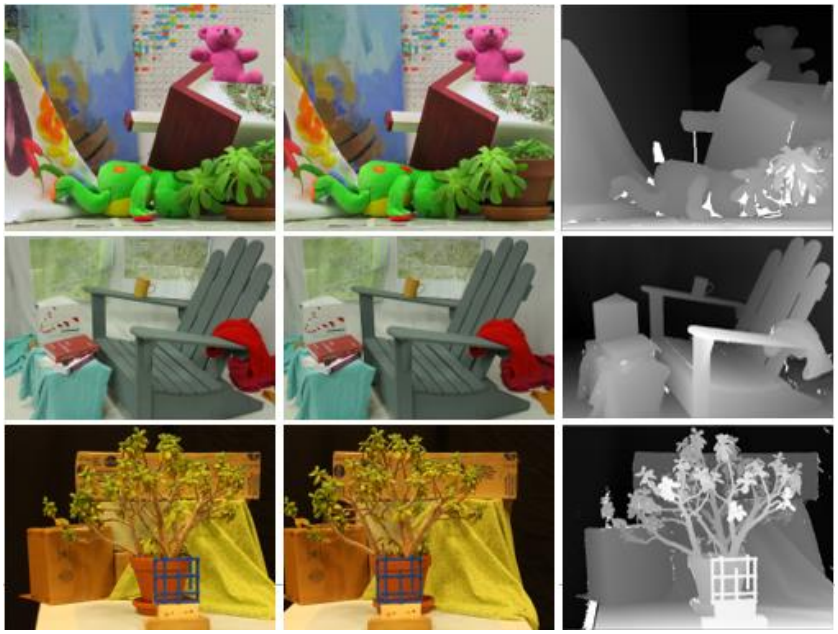
本文利用回归聚焦损失来对提出的端到端立体匹配网络进行监督训练，此时立体匹配网络总的损失为：

$$Loss=\frac{1}{N}\sum_{n=1}^{N}RFL\left(\left\|d_n-\hat{d}_n\right\|_1\right) \tag{3-5}$$

其中 N 表示所有带 ground truth 的像素数目，d_n 和 $\hat{d}_n$ 分别表示实际视差和估计视差，RFL 的计算方式见(3-4)式。

3.3 实验数据集及评价指标

实验所用的数据集为 Middlebury 2014。Middlebury 2014 由 Scharstein[51]等人利用结构光系统在室内场景捕获的高分辨率立体图像数据集，并且该数据集还包含高精度的 ground truth 视差值，图 3.8 列举了部分图像样本。另外，Scharstein 等人还建立了 Middlebury stereo benchmark[52]，研究者们可以将自己的算法上传到 benchmark 上与 state-of-the-art 算法进行对比，Middlebury stereo benchmark 是目前比较权威的立体匹配算法评估平台。



左视图　　右视图　　ground truth

图 3.8 Middlebury2014 训练集部分图像样本。

Middlebury 2014 数据集总共包含 30 对立体图像对，其中 15 对为训练集，另外 15 对为测试集，并且提供了遮挡区域掩膜。为了防止过度地参数拟合，平台仅提供了训练集的 ground truth，且对于每种算法仅有一次上传测试集结果的机会。平台考虑到某些算法的应用限制，提供了全分辨率、半分辨率和 1/4 分辨率三种版本。全分辨率的上限为3000×2000，最大视差为 800；半分辨率的上限1500×1000，最大视差为 400；1/4 分辨率的上限为750×500，最大视差为 200。本文利用半分辨率数据集

进行实验，具体参数如表 3-2 所示。

表 3-2 Middlebury 2014 半分辨率数据集参数

训练集	分辨率	最大视差	测试集	分辨率	最大视差
Adirondack	1436x992	145	Australia	1430x984	145
ArtL	694x554	128	AustraliaP	1434x984	145
Jadeplant	1318x994	320	Bicycle2	1426x976	125
Motorcycle	1482x994	140	Classroom2	1500x948	305
MotorcycleE	1482x994	140	Classroom2E	1500x948	305
Piano	1414x962	130	Computer	664x554	128
PianoL	1414x962	130	Crusade	1440x948	400
Pipes	1470x970	150	CrusadeP	1440x958	400
Playroom	1398x952	165	Djembe	1438x988	160
Playtable	1360x926	145	DjembeL	1438x988	160
PlaytableP	1362x924	145	Hoops	1442x996	205
Recycle	1440x972	130	Livingroom	1484x992	160
Shelves	1476x994	120	Newkuba	1402x974	285
Teddy	900x750	128	Plants	1420x992	160
Vintage	1444x960	380	Staircase	1380x936	225

此外，Middlebury stereo benchmark 还提供了 Scharstein[12]等人利用早期的结构光系统拍摄的立体图像数据集 Middlebury 2001、2003、2005、2006，共包含 35 对带有 ground truth 的立体图像。在后续的实验中，我们会用这些额外的数据集进行回归聚焦损失函数实验验证和网络结构消融实验验证。

Middlebury stereo benchmark 提供的评价指标主要有坏点率和平均误差。

坏点率：对于估计得到的视差与 ground truth 范围之差大于某个范围就被视为匹配错误的点，反之则视为匹配正确。计算方式如下，

$$\text{坏点率}=\frac{1}{N}\sum_{n=1}^{N}\left[\left|d_n-\hat{d}_n\right|>\sigma\right] \tag{3-6}$$

其中 N 表示所有带 ground truth 的像素数目，d_n 和 $\hat{d}_n$ 分别表示实际视差和估计视差。$[\bullet]$ 为判断操作，满足条件则为 1，否则为 0。σ 为坏点阈值，Middlebury benchmark 默认取值为 2。坏点率的单位为百分比，后续实验记阈值为 2 的坏点率为 bad2.0。

平均误差：计算估计得到的视差与 ground truth 之间的平均误差。计算方式如下，

$$\text{平均误差}=\frac{1}{N}\sum_{n=1}^{N}\left|d_n-\hat{d}_n\right| \tag{3-7}$$

其中 N 表示所有带 ground truth 的像素数目，d_n 和 $\hat{d}_n$ 分别表示实际视差和预测视差。平均误差的单位为像素，后续实验记平均误差为 avgerr。

对于立体匹配算法，坏点率和平均误差均越小越好。另外，由于 Middlebury 提供了遮挡区域掩膜，故可以分析在不考虑遮挡区域和考虑全部像素情形下的算法性能。在后续的实验结果中，我们记 nonocc 表示不考虑遮挡区域的情形，all 表示考虑全部像素的情形。

3.4 实验设置

本章实验主要分为三个部分，分别是回归聚焦损失函数实验、Middlebury stereo benchmark 实验和网络结构消融实验。由于 Middlebury 2014 数据集仅公开了训练集的 ground truth，且 Middlebury stereo benchmark 仅允许上传一次测试集结果。在回归聚焦损失函数实验和网络结构消融实验中，本文选择 Middlebury 2014 训练集的 15 对图像和 Middlebury 2001、2003、2005、2006 中的 30 对图像共 45 对图像进行实验，其中 80%（36 对）的图像对进行训练，20%用于测试验证（9 对，Jadeplant、Recycle、Playtable、Books、Dolls、Tsukuba、Venus、Cones 和 Teddy）。在 Middlebury stereo benchmark 实验中，则是利用 Middlebury 2014 训练集的 15 对图像进行训练，然后将 15 对测试集的实验结果上传到 benchmark 进行算法对比。

表 3-3 本文实验的软硬件运行环境配置

软硬件名称	配置
CPU	Intel Core i7-6700k，主频 4GHz
内存	16GB
GPU	Nvidia GeForce GTX 1080 ti，11G 显存
操作系统	Windows 10
CUDA	CUDA 9.0
深度学习框架	Pytorch 0.4.1

本文实验的软硬件运行环境如表 3-3 所示，实验利用 Pytorch 深度学习框架进行实现。本文利用 Adam[53]优化方法进行端到端的训练，在训练阶段，对于表 3-1 中的网络结构参数，设置为 $H=320$，$W=640$，$D=400$，$C=32$，通过在原始图像对上进行密集随机采样的方式构建训练样本。利用 Kaiming 高斯初始化方法[54]对网络参数初始化，将学习率恒定设为 0.001 训练 10000 个周期。

3.5 实验结果比较与分析

3.5.1 回归聚焦损失函数实验

（1）定量分析

回归聚焦损失函数引入了一个新的超参，聚焦参数 γ，其控制着调制项的强度。当 γ=0 时，回归聚焦损失函数就是 L1 损失。本小节利用第三章提出的端到端立体匹配网络，分析回归聚焦损失在不同 γ 取值下的性能。

我们分别对 γ 取 0，2，5，7 和 9 时进行了测试，整体的实验结果如表 3-4 所示，表 3-5 表示在考虑全部像素的情况下不同 γ 取值的 bad2.0 结果。从表 3-4 和表 3-5 中可以观察到，当 $\gamma=5$ 时性能是最优的，在绝大多数验证样本上的结果都是最好的。我们提出的回归聚焦失相较 L1 损失（即 γ 为 0 时），使得模型在不考虑遮挡区域和考虑全部像素的情况下，评价指标 bad2.0 和 avgerr 都有一定的下降，说明了本文提

出的回归的聚焦损失函数的有效性。考虑全部像素的情况下，设γ=5时，bad2.0 下降了 1.8%，比在不考虑遮挡区域时下降的 0.6%多了 1.2%，说明回归聚焦损失函数提高了深度网络模型对于遮挡区域的处理能力。

表 3-4 不同γ取值的 9 对验证图像上的实验结果

γ	nonocc		all	
	bad 2.0(%)	avgerr(pixels)	bad 2.0(%)	avgerr(pixels)
0	9.27	4.15	15.2	7.28
2	8.85	3.97	14.5	6.96
5	**8.23**	**3.55**	**13.4**	**6.23**
7	8.60	3.84	14.1	6.74
9	8.91	3.98	14.6	6.98

表 3-5 考虑全部像素时不同γ取值在 9 对验证图像上的 bad2.0(%)结果

图像对	γ=0	γ=2	γ=5	γ=7	γ=9
Jadeplant	33.4	32.9	**31.7**	32.1	32.7
Recycle	11.7	11.1	**9.94**	10.5	10.9
Playtable	12.1	10.7	**9.48**	10.4	11.4
Books	11.5	10.6	10.4	**9.93**	10.8
Dolls	18.2	17.3	16.5	**16.3**	17.4
Tsukuba	15.4	14.8	**13.7**	14.9	14.6
Venus	12.2	11.5	**10.1**	11.4	11.8
Cones	12.1	11.6	**9.83**	11.2	11.5
Teddy	10.6	9.87	**9.28**	9.95	10.1
Average	15.2	14.5	**13.4**	14.1	14.6

（2）定性分析

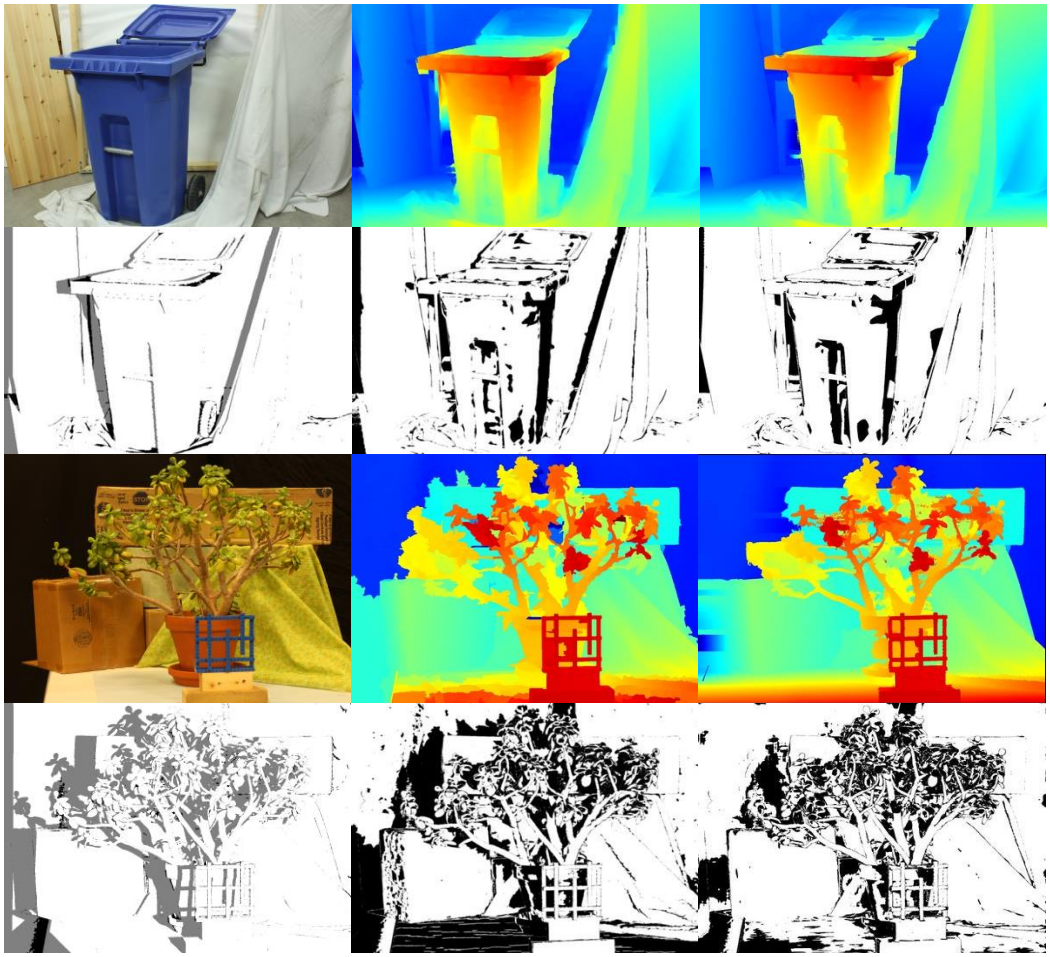


图 3.9　L1 损失($\gamma=0$)和回归聚焦损失($\gamma=5$)在 Recycle 和 Jadeplant 上的实验对比。(a)是左视点图像和对应的遮挡掩膜，(b)是 L1 损失生成的视差图和 bad2.0 误差图，(c)是回归聚焦损失生成的视差图和 bad2.0 误差图。

如图 3.9 所示的回归聚焦损失和 L1 损失在 Recycle 和 Jadeplant 图像对上的实验对比。上面两行对应 Recycle 实验，下面两行对应着 Jadeplant 实验。对于遮挡区域掩膜图像，其中灰色的区域就代表遮挡像素区域；对于 bad2.0 误差图，其中黑色的像素就代表误匹配像素，即估计视差与 ground truth 误差超过 2 个像素。在后续实验中的掩膜图像和误差图像表达相同的意义，就不再依次介绍。从图 3.9 的结果中可以看到，回归聚焦损失在遮挡区域的误匹配点有明显减少，而且整体图像的 bad2.0 误

差较 L1 损失表现更好，生成的视差图在视差不连续区域更加平滑，主观质量也有提高。上述实验结果表明了提出的回归聚焦损失函数的有效性。

3.5.2 Middlebury stereo benchmark 实验结果

在上一小节中，我们进行了回归聚焦损失函数实验，在验证实验中取 γ 为 5 是最优的。这一小节我们对提出的利用层次化上下文信息的端到端立体匹配算法进行实验，将回归聚焦损失函数中的 γ 设为 5，在 15 对 Middlebury 2014 训练集图像上进行网络训练。并将 15 对测试集图像结果上传到 Middlebury stereo benchmark 进行算法对比，在 Middlebury stereo benchmark 中，本文算法命名为 EHCI_net（Exploiting hierarchical context information）。

（1）定量分析

表 3-6 显示了本文算法同算法 SGM[17]、TMAP[28]、MC-CNN-fst[36]、MC-CNN-acrt[36]和 iResNet[45]之间的实验结果对比。从表中可以看到，在最重要的指标上，即考虑全部像素的 bad2.0 坏点率，本文算法 EHCI_net 是最佳的。在不考虑遮挡区域的情况下，MC-CNN-acrt[36]性能是最佳的，在坏点率和平均误差两个指标上都是最低的，本文算法 EHCI_net 则仅次于 MC-CNN-acrt[36]，在坏点率和平均误差两个指标上都是次最优的，bad2.0 指标与 MC-CNN-acrt[36]相差 1.39%，avgerr 指标相差 0.47。在考虑所有像素时，本文算法 EHCI_net 则是性能最佳的，在坏点率 bad2.0 指标上是最低的 15.5%，相较 MC-CNN-acrt[36]提升了 3.6%；在平均误差 avgerr 指标上则是次最低的 7.47，相较 MC-CNN-acrt[36]提升了 10.43。表 3-6 的数据表明的本文提出的利用层次化上下文信息的端到端立体匹配网络和回归聚焦损失函数的有效性，且本文算法在难处理的遮挡区域表现更好。印证了本文考虑利用上下文信息对难处理区域进行视差推断和利用回归聚焦损失函数防止模型在难处理区域退化思想的有效性。表 3-7 表示的是在考虑所有像素的情况下，不同算法在 Middlebury 2014 的 15 对测试图像上的 bad2.0 指标对比。从中可以观察到，本文算法 EHCI_net 在绝大多数测试样本中的结果都是最好的。

表 3-6 本文算法在 Middlebury stereo benchmark 上的结果对比

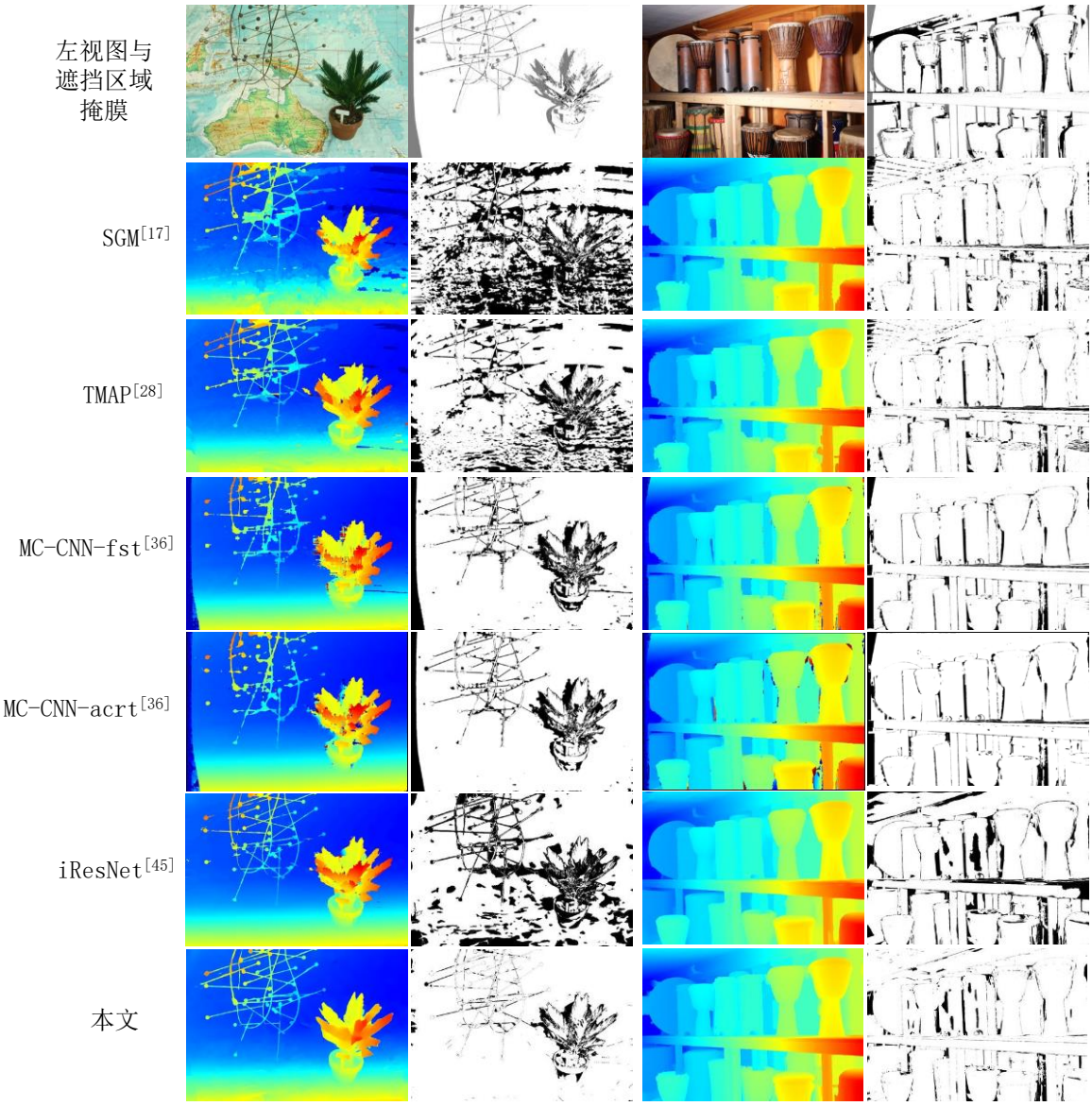
算法	nonocc		all	
	bad 2.0(%)	avgerr(pixels)	bad 2.0(%)	avgerr(pixels)
SGM[17]	18.4	5.32	25.7	9.27
TMAP[28]	16.9	4.75	24.6	10.1
MC-CNN-fst[36]	9.47	4.37	20.6	19.3
MC-CNN-acrt[36]	**8.08**	**3.82**	19.1	17.9
iResNet[45]	24.8	4.51	31.7	**6.56**
本文(EHCI_net)	9.47	4.29	**15.5**	7.47

表 3-7 考虑全部像素时不同算法在 Middlebury stereo benchmark 上的 bad2.0(%)结果

图像对	SGM[17]	TMAP[28]	MC-CNN-fst[36]	MC-CNN-acrt[36]	iResNet[45]	本文
Austr	37.8	23.4	13.8	12.1	25.1	**6.16**
AustrP	14.2	8.58	11.5	10.9	13.1	**6.19**
Bicyc2	19.3	13.3	14.2	**12.4**	19.1	15.3
Class	25.8	21.6	17.9	**15.6**	20.0	33.8
ClassE	39.3	37.8	28.4	23.3	32.6	**11.3**
Compu	31.8	25.2	24.3	21.4	25.2	**13.9**
Crusa	36.7	38.6	23.7	24.4	39.4	**10.7**
CrusaP	30.3	32.6	23.8	25.1	37.3	**10.6**
Djemb	13.3	8.11	8.56	8.39	15.3	**6.63**
DjembL	42.4	33.4	25.4	21.4	48.6	**11.5**
Hoops	43.5	36.5	33.6	30.7	49.2	**23.1**
Livgrm	28.7	27.5	24.5	22.3	28.4	**14.9**
Nkuba	27.3	25.8	25.3	23.3	34.5	**20.9**
Plants	30.7	26.6	20.4	**18.8**	51.0	25.7
Stairs	41.8	28.3	25.7	23.9	69.8	**19.1**
Average	28.8	24.6	20.6	19.1	31.7	**15.5**

（2）定性分析

图 3.10 表示的是不同算法在 Australia、Djembe、CrusadeP 和 Hoops 测试图像上的实验结果对比。从上到下，前 7 行对应着在 Australia 和 Djembe 测试样本上的实验，后 7 行对应着在的 CrusadeP 和 Hoops 测试样本上的实验。以前 7 行图像为例，左边的两列对应 Australia 样本，右边两列对应 Djembe 样本，第 1 行是左视点图像和和遮挡区域掩膜图像，第 2 行至第 7 行依次对应着 SGM[17]、TMAP[28]、MC-CNN-fst[36]、MC-CNN-acrt[36]和 iResNet[45]和本文算法 EHCI_net 的实验结果，左边为视差图像，右边为考虑全部像素情形下的 bad2.0 误差图。从这 4 组测试实验结果中可以



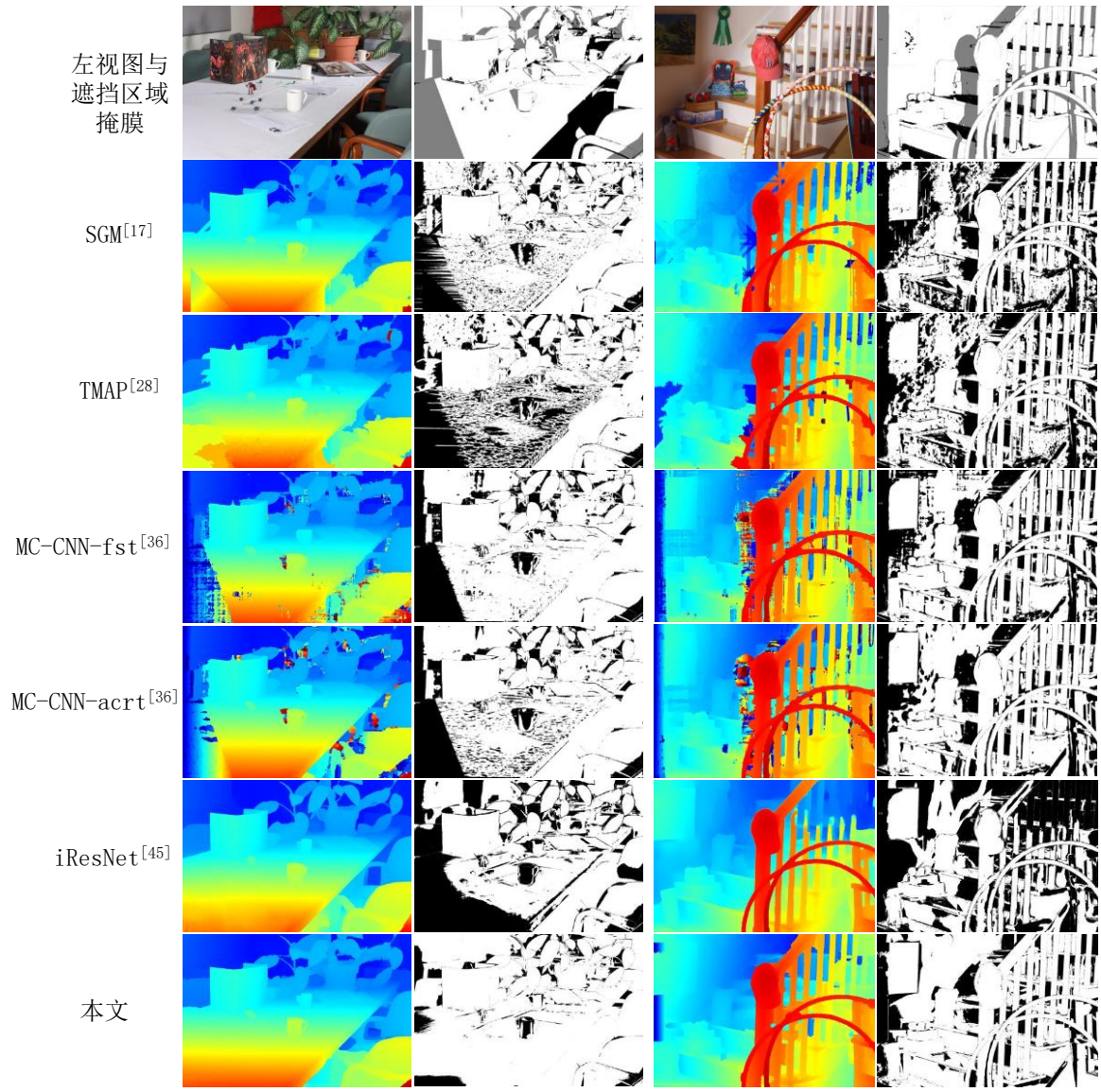


图 3.10 不同算法在 Middlebury 2014 测试集图像对 Australia、Djembe、CrusadeP 和 Hoops 上的实验结果对比。

看到，iResNet[45]生成的视差图的视觉质量最好，在视差不连续的区域保留了细节，但是视差值却估计偏差较大。iResNet[45]单独设计利用彩色空间中特征一致性的后处理网络，使得视差图与彩色纹理图保持了不错的一致性。从图 3.10 中可以明显观察到，本文算法 EHCI_net 在取得不错视觉质量的同时，也保证了视差估计的精度。尤其是对于存在遮挡的难点区域，本文算法较 SGM[17]、TMAP[28]、MC-CNN-fst[36]、MC-CNN-acrt[36]和 iResNet[45]等对比算法有着明显的优势。

3.5.3 网络结构消融实验

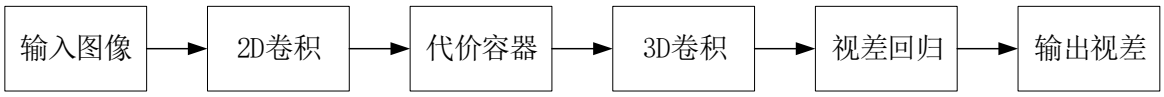


图 3.11 基干网络 GC-Net[44]构成模块示意图

在 3.2 节，我们介绍到本文提出的利用层次化上下文信息的端到端网络以 GC-Net[44]为基干网络，如图 3.11 所示。这一小节我们将对设计的空间金字塔池化模块和编码-解码结构 3D 卷积模块进行消融实验。本文依次训练了四种网络结构，分别为基干网络 GC-Net、基干网络+空间金字塔池化、基干网络+编码-解码结构 3D 卷积和本文提出网络。将四种网络训练的模型对 9 对验证图像进行了测试。

（1）定量分析

表 3-8 不同网络结构的消融实验结果

网络结构	nonocc		all	
	bad 2.0(%)	avgerr (pixels)	bad 2.0(%)	avgerr (pixels)
基干网络 GC-Net[44]	10.3	4.62	16.9	8.11
基干网络+空间金字塔池化	8.61	3.86	14.1	6.77
基干网络+编码-解码结构 3D 卷积	9.15	4.39	15.0	7.20
本文网络	**8.17**	**3.67**	**13.4**	**6.43**

不同网络结构的消融实验结果如表 3-8 所示，对于不考虑遮挡区域情形下的评价指标，在基干网络 GC-Net 中加入空间金字塔的池化模块，bad2.0 下降了 1.69%，avgerrr 下降了 0.76；加入编码-解码结构 3D 卷积，bad2.0 下降了 1.15%，avgerr 下降了 0.23；本文网络相对 GC-Net，bad2.0 下降了 2.13%，avgerr 下降了 0.95。对于考虑全部像素情形时，加入的空间金字塔池化模块和编码-解码结构 3D 卷积模块，bad2.0 和 avgerr 指标性能均有提升，并且比在不考虑遮挡情形的提升要大。表 3-9 是四种网络结构在考虑全部像素情形下，在 9 对测试图像上的 bad2.0 结果。上述实验

结果说明了利用空间金字塔模块和编码-解码结构 3D 卷积模块进行层次化上下文学习的有效性，对存在遮挡的难点区域有更好的处理效果。

表 3-9 考虑全部像素时不同网络结构在 9 对验证图像上的 bad2.0(%)结果

网络 / 图像对	基干网络 GC-Net[44]	基干网络+空间金字塔池化	基干网络+编码-解码 3D 卷积	本文网络
Jadeplant	35.1	32.4	33.4	**31.7**
Recycle	13.9	10.3	11.7	**9.94**
Playtable	13.5	10.5	11.3	**9.48**
Books	14.1	10.4	12.8	**10.1**
Dolls	20.3	16.5	17.8	**16.3**
Tsukuba	16.8	14.6	15.2	**13.7**
Venus	13.4	11.2	11.6	**10.1**
Cones	12.3	10.2	10.4	**9.83**
Tedd	12.6	10.6	11.1	**9.28**
Average	16.9	14.1	15.0	**13.4**

（2）定性分析

图 3-12 表示的是四种网络结构在 Recycle 和 Jadeplant 上的实验结果，第 1、2 列和第 3、4 列分别对应着 Recycle 和 Jadeplant 上的实验结果。 从图 3-13 中可以直观看到，加入空间金塔池化模块和编码-解码结构 3D 卷积模块，生成的视差图质量均有提高，从对应的 bad2.0 误差图中可以看到，误匹配像素较基干网络有明显减少，且遮挡区域的视差估计质量也有提高。

图 3-13 显示的是局部区域放大实验结果。在第一排 Recycle 图像对中，上面的一小行对应的是四种网络结构对于红色框区域的处理结果，下面的一小行对应的是绿色框区域的处理结果，红色框和绿色框所对应着的是立体图像对中的遮挡区域以及视差不连续区域。从右边的放大结果中可以看到，加入空间金字塔池化和编码-解码结构 3D 卷积的网络模型生成的视差图，在窗帘和垃圾桶箱边缘遮挡区域，相对基干

网络的结果更加平滑，主观质量更好。而从旁边的 bad2.0 误差图可以清楚地看到误匹配点也更少。从第二排的 Jadeplant 图像对中，我们也有相同的观察结果。

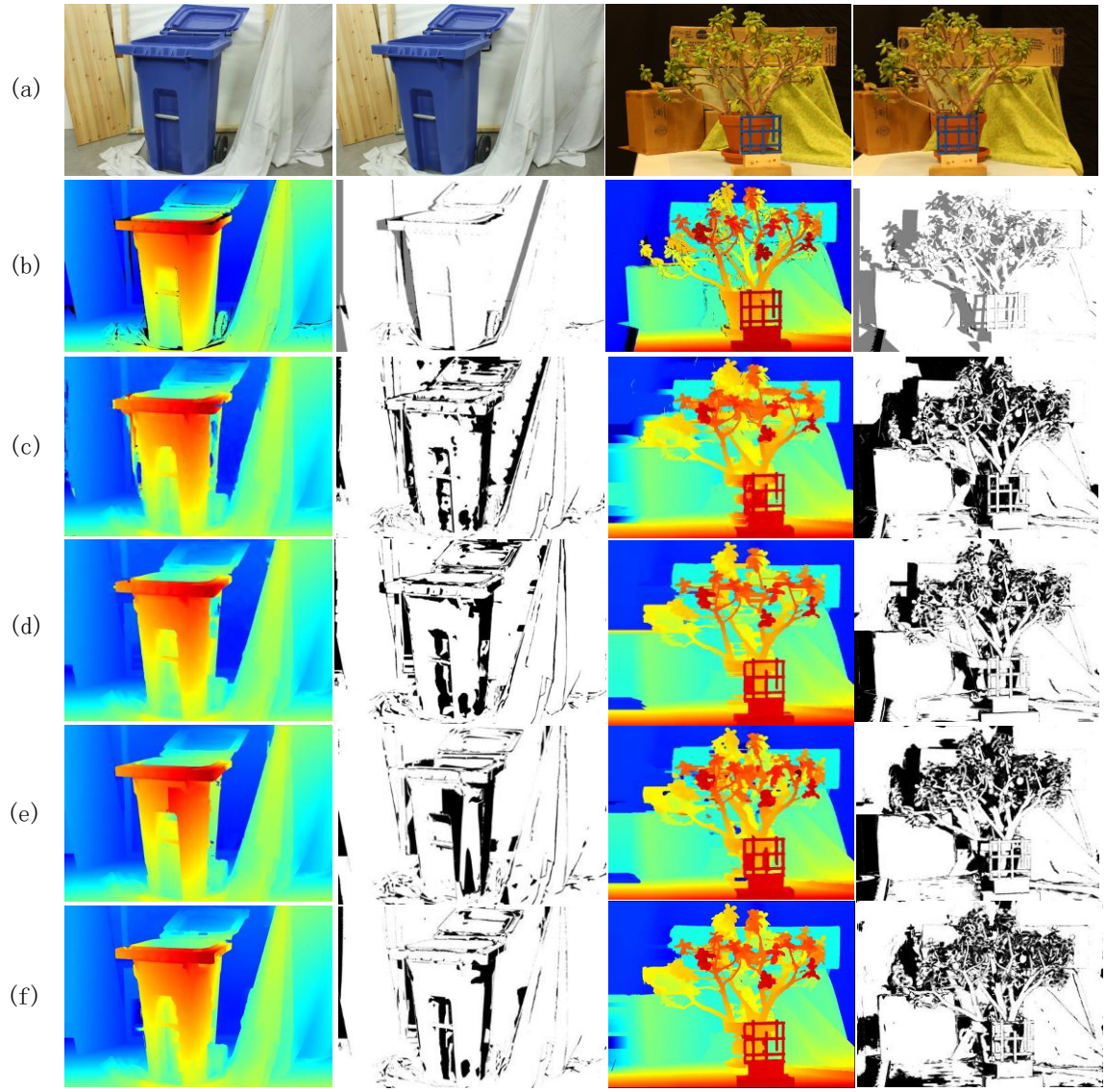


图 3-12 四种网络结构在 Recycle 和 Jadeplant 上的实验结果。(a)是左右视点图像对，(b)是 ground truth 和遮挡区域掩膜，(c)是基干网络 GC-Net[44]的实验结果，(d)是基干网络+空间金字塔池化的实验结果，(e)是基干网络+编码-解码结构 3D 卷积的实验结果，(f)是本文网络的实验结果。

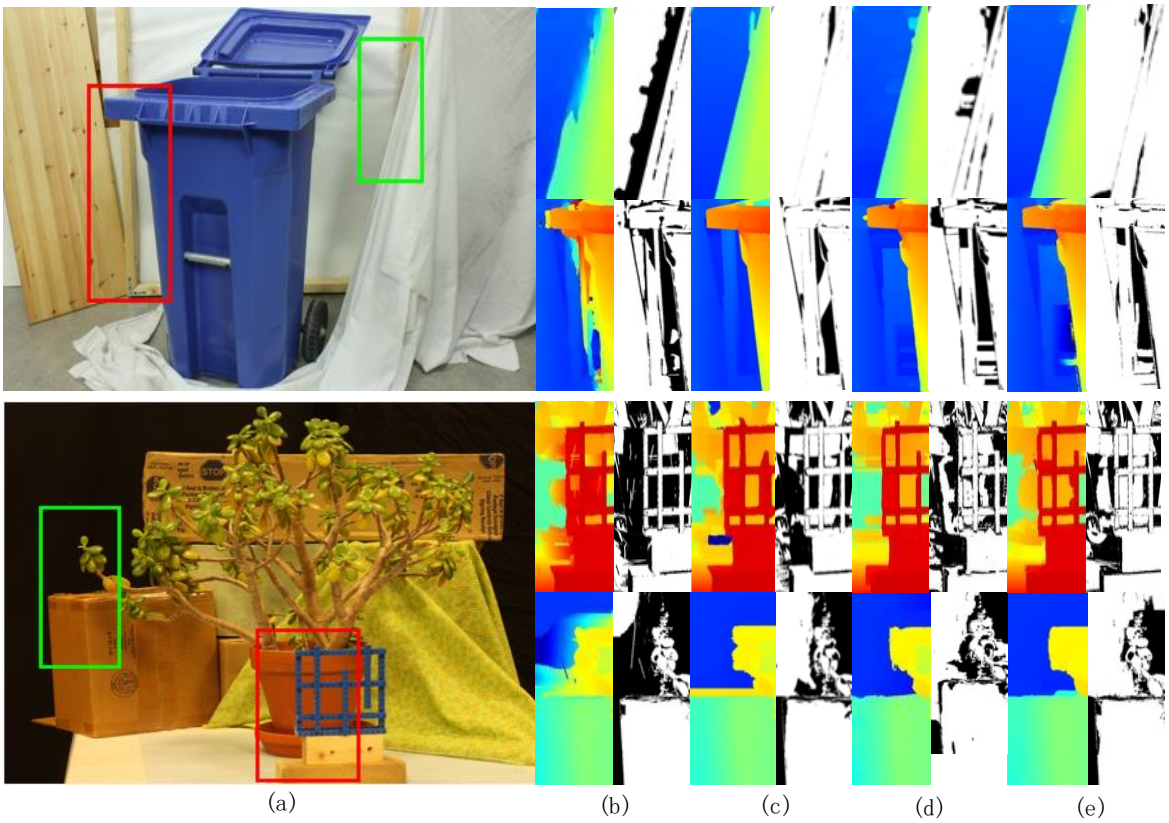


图 3-13 四种网络结构在 Recycle 和 Jadeplant 上的局部区域放大的实验结果。(a)是左视点图像，(b)是基干网络 GC-Net[44]的实验结果，(c)是基干网络+空间金字塔池化的实验结果，(d)是基干网络+编码-解码结构 3D 卷积的实验结果，(e)是本文网络的实验结果。

3.6 本章小结

本章首先进行了算法动机分析，然后详细介绍了提出的利用层次化上下文信息的端到端立体匹配算法。该算法以在立体匹配中充分利用上下文信息为出发点，从语义分割中提取上下文信息得到两条线索。设计空间金字塔池化和编码-解码结构 3D 卷积，将其嵌入端到端的立体匹配网络实现层次化的上下文提取，在不需要后处理的情况下，直接回归输出精细化的视差图。其次，本章分析了基于卷积神经网络的立体匹配方法中存在的遮挡区域训练像素样本不平衡问题，并针对此问题提出了回归聚焦损失函数来进行网络的监督训练。回归聚焦损失函数实验和网络结构消融实验的结

果表明了设计回归聚焦损失防止模型在遮挡区域退化和设计空间金字塔池化模块和编码-解码结构 3D 卷积模块提层次化的上下文信息对难处理区域进行视差推断的有效性。在 Middlebury stereo benchmark 上的结果表明了本文提出的利用上下文信息的端到端立体匹配算法在 bad2.0 性能指标上已经超过了当前的经典算法。

4 总结与展望

4.1 总结

本文基于卷积神经网络进行立体匹配研究，着力于解决立体匹配中存在的遮挡、反射和弱纹理等易发生误匹配的难点区域。全文的工作可以总结为以下两个方面：

（1）对于立体匹配中存在的难点区域，利用像素的上下文信息进行视差推断是一个重要思路。本文提出了一个利用层次化上下文信息的端到端立体匹配算法。设计空间金字塔池化模块，在不同尺度和位置上提取层次化的上下文信息以适应不同的匹配区域，并由此构建匹配代价容器。接着设计了编码-解码结构 3D 卷积在匹配代价容器上优化学习得到 DSI，编码-解码结构 3D 卷积通过跳跃连接可以聚合多尺度的上下文信息。最后通过视差回归直接输出精细化的视差。实验结果表明了本文设计的空间金字塔池化和编码-解码结构 3D 卷积方法的有效性，对于立体匹配中存在的难点区域有着比较好的处理效果，尤其是对于难处理的遮挡区域。

（2）本文分析了基于深度学习方法中存在的遮挡区域像素和非遮挡区域像素训练样本不平衡问题，针对此问题，提出了回归聚焦损失函数。回归聚焦损失函数可以在训练过程中根据样本的估计好坏程度自适应地调整损失，抑制估计良好样本的损失，使模型聚焦在难估计处理的样本上，防止模型退化。实验结果表明了提出的回归聚焦损失函数的有效性，能有效提高网络在遮挡区域的估计精度。

4.2 展望

本文利用卷积神经网络进行了立体匹配研究，设计了利用层次化上下文信息的端到端立体匹配算法和回归聚焦损失函数。针对本文工作的不足，未来还有可以着力于以下几个方面研究：

（1）在特征提取阶段，可以考虑利用膨胀卷积操作扩大网络的感受野，提取更大尺度的特征信息。

（2）在根据代价容器学习优化匹配代价阶段，本文设计了编码-解码结构 3D 卷积模块。为了在此阶段学习到更多的上下文信息，可以考虑使用多层级联的编码-解码结构，对每个阶段的输出进行加权得到最终的视差图。

（3）本文并没有专门设计后处理的方法来进行视差图优化处理，后续研究可以考虑将卷积神经网络与条件随机场结合起来构建端到端的算法，进一步提高算法的视差估计精度。

致 谢

参考文献

[1] Andrea C, Giancola S, Mainetti G, and Sala R. A metrological characterization of the Kinect V2 time-of-flight camera[J]. Robotics and Autonomous Systems,2016,75:584-594.

[2] Zhang Z. Microsoft Kinect Sensor and Its Effect[J]. IEEE Transactions on Multimedia,2012,19(2):4-10

[3] Hamed S, Lefloch D, and Kolb A. Kinect range sensing: Structured-light versus Time-of-Flight Kinect [J]. Computer Vision and Image Understanding ,2015,139: 1-20.

[4] Menze M and Geiger A. Object scene flow for autonomous vehicles[C]. IEEE Conference on Computer Vision and Pattern Recognition,2015:3061-3070.

[5] Barry A and Tedrake R. Pushbroom stereo for high-speed navigation in cluttered environments[C]. IEEE International Conference on Robotics and Automation, 2015: 3046-3052.

[6] Zhang C, Li Z, Cheng Y, Cai R, Chao H, and Rui Y. Meshstereo: A global stereo model with mesh alignment regularization for view interpolation[C]. In Proceedings of the IEEE International Conference on Computer Vision, 2015:2057–2065.

[7] Raul M and Tardós J. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras[J]. IEEE Transactions on Robotics,2017,33(5):1255-1262.

[8] Wang F, Jiang M, Qian C, Yang S, Li C, Zhang H and Tang X. Residual attention network for image classification[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2017:3156-3164.

[9] Lin T, Dollár P, Girshick R, He K, Hariharan B and Belongie S. Feature pyramid networks for object detection. The IEEE Conference on Computer Vision and Pattern Recognition ,2017:2117-2125.

[10] Gidaris S and Komodakis N. Detect, replace, refine: Deep structured prediction for pixel wise labeling[C]. The IEEE Conference on Computer Vision and Pattern

Recognition,2017:5248-5257.

[11] Chen L ,Papandreou G, Kokkinos , Murphy K, Yuille A . DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018,40(4):834-848.

[12] Scharstein D and Szeliski R. A Taxonomy and Evaluation of Dense Two-Frame Stereo Correspondence Algorithms[J]. International Journal of Computer Vision, 2002, 47(1):7–42.

[13] Martin J and Crowley J. Experimental comparison of correlation techniques[C]. In Proc. Int. Conf. on Intelligent Autonomous Systems, 1995:86–93.

[14] Zabih R and Woodfill J.Non-Parametric Local Transforms for Computing Visual Correspondance[C]. European Conference Computer Vision,1994:151-158.

[15] Mei X, Sun X , Zhou M , Jiao S, Wang H , Zhang X. On building an accurate stereo matching system on graphics hardware[C]. IEEE International Conference on Computer Vision Workshops , 2011:467-474.

[16] Tombari F, Stefano L, Mattoccia S, Galanti A. Performance evaluation of robust matching measures[C]. International Conference on Computer Vision Theory and Applications, 2008:473-478.

[17] Hirschmuller H. Stereo vision in structured environments by consistent semi-global matching[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(2):328-341.

[18] Hirschmuller H and Scharstein D. Evaluation of cost functions for stereo matching[C]. IEEE Conference on Computer Vision and Pattern Recognition ,2007:1-8.

[19]Zhang K, Lu J, and Lafruit G. Cross-Based Local Stereo Matching Using Orthogonal Integral Images[C]. IEEE Transactions on Circuits and Systems for Video Technology, 2009:1073-1079.

[20] Yang Q. A non-local cost aggregation method for stereo matching[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2012:1402-1409.

[21] Zhang K., Fang Y, Min D, Sun L, Yang S, Yan S and Tian Q. Cross-scale cost aggregation for stereo matching[C]. IEEE Conference on Computer Vision and Pattern Recognition , 2014:1590-1597.

[22]Xu Y, Wang D, Feng T, and Shum H. Stereo computation using radial adaptive windows[C]. International Conference on Pattern Recognition, 2002:595–598.

[23]Yoon K and Kweon S. Adaptive support-weight approach for correspondence search[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2006, 28(4):650–656.

[24]]Tombari F, Mattoccia S, and Stefano L. Segmentation based adaptive support for accurate stereo correspondence[C]. Pacific Rim Symp. Image Video Technol, 2007:427–438.

[25]Felzenszwalb P and Zabih R. Dynamic Programming and Graph Algorithms in Computer Vision[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2011,33(4):721–740.

[26]Felzenszwalb P and Huttenlocher D. Efcient Belief Propagation for Early Vision[J]. International Journal of Computer Vision, 2006,70(1):41–54.

[27]Yamaguchi K, Hazan T, McAllester D, and Urtasun R. Continuous Markov Random Fields for Robust Stereo Estimation in Computer Vision[C]. European Conference on Computer Vision, 2012:45–58.

[28] Psota T, Kowalczuk J, Mittek M, Perez L. MAP disparity estimation using hidden markov trees[C]. The IEEE International Conference on Computer Vision ,2015:2219-2227.

[29]Boykov Y, Veksler O, and Zabih R. Fast approximate energy minimization via graph cuts[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2001, 23(11):1222–1239.

[30] Taniai T, Matsushita Y, Sato Y, Naemura T. Continuous 3D label stereo matching using

local expansion moves[J]. IEEE transactions on Pattern Analysis and Machine Intelligence, 2018, 40(11):2725-39.

[31] Ohta Y and Kanade T. Stereo by Intra- and Inter-Scanline Search Using Dynamic Programming[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,1985, 1(2):139–154.

[32] Schonberger J, Sinha S and Pollefeys M. Learning to Fuse Proposals from Multiple Scanline Optimizations in Semi-Global Matching[C]. In Proceedings of the European Conference on Computer Vision , 2018:739-755.

[33]Ranftl R, Gehrig S, Pock T, and Bischof H. Pushing the limits of stereo using variational stereo estimation[C]. IEEE Intelligent Vehicles Symposium,2012:401–407.

[34]Kuschk G and Cremers D. Fast and Accurate Large-Scale Stereo Reconstruction Using Variational Methods[C]. IEEE International Conference on Computer Vision Workshops, 2013:700–707.

[35] Zbontar J and LeCun Y. Computing the stereo matching cost with a convolutional neural network[C]. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015:1592–1599.

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

[37] Luo W, Schwing A G, and Urtasun R. Efficient Deep Learning for Stereo Matching[C]. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition , 2016:5695-5703.

[38] Chen Z, Sun X, Wang L, Yu Y, and Huang C. A deep visual correspondence embedding model for stereo matching costs[C]. In Proceedings of the IEEE International Conference on Computer Vision, 2016:972–980.

[39] Shaked A and Wolf L . Improved Stereo Matching with Constant Highway Networks and Reflective Confidence Learning[C]. Proceedings of the IEEE Computer Society

Conference on Computer Vision and Pattern Recognition, 2017:35-46.

[40] Seki A and Pollefeys M. Patch Based Confidence Prediction for Dense Disparity Map[C]. British Machine Vision Conference, 2016,2(3):4.

[41]Long J, Shelhamer E, Darrell T. Fully convolutional networks for semantic segmentation[C]. IEEE Conference on Computer Vision and Pattern Recognition, 2015:3431-3440.

[42] Mayer N, Ilg E, Hausser P, Fischer P, Cremers D, Dosovitskiy A, and Brox T. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation[C]. In The IEEE Conference on Computer Vision and Pattern Recognition, 2016:4040-4048.

[43] Pang J, Sun W, Ren S Q, Yang C, and Yan Q. Cascade residual learning: A two-stage convolutional neural network for stereo matching[C]. In The IEEE International Conference on Computer Vision, 2017:878-886.

[44] Kendall A, Martirosyan H, Dasgupta S, Henry P, Kennedy R, et al. End-to-end learning of geometry and context for deep stereo regression[C]. In The IEEE International Conference on Computer Vision, 2017:66-75.

[45] Liang Z, Feng Y, Guo Y, Liu H, Chen W, Qiao L, Zhou L, Zhang J. Learning for disparity estimation through feature constancy[C]. The IEEE Conference on Computer Vision and Pattern Recognition , 2018:2811-2820.

[46] He K, Zhang X ,Ren S, Sun J. Deep Residual Learning for Image Recognition[C]. The IEEE Conference on Computer Vision and Pattern Recognition , 2016:770-778.

[47] Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. arXiv preprint arXiv,2015:1502.03167.

[48] Zhao H, Shi J, Qi X, Wang X, and Jia J. Pyramid scene parsing network[C]. In The IEEE Conference on Computer Vision and Pattern Recognition, 2017:2881:2890.

[49] Bahdanau D, Cho K, and Bengio Y. Neural machine translation by jointly learning to

align and translate. arXiv preprint arXiv,2014:1409.0473.

[50] Lin T, Goyal P, Girshick R, He K, Dollar P， Focal Loss for Dense Object Detection[C]. The IEEE International Conference on Computer Vision , 2017: 2980-2988.

[51] Scharstein D, Hirschmüller H, Kitajima Y, Krathwohl G, Nesic N, Wang X, and Westling P. High-resolution stereo datasets with subpixel-accurate ground truth[C]. In German Conference on Pattern Recognition ,2014:31-42.

[52] http://vision.middlebury.edu/stereo/eval3/

[53] Kingma D, Ba J. Adam: A Method for Stochastic Optimization. arXiv preprint arXiv,2014:1412.6980.

[54] He K., Zhang X., Ren S and Sun J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C]. The IEEE International Conference on Computer Vision, 2015:1026-1034.