# 博士学位论文

# 多源多视的三维场景和物体重建

# 3D SCENE AND OBJECT RECONSTRUCTION FROM MULTIPLE SOURCES AND VIEWPOINTS

**谢浩哲**

哈尔滨工业大学

**2021 年 6 月**

国内图书分类号：TP391.41　　学校代码：10213
国际图书分类号：681.39　　密级：公开

**工学博士学位论文**

# 多源多视的三维场景和物体重建

**博士研究生**：谢浩哲

**导　　师**：佟晓筠教授

**副 导 师**：姚鸿勋教授

**申请学位**：工学博士

**学　　科**：计算机科学与技术

**所在单位**：计算学部

**答辩日期**：2021 年 6 月

**授予学位单位**：哈尔滨工业大学

Classified Index: TP391.41
U.D.C: 681.39

Dissertation for the Doctoral Degree in Engineering

# 3D SCENE AND OBJECT RECONSTRUCTION FROM MULTIPLE SOURCES AND VIEWPOINTS

**Candidate:** Xie Haozhe
**Supervisor:** Professor Tong Xiaojun
**Associate Supervisor:** Professor Yao Hongxun
**Academic Degree Applied for:** Doctor of Engineering
**Specialty:** Computer Science and Technology
**Affiliation:** Faculty of Computing
**Date of Defence:** June, 2021
**Degree-Conferring-Institution:** Harbin Institute of Technology

# 摘 要

赋予机器像人类一样感知三维世界的能力一直是人工智能领域的一个长期研究的问题。受到人类认知方式的启发，本文展开了多源多视的三维场景和物体重建的研究。所提出的方法从大量样本中学习形状先验，因此甚至可以从单视角的彩色或者深度图像推测某个物体完整的三维结构。随着输入视图数量的增多，三维重建结果可以被不断改善。三维重建有许多潜在的应用，例如计算机辅助设计、混合现实、自动驾驶和机器人等。

本文在分析了相关研究后发现，现有的三维重建方法主要存在三个问题：第一，这些方法需要扫描完整的物体才可以重建物体完整的三维结构，然而这在一些情况下是不可行的；第二，这些方法难以充分利用不同数据源和视角的数据，仅能从彩色或深度图像中重建三维结构，然而彩色图像的多视角特征匹配在弱纹理或重复纹理物体上会失败，深度图像也无法获取不发生反射物体的几何结构。第三，这些方法不考虑场景的语义信息，因此重建后的物体和背景融为一体，难以将物体从重建的场景中分离。本文针对上述问题，依次从单源单视三维物体重建、多源多视三维物体重建和多源多视三维场景重建三个层面展开研究。具体地，本文的研究内容和主要贡献分为以下三个方面：

首先，为了解决现有方法无法恢复物体不可见部分三维结构的问题，本文针对单目彩色相机、双目彩色相机和深度相机从单视角拍摄的图像提出了三种几何结构感知的单源单视三维物体重建方法，利用已知的颜色或空间信息以及学习的几何先验推断物体未知部分的三维结构。对于**单目彩色相机**，本文提出了基于几何先验的三维物体重建方法，从大规模三维数据集中学习几何先验，隐式地建立图像空间和三维模型空间的映射关系；对于**双目彩色相机**，本文提出了基于深度感知的三维物体重建方法，利用双目视图估计物体的深度图，在恢复物体完整三维结构时更好地保留物体几何结构的细节；对于**深度相机**，本文提出了基于网格化残差网络的三维物体重建方法，将 3D Grid 作为几何结构的中间表示，使得在计算时充分利用上下文信息，同时更好地保留了深度相机所捕获的几何结构。在 ShapeNet、Pix3D 和 KITTI 等数据集上的实验结果表明，所提出的这三个方法可以从单视角拍摄的图像中恢复某个物体的完整三维结构，其重建质量相比于现有方法有 3% 至 18% 不等的提升。

其次，为了解决现有方法无法充分利用不同数据源和视角数据的问题，本文提出了多尺度上下文感知融合的多源多视三维物体重建方法，通过在三维模型空间融合多个彩色相机和深度相机的重建结果，使得不同数据源和视角的信息得以相互补充。一方面，不同模态的数据对于不同材质的物体具有不同的鲁棒性：对于弱纹理、重复纹理的物体，多视角彩色图像难以恢复其三维结构；对于不发生反射的物体，深度图像也无法获取其几何信息。另一方面，不同的视角可以观察到物体的不同部件，而往往可见部件的重建结果优于不可见部件的重建结果。利用这两个特性，本文提出了多尺度上下文感知融合，对来自不同数据源和不同视角重建结果中的每个部件的重建质量进行评估，从中选取重建质量最佳的部件生成最终的重建结果。在 ShapeNet、Pix3D 和 Things 3D 数据集的实验结果表明，所提出的多尺度上下文感知融合不仅在重建质量上相比现有的方法有 4% 至 20% 不等的提升，而且拥有更好的可解释性。

最后，为了使得场景中的物体可以直接从重建的场景中分离，本文提出了基于场景语义感知的多源多视三维场景重建方法，通过在重建时对场景进行语义建模，实现在重建场景的同时恢复场景中每个物体完整的三维结构。为了实现场景语义感知，本文提出了基于局部特征记忆网络的视频物体分割方法，该方法将场景中的物体从图像序列中分离，并更好地区分具有相似外观的物体。为了重建场景和其中的物体，本文设计了基于场景语义建模的三维场景重建方法，该方法通过重建每个物体的完整三维结构，并估计物体的位置和位姿，从而完成对三维场景的重建。在 SUN3D 数据集和实地拍摄的视频上的实验结果表明，所提出的基于场景语义感知的多源多视三维场景重建方法对场景和其中的物体能取得比现有方法更好的重建结果。

通过上述研究，本文对三维场景和物体重建进行了深入的探索，并为真实场景的三维重建提供了切实可行的解决方案。本文从单源单视单物体三维重建问题出发，进而提出了针对多源多视单物体三维重建和多源多视多物体的三维重建方法。针对现有三维重建方法所存在的三个问题，本文所提出的方法在重建场景时对场景进行语义建模，使得场景中物体的三维结构可以被更完整地恢复和并更容易地分离；同时，该方法对弱纹理、重复纹理和不发生反射的物体更加鲁棒。

**关键词**：三维重建，三维物体重建，场景语义感知，多尺度上下文感知，几何结构感知

# Abstract

To endow machines with the ability to perceive the real-world in 3D representation as we do as humans is a fundamental and longstanding topic in artificial intelligence. Motivated by human cognition, the dissertation conducts the research on reconstructing 3D scenes and objects from multiple sources and viewpoints. By leveraging prior shape knowledge, the proposed method reconstructs the 3D shape of an object from a single RGB or depth image. As the number of input images increases, the reconstruction results are incrementally refined. There are many applications for 3D reconstruction, including computer-aided design, mixed reality, robotics, and autonomous driving.

Based on the analysis of the present situation of research, the existing 3D reconstruction methods mainly suffers from three challenges. First, they typically require scanning all surfaces of an object before reconstruction, which is not always feasible in practice. Second, they only reconstruct the 3D structure from color or depth images, which can not make full use of data from different modalities and viewpoints. However, the feature matching of RGB images fails on weak or repeated texture objects. Moreover, the depth information can not be obtained from the objects without reflection. Third, they are semantic-free and thus the reconstructed objects and the background are mixed together, which causes difficulties to separate the objects from the reconstructed scene. To solve the three problems, the dissertation studies the corresponding problems from three levels: single-view 3D object reconstruction, multi-view 3D object reconstruction, and multi-view 3D scene reconstruction. Specifically, the main content and contributions are summarized as the following three aspects.

First, three geometry-structure-aware single-view 3D object reconstruction methods for monocular RGB cameras, stereo RGB cameras, and depth cameras are proposed to solve the problem that existing 3D reconstruction methods cannot restore the invisible part of the 3D structure of the object. The proposed methods recover the 3D shape of the invisible parts of objects by leveraging known colors, structure, and priors. For **monocular RGB cameras**, the geometry prior network is proposed to learn geometric priors from large-scale 3D datasets and implicitly establish the mapping relationship between image

space and 3D model space. For **stereo RGB cameras**, the depth-aware network is proposed for 3D object reconstruction. The proposed method estimates the depth map by leveraging the constraints of the two views, which better preserves the detailed 3D structure of objects when reconstructing the complete 3D shape of an object. For **depth cameras**, the gridding residual network is proposed for 3D object reconstruction. The proposed method takes the 3D grid as an intermediate representation of geometry structure, so that the context information is fully utilized in the calculation. Moreover, the geometry structures captured by the depth camera are better preserved. Experimental results on the ShapeNet, Pix3D, and KITTI datasets indicate that the three proposed methods recover the complete 3D shape of an object from a single-view image, which outperforms the existing 3D reconstruction methods with 3% to 18% performance improvement.

Second, the multi-source and multi-view 3D object reconstruction method based on the multi-scale context-aware fusion is proposed to solve the problem that existing methods cannot make full use of data from different modalities and viewpoints. On the one hand, the robustness of different data modalities to the objects with different materials are different. For example, it is difficult to recover the 3D structure of weak- or repeated-texture objects from multi-view RGB images. So do the depth cameras for the objects without reflection. On the other hand, different visible parts of an object from different viewpoints. The reconstruction qualities of the visible parts are much higher than those of invisible parts. Inspired by this observation, the multi-scale context-aware fusion module is proposed to adaptively select high-quality reconstruction for each part from different 3D shapes generated from different viewpoints or cameras. The selected reconstructions are fused to generate a 3D shape of the whole object. Experimental results on the ShapeNet, Pix3D and Thing 3D datasets indicate that the proposed method outperforms the state-of-the-art method with 4% to 20% performance improvement. Moreover, the proposed method is at most 7 times faster than the existing methods.

Finally, the semantic-aware multi-view 3D scene reconstruction method is proposed, which makes separating the 3D objects from the reconstructed scene easier. The proposed method reconstructs the 3D scene and the complete shape of 3D object simultaneously by modeling the semantics of the scene. To perceive the semantics of the scene, the regional memory network is proposed for video object segmentation, which separates the objects

from the image sequence and better distinguishes the objects with similar appearances. Furthermore, the semantic-modeling-based multi-view 3D scene reconstruction is proposed to reconstruct the scene and the inside objects by recovering the complete 3D shape of each object and estimating the position and pose of the object. Experimental results on the SUN3D dataset and on-site videos indicate that the proposed method achieves better reconstruction results for 3D scenes and objects compared to existing methods.

Through the above studies, the dissertation deeply explores the 3D scene and object reconstruction, providing feasible and effective solutions toward the key technical issues for real-world scenes. Starting from single-view single-object reconstruction, the dissertation further presents methods for multi-view single-object reconstruction and multi-view multi-object reconstruction. In view of the three problems in the existing 3D reconstruction methods, the proposed method reconstructs 3D scenes while perceiving the semantics, which recovers more complete shapes of objects and makes the reconstructed objects easier to be separated from the reconstructed scenes. The proposed method are more robust to recover the 3D shape to the weak- or repeated-texture objects and the objects without reflection.

**Keywords:** 3D reconstruction, 3D object reconstruction, scene semantic aware, multi-scale context aware, geometry structure aware

# 目　录

# Contents

# 第 1 章 绪论

## 1.1 研究背景和意义

### 1.1.1 研究背景

赋予机器像人类一样感知三维世界的能力一直是人工智能领域的一个长期研究的问题。想认知现实世界，人工智能系统必须理解三维视觉场景。图像和视频所描述的场景和物体本身仍是二维的，而真正的人工智能系统必须从图像和视频中理解所在场景和物体的三维结构，才能与之进行交互。因此，在研究领域和工业应用中，如何理解并重建三维场景成为了一个重要的研究课题。

计算机图形学和计算机视觉是两个互逆互补的方向。前者使用几何图形表达虚拟世界，后者用图像和视频表达真实世界。三维模型为用户提供了对虚拟空间的交互接口，但其真实性依赖于对模型的纹理、光照、运动等建模和绘制技术。而图像和视频则以更为高效的方式，动态、多视角地向用户刻画现实世界，有效弥补了理想数学模型和传统图形处理技术的缺陷。然而，图像和视频仅仅稀疏采样了真实世界在图像平面上的投影, 无法反映实际场景的三维结构；而且完整地表达客观世界需要庞大的图像和视频采样信息，这会导致计算和存储资源的占用问题。

为了解决上述问题，从图像和视频中恢复场景或物体的三维模型被广泛研究。在过去的几十年中，研究人员研究了从 X 恢复形状（Shape from X）[1] 进行三维重建。X 包括双目视觉（Stereo）[2]、立体视觉（Multi-view Stereo）[3]、轮廓（Silhouette）[4]、运动（Motion）[5]、聚焦区域（Focus）[6]、阴影（Shading）[7]、遮挡（Occlusion）[8]、纹理（Texture）[9] 和消失点（Vanishing Points）[10]。其中从双目视觉恢复形状，从立体视觉恢复形状，从轮廓恢复形状，从运动恢复形状和从聚焦视觉恢复形状为多视图图像输入，剩余的为单视图图像输入。然而这些方法均只能恢复可见部分的三维结构，并且对于输入图像有着诸多制约。然而，很多情况下无法在重建前扫描完整的三维物体，从而导致最终重建的三维结构不完整。

相比之下，人类的大脑非常擅长解决三维重建的问题。人类可以凭借所看到的部分物体推断出物体完整的三维结构。人类可以做到这些是因为之前长期积累的先验知识，这些先验知识不仅让人类可以推测物体完整的三维形状，还可以帮助人类理解真实场景。这种能力对于机器人抓取场景中的物体至关重要。如图 1-1所

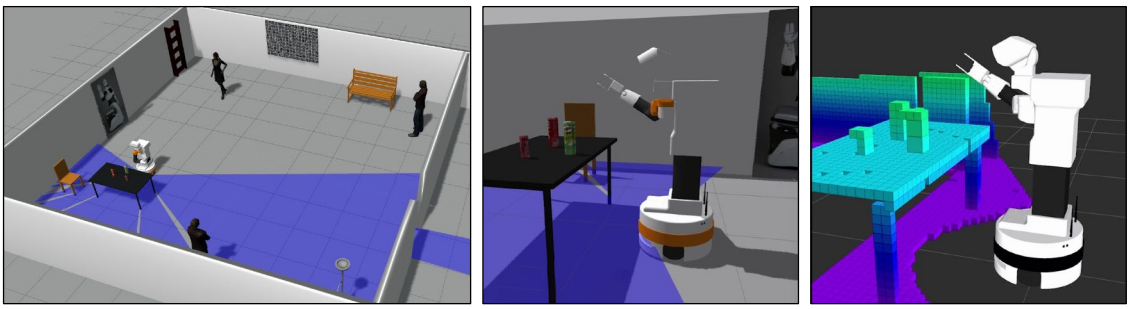


图 1-1 机器人抓取场景中物体的示意图
Fig.1-1 An illustration of the robot grabbing objects in the scene

示，机器人在抓取物体前需要理解所处三维场景，并了解该物体在场景中的位置和完整的三维结构。由于一些情况下无法在抓取前完整扫描该物体，因此从部分视角推断物体完整三维结构的能力变得尤为重要。

对于现阶段的智能系统而言，实现三维场景的理解与重建依然充满挑战，这主要有两个主要原因。第一，图像和视频可以看作是三维模型的投影，这些投影理论上可以对应无限个可能的三维模型，特别是这些投影非常稀疏时（即仅给定一张或者少量张数的图像时）。第二，真实场景通常较为复杂，多个物体会相互遮挡或存在自遮挡的情况，这导致了所观察到场景和物体信息的缺失。本文受到人类对于真实场景认知方式的启发，从大量的三维数据中学习形状先验，提出了针对不同图像采集设备的三维物体和场景重建算法。研究成果直接促进了三维场景和物体重建任务的发展，也为计算机辅助设计、混合现实、自动驾驶、机器人等领域提供了技术支撑。

### 1.1.2 研究意义

计算机视觉的研究正在从感知阶段发展至认知推理阶段。在真实场景中，我们不仅希望得到场景中的可见信息，更希望得到那些不可见的信息。例如，机器人在得知所处场景中包含一个杯子和它可见部分的三维结构，它也需要获得不可见部分的三维结构。这样，它才可以知道如何与之交互。而现有的三维重建方法不仅无法获取场景中的语义信息，并且也无法恢复不可见部分的三维结构。因此，三维场景的重建对于计算机辅助设计、混合现实、自动驾驶、机器人等领域均有重要的研究价值和现实意义。

计算机辅助设计在机械设计和工业制造等领域被广泛使用。然而，重新设计三维模型的过程往往耗时、耗力、耗资源。相比之下，通过三维重建从真实场景中获取三维模型并在此基础上进行修改可以节省大量成本，提高生产效率。然而，

现有基于从 X 恢复形状[1] 的三维重建方法仅能恢复可见部分的三维结构，如需恢复物体的完整形状，则需要在重建前扫描完整的物体，这在一些情况下是不可行的。另外，现有的三维重建方法依赖于特征匹配，对于弱纹理、重复纹理、半透明的物体或者当相机视角位移过大时，特征匹配将无法产生正确的结果，这些情况下三维结构将不能被准确的恢复。近几年，通过使用形状先验进行三维重建方法引发了越来越多的关注，并为更好地解决以上问题提供了可能。

混合现实技术分为后期混合现实和实时混合现实[11]。前者常见于影视特效的后期制作，例如预先拍摄完演员的动作，然后在后期时加上对于虚拟场景或物体的互动；后者则研究如何使得人们在现实世界中与虚拟物体互动，真切地感受来自虚拟空间的幻象事物。混合现实涉及对观察者方位的精确跟踪，对真实世界中场景和物体几何和材质的恢复，以及对光照环境的重建。然而目前采用的技术对硬件要求较高，这使得广大业余爱好者和家庭用户因成本问题而无法使用。本文所研究的三维场景和物体重建方案针对市场上主流的单目彩色相机、单目深度相机和双目彩色相机提出了对应的解决方案，从而为混合现实的民用化提供了技术支撑。

近年来，人工智能的发展也推动了自动驾驶技术的进步。研发自动驾驶技术对保障交通安全、改善交通拥塞以及降低人为交通事故所带来的经济损失有着重要意义。自动驾驶技术使得车辆在无需人类干预的情况下，从光学相机、激光雷达、毫米波雷达等多种传感器采集周围环境数据，通过自动驾驶算法感知周围环境并做出安全合理的决策。自动驾驶算法对采集到的信息进行分析、组织和解释，对行驶中的目标或障碍物进行识别、分割、检测、重建，获取其类别、尺寸、几何结构、行驶方向、位置等物理和语义信息。因此，对车辆周围的环境全面准确地感知是行车安全性和智能的保障，也是决策环节的前提。基于此，本文针三维场景和物体重建展开研究，帮助自动驾驶算法更好地感知周围环境，提高在复杂交通条件下的鲁棒性、准确性和适应性。

机器人技术在近几年飞速发展，机器人在工业、医疗、安防和生活中都得到了广泛应用。例如，工业机器人代替人类完成组装、焊接、配送等任务，提高了工厂的生产效率；微型医疗机器人可以帮助医生完成血管疏通、细胞切割等任务，为人类健康带来福音；安防机器人可以完成入侵检测、火灾报警等任务，保障人类财产安全。人类对于机器人的要求越来越高，可预见在不久的将来，机器人有望完成老人看护、家庭护理等任务。例如，它们可以定位客厅中的桌子、椅子、杯

子等，并将桌上的水杯送人们的的手中。为了完成这个目标，本文所研究的三维场景和物体重建方案可以帮助它们理解所在所在的场景，并通过局部视角估计每个物体完整的三维结构，从而帮助它们更好地和这些物体交互。

## 1.2 研究现状与分析

### 1.2.1 单视图三维物体重建

基于单视图的三维物体重建问题因其输入形式的特殊性使得更具挑战性，因为仅以单视角图像作为输入使得重建丢失了很多几何信息，这就需要一些假设或者先验知识，亦或是从已有的模型基于学习来进行重建。根据先验知识的来源和约束不同，现有的单视图三维场景和物体重建可以分为三类：基于模型的重建方法、基于几何的重建方法和基于学习的重建方法，如图 1-2所示。

基于模型的重建方法主要由待重建对象的模型和参数组成，通过寻找输入图像和模型投影之间的最佳参数完成重建[12]。在早期的工作中，基于规则模型的方法预先定义了某类形状的模型。广义柱体[13] 提出了柱类外形的紧凑描述。多面体模型[14] 只能重建方形物体，类似地，一些工作设计了针对车辆的刚性三维模型[15] 和表达圆角或近似方角的立方体、八面体、圆柱等形状的超二次曲面模型[16]。然而，这些模型都仅能对某种外形进行描述，导致可描述的对象太具有局限性。基于多边形网格的重建方法[17,18] 通过估计已有的多边形网格和输入图像的投影矩阵完成重建。然而实际场景中的物体并无法被已有的多边形网格所覆盖，因此限制了这类方法的泛化性能。近年来，可变形的多边形网格因更具表现力引起了更多学者的注意。在三维人脸重建[19] 和三维人体重建[20] 任务中，形变模型（Morphable Model）已经被大量使用。它是一种线性组合模型，通过图像光流算法建立图像和三维模型间的稠密对应，并通过调节形变和投影参数使得输入图像和模型匹配。形变模型通常需要通过繁琐的三维扫描获得，为了简化这个过程，Cashman 等人[21] 从单个模版三维模型和大量图像生成海豚的形变模型。Vicente 等人[22] 在 PASCAL VOC 数据集中检索出与输入图像同类型的相似视角不同物体的图像，并使用 Visual Hull 算法[23] 对目标物体进行重建。Kar 等人[24] 先对单视角的图像进行实例分割和视角估计，然后以此为基础估计出相对于某个类别平均模型的形变参数完成对目标物体的重建。总体而言，基于模型的方法在针对特定形状物体的重建上有较好的效果，因为在模型设计阶段它们较好的利用了目标物体的先验知识，不过这也导致

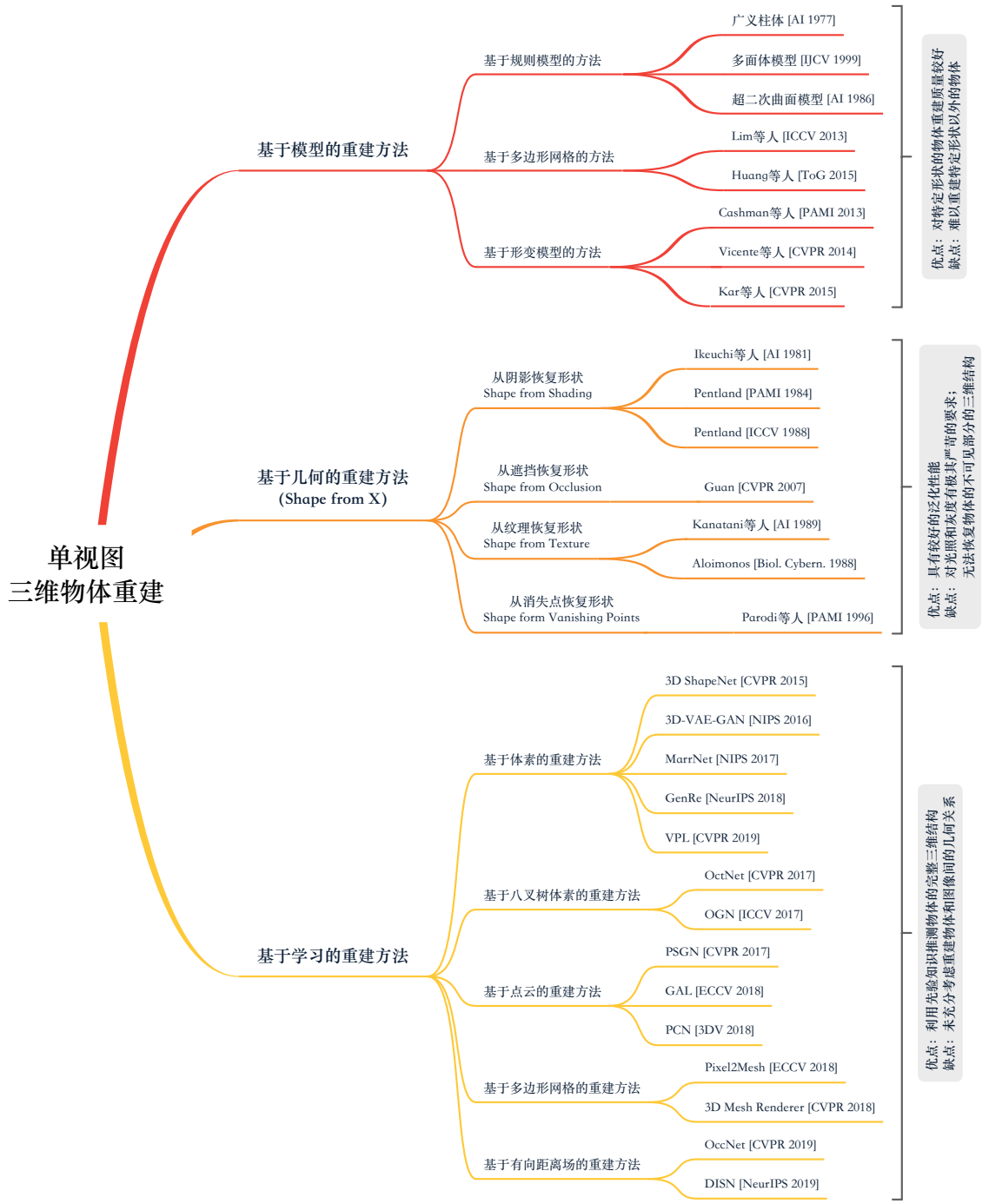



图 1-2 单视图三维场景和物体重建方法的思维导图

Fig. 1-2 The mind map of the single-view 3D scene and object reconstruction methods

了这类方法很难扩展至其他物体上。

基于几何的重建方法根据图像中的三维信息恢复物体的三维结构，这类方法通常被称为从 X 恢复形状[1]。针对单视角的三维重建，X 通常为阴影（Shading）、遮挡（Occlusion）、纹理（Texture）和消失点（Vanishing Points）。Shape from Shad-

ing[7] 利用物体表面的明暗变化恢复其表面各点的相对高度或表面法方向等参数值。对于真实采集的图像，物体表面的亮度受到光源、物体表面材料和摄像机位置等多种因素的影响。为了简化问题，Shape from Shading 通常假定物体表面模型为纯 Lambertian、纯镜面[25] 或更复杂的模型[26]。大多数的方法假定物体表面满足 Lambertian，即均匀照明的表面从任何一个方向观察其亮度不变。Shape from Occlusion[8] 利用遮挡产生的曲率变化推测物体的深度。Shape from Texture[9] 通常假定物体的纹理满足某种性质（例如纹理分布具有均一性[27] 或纹理由纹理基元组成[28]）。消失点是指某平面的一组平行线在透视投影下会聚的点，可以通过它得到其所对应平面的深度图。Shape from Vanishing Points[10] 又被称为 Depth from Geometrical Perspective，通常在室内环境下，沿着平行于消失点的平面，将靠近消失点的平面赋予较大的深度值。基于几何的重建方法往往具有更好的泛化性，因为其重建的是不针对特定类别的物体。但该类方法对光照和灰度等提出了苛刻的要求，例如使用理想光源来确保重建解的唯一性。此外，这类方法仅能恢复可见部分的三维结构，而无法恢复像基于几何的方法一样恢复物体的完整三维结构。

随着深度学习技术的飞速发展及大规模三维模型数据集的出现，基于学习的重建方法在过去的几年引发了广泛关注。3D ShapeNet[30] 利用深度卷积置信网络（Convolutional Deep Belief Network，CDBN）将输入的深度图像通过吉布斯采样（Gibbs Sampling）不断地预测外形类型和填补未知的体素完成重建。TL-embedding Network[31] 构建了一个自编码器，以 $20 \times 20 \times 20$ 的体素网格作为输入和输出，同时形成一个 64 维的特征向量；在重建时，通过 ConvNets 输入二维图像形成对应的特征向量，并将该特征向量送至解码器输出对应的体素。3D-VAE-GAN[32] 改进了 TL-embedding Network，通过使用变分自编码器得到图像的特征向量，在通过生成对抗网络得到重建的物体。3D-RecGAN++[33] 改进了 3D-VAE-GAN，将单张深度图的三维重建视作形状补全问题，提出了从单张深度图恢复某个物体的完整三维结构的方法。ProjectiveGANs[34] 和 VPL[35] 在生成器中加入了投影模块，将体素投影至特定视角的图像，并通过判别器判断该图像的真实性。为了使所提出的方法有更好的泛化性能，MarrNet[36]、ShapeHD[37] 和 GenRe[38] 从单视角彩色图像中估计深度图像，并使用形状补全解决了真实场景的三维物体重建问题。至此，基于学习的三维物体重建已经取得不错的进展。不过这些方法均选择了体素（Voxel）作为三维物体的表达形式。体素表示的三维物体虽然可以直接使用现有的 3D 卷积神经网络进行计算，但是它在计算时所需要的显存随着体素分辨率的提升而快速

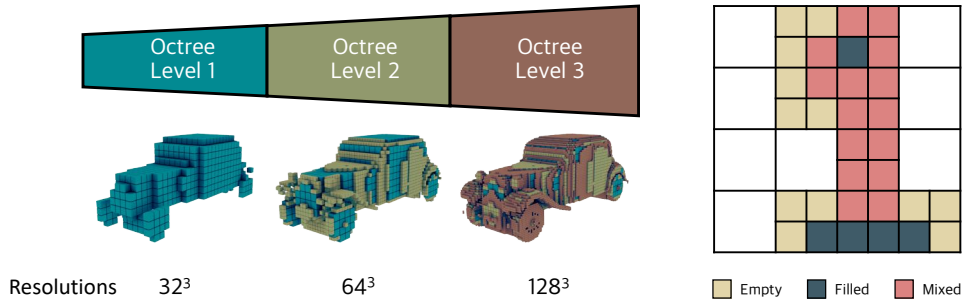



图 1-3 基于 HSP 八叉树结构的示意图[29]

Fig. 1-3 The illustration of the structure of HSP-based octree[29]

增长。针对这个问题，基于八叉树的卷积神经网络[29,39]被提出。OctNet[39]根据数据的密度不同将体素空间分割成一组不平衡的八叉树，其充分利用了体素数据的稀疏性，从而更合理的分配存储和计算资源。受此启发，HSP（Hierarchical Surface Prediction）[40]将体素分为占用、未占用和边界三类，从而可以重建时由粗粒度到细粒度逐步预测出某个物体在高分辨率下的形状，并显著减少多余显存的占用，其结构如图 1-3所示。OGN（Octree Generating Network）[29]采用了 HSP 的结构，在重建时同时预测八叉树的结构和物体的形状，并仅对那些未占用和边界的八叉树节点进行进一步的预测。除了八叉树，将点云（Point Cloud）、多边形网格（Polygon Mesh）、有向距离场（Signed Distance Field）作为三维物体表达形式的重建方案被陆续提出。PSGN（Point Set Generating Network）[41]第一次通过卷积神经网络实现从单视角图像到点云的重建。它有多个预测分支，其中包含卷积模块、反卷积模块、全连接模块。反卷积模块和全连接模块可以更好地刻画空间的连续性，从而更好地恢复三维物体中的光滑结构。而所采用的 HourGlass 结构[42]可以更好地联合全局和局部信息。GAL[43]中提出了几何对抗损失，通过保持预测点云和真实点云在不同二维视角上的几何一致性来从整体上规范生成的点云，同时它要求生成的点云与输入图片的语义相符。PCN 及其一系列的后续工作[44–48]借助于多层感知机（Multi-Layer Perceptrons）实现了从深度图像的三维重建，并和 3D-RecGAN++[33]一样，将其转换为三维形状补全的问题。 Pixel2Mesh[49]首次借助图卷积神经网络表示多边形网格，利用从输入图像中提取的特征逐步对椭球进行变形从而产生正确的几何形状。为了保证网格形状的正常更新，Pixel2Mesh 设计了一个图的上池化层（Graph Unpooling layer），使得点的数量逐渐由少到多，相应的网格形状由粗到细，既保留了全局信息，又具有细节的表达。Kato 等人[50]提出了 3D Mesh Renderer，它可以将渲染多边形网格变成一个可微分的操作，从而实现从彩色图像到多边形网格的三维重建。OccNet[51]采用了有向距离场隐式地将三维曲面表示为

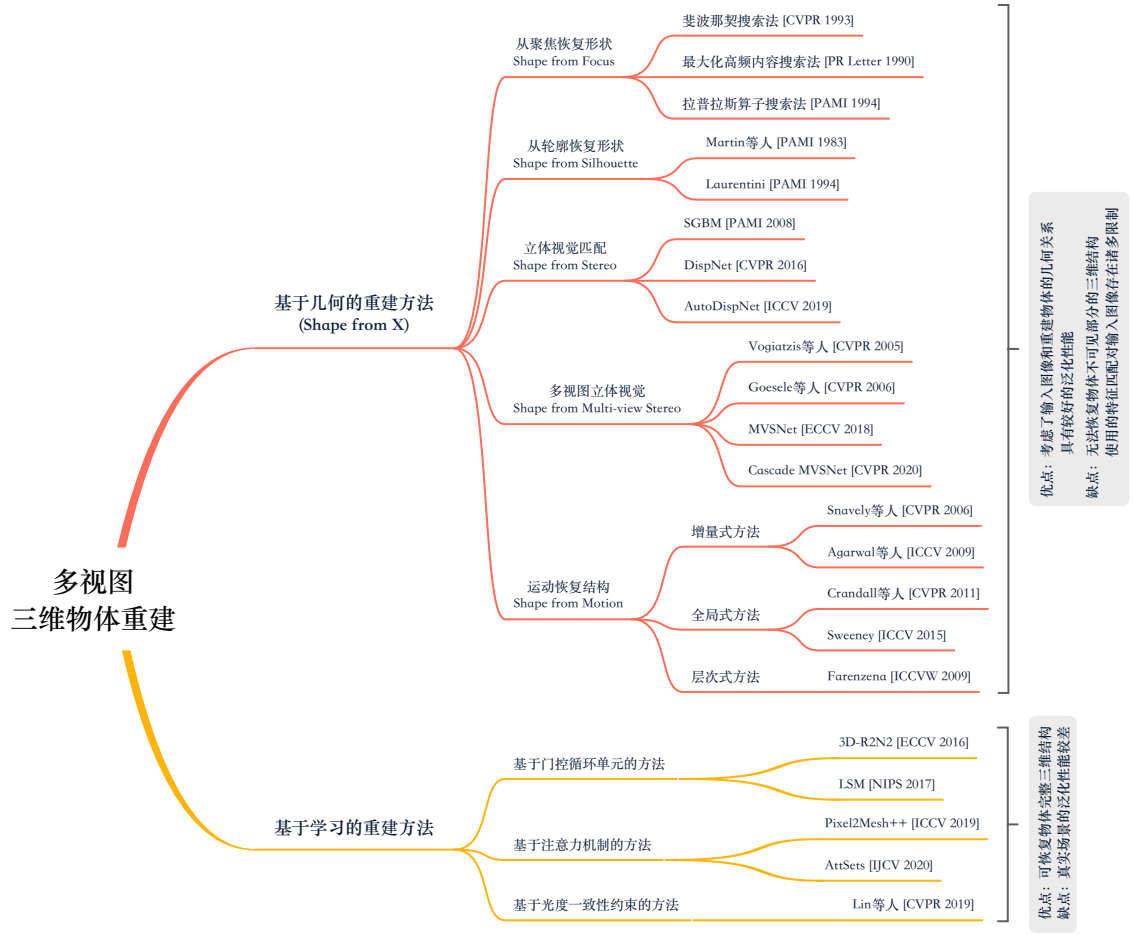



图 1-4 多视图三维物体重建方法的思维导图

Fig. 1-4 The mind map of the multi-view 3D object reconstruction methods

神经网络分类器的连续决策边界。与现有方法相比，该表示方式编码了高分辨率的三维物体表达，并且没有过多的内存占用。同时该方法能够高效地编码三维结构，并且能够从不同种类的输入推断出模型。DISN[52] 亦采用了有向距离场表示三维物体的结构，为了更好地保证重建结果和输入图像的一致性，DISN 估计了相机的位姿，并使用相机位姿将重建结果投影至图像计算损失。相比于 OccNet，DISN 可以更精确地还原三维物体的结构。

### 1.2.2 多视图三维物体重建

根据先验知识的来源和约束不同，现有的多视图三维物体重建可以分为两类：基于几何的重建方法和基于学习的重建方法，如图 1-4所示。

早期的多视图三维重建主要通过从 X 恢复形状实现，其中 X 包括聚焦区域（Focus）、轮廓（Silhouette）、双目视觉（Stereo）、立体视觉（Multi-view Stereo）和

运动（Motion）。从聚焦恢复形状（Shape from Focus）[6] 采集物体在不同焦距下的图像，对于每个表面区域检索出聚焦最佳的图像，从而推断出物体的深度，并得到三维结构信息。由于该方法需要对比图像中每个点在不同焦距下的情况，因此斐波那契搜索法[53]、最大化高频内容搜索法[54] 和拉普拉斯算子搜索法[55] 被提出以提高搜索速度。然而由于图像中噪声的存在，这些搜索方式难以达到很好地效果。从轮廓恢复形状（Shape from Silhouette）[4] 通过物体在不同视角下的轮廓图像（Contour）或侧影轮廓线（Silhouette）推断其三维模型。Martin 等人[56] 将物体所在的三维空间离散化成体素，并使用正向试探消除投影在轮廓区域外的体素，从而得到物体的三维模型。Laurentini[23] 引入了可视壳（Visual Hull），它代表所有轮廓图像反投影至三维空间所形成的三维锥壳的交集。当使用的轮廓图像足够多时，可视壳可以认为是三维模型的一个合理逼近。Szeliski[57] 和 Tarini[58] 分别使用了八叉树和 Matching Intersection 的结构提高了体素判断的效率。Matusik 等人[59] 使用多边形逼近轮廓图像侧影轮廓线，将复杂的三维锥壳相交简化为二维多边形求交集；不仅提高了建模的效率，也避免了锯齿形状的产生。因为轮廓图像的取值只有 2 种（前景区域或背景区域），所以基于轮廓的三维重建无法计算出物体表面的精确深度，并且也无法恢复物体表面的凹陷和空洞。Shape from Stereo 又被称为立体视觉匹配 (Stereo Matching)[2]，旨在从双目图像中找到匹配的对应点，从而恢复场景的 2.5 维结构。如图 1-5 a) 所示，对于真实世界中的某一点 $P$，在左图和右图中的像素点分别记作 $X_L$ 和 $X_R$，$X_L$ 和 $X_R$ 的水平位移被称为 $X_L$ 到 $X_R$ 的视差。现有立体视觉匹配的方法主要遵循图 1-5 b) 所示的流程，其中 Cost Volume 表示左右视差搜索空间，被用于在某个邻域内寻找与之匹配的像素点。在深度学习出现之前，立体视觉匹配被转化为求解能量函数的最小化问题，使得匹配代价最小，这类方法[60] 虽然取得了令人满意的结果，但是它们无法处理遮挡、弱纹理和重复纹理的场景。基于深度学习的方法在近几年飞速发展[2,61,62]，手工设计的特征描述子被卷积神经网络所替代，从而表现出更强大的性能。Shape from Multi-view Stereo 也被称为多视图立体视觉（Multi-view Stereo）[3]，主要用于从多张图像中恢复场景的深度图像（可视作稠密点云）。它可以看作是立体视觉匹配的扩展，相比于以 2 张图像作为输入的立体视觉匹配，它可以接受任意数量的图像。MVSNet[63] 借鉴了基于 Cost Volume 的双目立体视觉方法，将其推广至任意多张图像。它通过可微分的单应性（Homography）变换将输入图像特征变形到参考相机坐标系，这样在三维空间中构建 Feature Volume。为了处理任意视角输入图像，基于方差的 Cost 将多

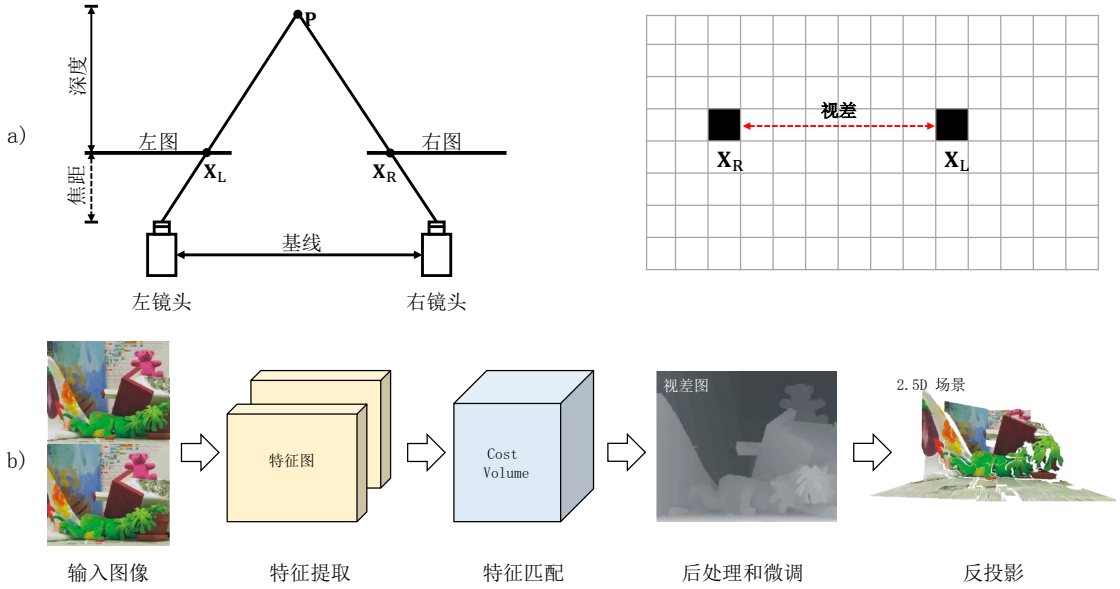


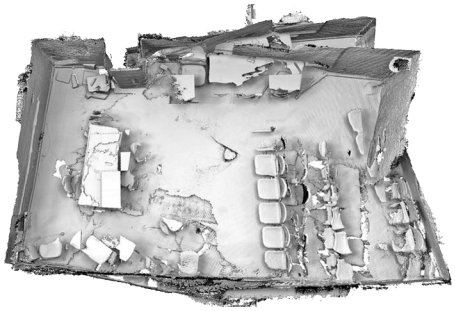

图 1-5 a) 立体视觉匹配（Stereo Matching）中视差的示意图 b) 立体视觉匹配的通用流程图

Fig.1-5 a) The illustration of the disparity in stereo matching b) The pipeline for stereo matching

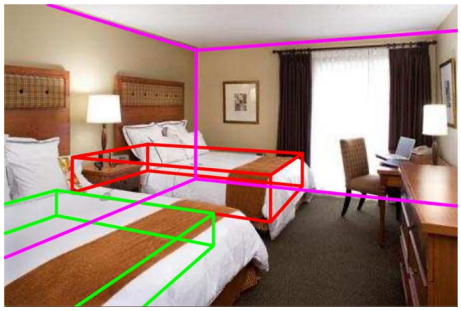
个 Feature Volume 映射到一个 Cost Volume，其中 Cost Volume 上的一个点是所有图像在这个点和深度值上特征的方差，越小的方差代表越高的置信度。之后，Cost Volume 被送至 3D 卷积神经网络生成深度值和对应的概率。RMVSNet[64] 将 3D 卷积神经网络替换为门控循环单元（Gated Recurrent Unit，GRU），从而提高了模型的效率并保证精度基本不下降。Point MVSNet[65] 在预测出深度图之后构建了三维点云，再引入了点云算法去优化深度估计结果。Cascade MVSNet[66] 将 MVSNet 改成了层次化结构，使得在构建 Cost Volume 时可以采用大的深度间隔和少的深度区间，从而有效降低了 50.6% 的 GPU 占用同时提高了 35.6% 的深度估计精度。Shape from Motion 通常被称为运动恢复结构（Structure from Motion）[5]，它从一组不同视角拍摄的无序或有序图像中同时恢复场景稀疏三维结构和相机姿态的技术。运动恢复结构通常分为 3 个阶段：提取并匹配图像特征，相机位姿估计、恢复三维场景结构。这些方法根据策略的不同可以被分为增量式（incremental）、全局式（Global）和层次式（Hierachical）。增量式的方法[67–69] 会先选出两张图像进行初始化。一对好的初始化图像应满足 2 个条件：（1）包含足够多的特征匹配点；（2）较宽的基线。在初始化完成后增量式地增加图像，估计图像所对应相机的内外参，并由三角化（Triangulation）[70] 得到三维点坐标，然后使用 Bundle Adjustment[71] 进行优化。增量式方法最明显的问题是产生的误差累积导致闭环场景的重建结果无法形成闭环。为了解决这个问题，全局式的方法[72,73] 考虑了相邻相机之间的关系，并

基于平均位移，从而缓解了增量式方法中误差累积效应的问题。然而全局式的方法受限于特征匹配的准确度，因此重建的精度和完整性都不如增量式的方法。相比增量式的方法，层级式的方法[74] 主要解决了其存在的 3 个问题：（1）重建结果易受到初始化的影响；（2）容易出现误差累积问题；（3）计算效率较低。层次式的方法使用了一种自底而上的层次聚类方法（Agglomerative Clustering），在每次迭代中合并具有最小距离的两个子簇（Cluster），每个子簇可以是一张图像或合并后的子簇。组成子簇的视图需要满足 2 个要求：（1）包含足够多的特征匹配点；（2）较宽的基线。在增量式的方法中，增加第 i 张图像需要做 i 次 Bundle Adjustment，时间复杂度为 $O(n^5)$；而在层级式的方法中，使用 Bundle Adjustment 仅优化每个子簇中的 k 个图像，因此时间复杂度可降至 $O(n^4)$。尽管在过去的几十年中，基于几何的多视角三维重建方法取得了较为瞩目的结果，然而这些方法只能重建物体或场景的可见部分的三维结构，并且依赖于不同图像之间的特征匹配，因此无法重建弱纹理和重复纹理的物体。

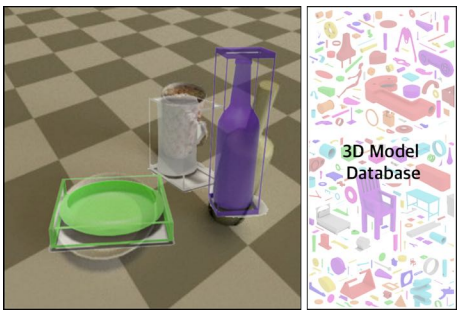
随着深度学习技术的飞速发展及大规模三维模型数据集的出现，基于学习的多视角三维重建方法在过去的几年引发了广泛关注。3D-R2N2[75] 使用了门控循环单元的卷积神经网络从单张或多张图像中恢复某个物体的完整三维结构。和之前依赖于特征匹配的三维重建方法不同，所提出的方法直接学习从二维到三维空间的映射关系，因此可以被更好地应用至特征匹配失败的场景（如弱纹理或重复纹理的物体）。LSM[76] 在 3D-R2N2 的基础上引入了相机的外参并提出了反投影的操作，从而更好地对齐来自于不同视角图像特征。由于 3D-R2N2 和 LSM 均采用了门控循环单元，然而它并不满足排列不变性：以不同顺序给定相同的图像序列，算法并不能产生一致的输出。为了解决这个问题，AttSets[77] 提出了注意力聚合模块（Attentional Aggregation Module），用于对不同视角的图像特征进行加权。Pixel2Mesh++[78] 将 Pixel2Mesh 扩展至多视角的图像输入并提出了多视角可变形网络（Multi-view Deformation Network），从而实现了多视角的三维物体重建。和 3D-R2N2 这类直接学习从二维到三维空间的映射关系的方法不同，Pixel2Mesh++ 将三维重建转换为预测物体形变的问题，这使得所生成的模型具有更好的细节。Lin 等人[79] 将多视角的三维物体重建视作每个网面投影（Mesh Face Projection）的分段图像对齐问题，通过优化目标网格以实现多视图光度一致性，同时用形状先验约束网格变形。所提出的方法允许在没有任何深度或蒙版信息的情况下，根据光度误差（Photometric Error）更新形状参数。尽管这些方法解决了从 X 恢复形状无法



a) 无语义的三维场景重建
a) Semantic-free 3D scene reconstruction



b) 基于场景和目标识别的场景重建
b) 3D Scene understanding by reasoning about the layout and objects




c) 基于物体检索的三维场景重建
c) 3D scene reconstruction by object retrieval

d) 基于物体重建的场景重建
d) 3D scene reconstruction by object reconstruction

图 1-6 典型三维场景重建方法的分类
Fig. 1-6 The categories of the representative 3D scene reconstruction methods

恢复物体完整三维结构的问题，但它们对真实场景中物体的重建并不理想。

### 1.2.3 三维场景重建

理解并重建场景中的三维物体是计算机图形学和计算机视觉领域长久研究的问题。在本节中，物体是指场景中可以移动的家具等而并非建筑结构本身。尽管这些物体和建筑结构本身是完全不同的，但一些场景重建的方法[80,81]并没有区分它们，而是直接恢复了物体和建筑联合的三维结构，如图 1-6 a)所示。早期的方法[82–87]通过单张图像获取场景中物体的三维边界框（Bounding Box），并获知它们对应的语义信息，如图 1-6 b)所示。这些方法通常包含 2 个较强的假设：（1）物体平面和墙面是垂直的；（2）物体底面和地面相接触。Lee 等人[83]提出了可以同时提取场景布局和物体位置的方法。它通过从消失点中估计曼哈顿世界的主导方向（Manhattan World Dominant Directions），并在场景空间中搜索和建筑布局对齐的物

体。Hedau 等人[82] 使用了一个基于外观的分类器分类视觉特征，并参数化一个三维边界框使得它可以表征一个物体。这些早期的方法为了减少计算复杂度仅使用了非常少量的样本，因此它们的性能非常有限。Del Pero 等人[84,88] 使用马尔可夫链蒙特卡洛算法（Markov Chain Monte Carlo）生成更多的样本。它们同时估计场景的三维结构和场景中物体的三维结构以更精确地恢复结构的细节。然而，这类方法对于相机高度和场景高度的比例有着非常严格的要求，从而限制了它的广泛使用。Schwing 等人[85] 利用了分支与界限策略（Branch-and-Bound Strategy）推断空间布局以及物体的位置和大小，从而达成了精确的场景理解。PanoContext[86] 证明了场景的上下文信息可以通过全景图像获得。给定场景的全景图，它可以输出场景的三维边界框以及场景内主要物体的边界框及其对应的类别。Pano2CAD[87] 使用曼哈顿世界布局将 PanoContext 拓展至非箱型场景。在获得三维边界框之后，一些方法[89,90] 使用了单视图或多视图三维重建的方法恢复场景和物体的三维结构。Bao 等人[89] 从针孔相机拍摄的视频（至少包含 10 帧）中推断图像中物体的标签，并被反投影至由运动恢复结构方法构建的三维场景中，从而获得每个物体的三维结构。Yang 等人[90] 利用了线条、消失点、方向图（Orientation Map）和表面法向量提供的三维结构线索以及显著性和物体检测提供的语义信息从全景图像中恢复了场景 2.5 维的结构。3D-SIS[91] 使用了语义分割算法获得场景的语义信息，它借助卷积神经网络融合了来自彩色和深度图像的信息，实现了多视角 RGB-D 图像的实例级体素语义分割。这些多视角的图像特征被反投影至一个更大的三维网格（3D Grid）中，从而聚合了来自不同视角信息。

随着大规模三维数据集的普及，越来越多的方法[92–97] 得以使用“检测-检索”的策略恢复场景中的物体的完整三维结构，如图 1-6 c)所示。Shao 等人[92] 将输入的 RGB-D 图像进行图像级的语义分割，并从三维模型数据库中检索对应的类别模型以完成场景中物体的重建。Nan 等人[93] 在检索时使用了非刚性配准（Non-Rigid ICP）算法从而更好地检索三维模型数据库中模型。3DNN[94] 实现了一个自顶向下的匹配方法从三维模型数据库中匹配图像中出现的模型。为了更好地匹配场景中的模型，他们在与主导的曼哈顿世界方向[83] 对齐的箱型场景中进行相机配准，然后将三维模型添加该场景中并进行渲染。Kim 等人[95] 在获取场景的三维扫描和对应的三维边界框之后查找场景中具有不同位姿的相同物体，并相互补全各自缺失的部件。这个方法引入了“部件级-部件级”的匹配，然而这种匹配仅限同一物体，无法匹配不同物体间相同的部件。为了突破这个限制，Shen 等人[96] 使用了一个包

含部件分割的三维模型数据库，它通过最小化检索得到的三维模型和输入三维模型的误差以实现场景中物体的重建。3D-RCNN[97] 在检索得到三维模型后，使用渲染模块得到重建后的图像，并最小化该图像和输入图像的误差从而更精确地检索三维模型。Points2Objects[98] 可以同时检索图像中的全部物体，在检索得到对应物体的三维模型后估计对应的 9-DoF（Degree of Freedom，自由度）边界框，并将检索后的模型放回至原场景中。

然而三维模型数据库无法穷举真实世界中全部的三维模型，从而导致这些方法无法重建不包含在其中的物体的三维结构。为了解决这个问题，一些方法在推理时不再依赖三维模型数据集，直接识别场景中的物体，并借助神经网络从大量数据中学习的先验知识重建物体的三维结构，如图 1-6 d)所示。作为这个方向的先驱者，Voxlets[99] 通过随机森林从单张深度图像中推断场景中物体的完整的三维体素结构。Mesh R-CNN[100] 利用目标实例分割任务的通用 Mask R-CNN[101] 获取场景中包含的物体，并用另一个分支预测该物体的三维结构。Total3DUnderstanding[102] 和 FroDo[103] 作为近期的工作，均使用了二维目标检测获取场景中物体的二维边界框，并使用了三维目标检测获取物体的位姿。它们在使用二维边界框裁剪图片后，将包含物体的图片输入至重建网络恢复物体的完整三维结构，再根据估计所得到的物体位姿将重建后的物体放回至场景中。CoReNet[104] 中提出的光线跟踪残差连接（Ray-traced Skip Connections）使得图像中所有的二维信息都可以被准确的映射至三维空间中，使得该网络同时重建单张彩色图像中全部的物体。其中所提出的三维体素混合表示（Hybrid 3D Volume Representation）使得它可以建立平移不变性的模型，同时对精细的对象细节进行编码而不占用过多的内存。

### 1.2.4 具有代表性的三维场景和物体重建的数据集

表 1-1总结并对比了从 2012 年至今具有代表性的三维场景理解与重建的数据集。和其他的深度学习方法一样，三维场景理解与重建任务也需要大规模的数据集训练神经网络。然而，构建大规模的三维场景理解与重建数据集却充满挑战：（1）获取图像是非常容易的，然而获取对应的三维模型却并不容易，因此很多数据集（如 IKEA[17]，PASCAL 3D+[108]，ObjectNet 3D[111] 和 Pix3D[117]）仅提供了极少量配对的数据集；（2）目前较大规模的数据集比如 ModelNet[30] 和 ShapeNet[109] 仅包含了三维模型却并不包含对应的自然背景，因此无法很好地将训练后的模型应用至自然场景中。

表 1-1 具有代表性的三维场景理解与重建的数据集

Table 1-1 Representative datasets for 3D scene understanding and reconstruction

| 数据集 | 年份 | 图像 | | | | | 物体 | | |
|---|---|---|---|---|---|---|---|---|---|
| | | 图像数量 | 尺寸 | 物体数量 | 类型 | 背景 | 类别数量 | 物体数量 | 类型 |
| NYU v2[105] | 2012 | 1,449 | 640 × 480 | 多个 | 室内场景 | 真实背景 | 894 | 35,064 | 深度图 |
| KITTI[106] | 2012 | 41,778 | 1240 × 376 | 多个 | 室外场景 | 真实背景 | 2 | 40,000 | 点云 |
| IKEA[17] | 2013 | 759 | 可变 | 单个 | 室内场景 | 真实背景 | 7 | 219 | 多边形网格 |
| SUN3D[107] | 2013 | 2,513,609 | 640 × 480 | 多个 | 室内场景 | 真实背景 | - | - | 深度图 |
| PASCAL 3D+[108] | 2014 | 30,899 | 可变 | 多个 | 室内外场景 | 真实背景 | 12 | 36,000 | 多边形网格 |
| ShapeNet[109] | 2015 | - | - | 单个 | 无场景 | 纯色背景 | 55 | 51,300 | 多边形网格 |
| ModelNet[30] | 2015 | - | - | 单个 | 无场景 | 纯色背景 | 662 | 127,915 | 多边形网格 |
| SUN RGB-D[110] | 2015 | 10,335 | 640 × 480 | 多个 | 室内场景 | 真实背景 | 800 | 64,595 | 体素 + 深度图 |
| ObjectNet3D[111] | 2016 | 90,127 | 可变 | 多个 | 室内外场景 | 真实背景 | 100 | 44,147 | 多边形网格 |
| SUNCG[112] | 2017 | 130,269 | 640 × 480 | 多个 | 室内场景 | 合成背景 | 84 | 5,697,217 | 体素 |
| SceneNet RGB-D[113] | 2017 | 5,068,500 | 320 × 240 | 多个 | 室内场景 | 合成背景 | 255 | - | 体素 + 深度图 |
| 2D-3D-S[114] | 2017 | 70,000 | 1080 × 1080 | 多个 | 室内场景 | 真实背景 | 13 | 6,005 | 多边形网格 |
| ScanNet[115] | 2017 | 2,492,518 | 640 × 480 | 多个 | 室内场景 | 真实背景 | 296 | 36,123 | 深度图 |
| Matterport3D[116] | 2017 | 194,400 | 1280 × 1024 | 多个 | 室内场景 | 真实背景 | 40 | 50,811 | 多边形网格 |
| Pix3D[117] | 2018 | 9,531 | 可变 | 单个 | 室内场景 | 真实背景 | 9 | 1,015 | 多边形网格 |
| InteriorNet[118] | 2018 | 20,000,000 | 640 × 480 | 多个 | 室内场景 | 合成背景 | 128 | 1,042,632 | 多边形网格 |
| 3D Scene Graph[119] | 2019 | - | 512 × 512 | 多个 | 室内场景 | 真实背景 | 80 | - | 多边形网格 |
| Structured3D[120] | 2020 | 196,515 | 1280 × 720 | 多个 | 室内场景 | 合成背景 | 40 | - | 多边形网格 |
| 3D-FRONT[121] | 2020 | 11,676 | 256 × 256 | 多个 | 室内场景 | 合成背景 | - | 62,424 | 多边形网格 |

目前解决这些问题的方法主要是通过数据增广（Data Augmentation）。在图像处理任务中，之前图像领域的工作通过应用平移、旋转和缩放至现有的数据以增加图像数据的丰富性。在三维重建任务中，还可以在生成图像的过程中使用不同的视角、相机位姿、光照条件和背景等进一步丰富数据集。现有的很多数据集均采用了这一方案，包括 SUNCG[112]、SceneNet RGB-D[113]、InteriorNet[118]、Structured3D[120] 和 3D-FRONT[121]。不过这种方案造成了领域偏移（Domain Shift）的问题，即渲染得到的合成图像和真实场景拍摄的图像的数据分布不一致，从而造成了模型无法很好地泛化至真实数据集。也有一些数据集直接通过 RGB-D 相机或激光雷达扫描真实场景构建三维场景理解与重建的数据集，例如 NYUv2[105]、KITTI[106]、SUN RGB-D[110]、ScanNet[115]、Matterport3D[116] 和 3D Scene Graph[119] 等。然而，这些数据集并未提供场景中物体完整的三维模型，因此无法用于训练三维物体重建的网络。解决真实数据和合成数据的鸿沟一直是机器学习领域长期研究的问题。目前，已经有一些领域迁移（Domain Transfer）的工作[122,123] 开始解决三维重建领域中所面临的这个问题，并可以在一定程度上缓解这个问题。

### 1.2.5 现有方法的主要问题

（1）现有的方法仅能重建可见部分的三维结构

传统的三维重建方法（如运动恢复结构、多视图立体视觉等）从图像序列中推断视图间的几何关系从而恢复场景的三维结构。因此，在重建完整物体的三维结构时需要扫描整个物体，但这在一些情况下是不可行的。并且在扫描整个物体时，不可避免地会产生回环，而传统的三维重建方法在这种情况下极易造成误差累积。相比之下，人类可以很容易地从部分观测中推断某个物体完整的三维结构。之所以能做到这一点，是因为所有之前看到的大量物体和场景都使人类建立了先验知识库。因此，如何像人类一样在重建时通过可见部分的信息推断不可见部分的三维结构成为了一个重要的研究问题。

（2）现有的方法无法充分利用来自多个数据源的特征

传统的三维重建方法主要是用彩色或深度图像重建场景的三维结构，无法融合来自不同数据模态的特征。尽管基于一些 RGB-D 的三维重建方法被提出，但这些方法使用彩色图像进行特征匹配并根据 PnP（Perspective-n-Point）算法[124] 确定相机位姿，再使用深度图像提供物体几何信息。然而，对于弱纹理或重复纹理的物体，特征匹配算法将会失效；对于不发生反射的物体，深度相机也无法提供物

体的几何信息。综上所述，现有的三维重建算法无法重建包含弱纹理、重复纹理或不发生发射物体的三维结构。因此，如何更充分利用来自多个数据源或多个视角的信息重建弱纹理、重复纹理或不发生发射的物体和场景是一个重要的研究点。

（3）现有的方法对场景中的物体进行分解建模

传统的三维重建方法在重建场景时并不感知场景的语义信息，而是直接重建整个场景的三维结构。在这种情况下，重建后的物体和场景将会融为一体，并难以分离。如果要将某个物体从场景中分离，则需对重建后的场景进行繁杂的人工后处理。因此，如何设计有效的方案对场景中的物体进行分解建模是一个亟待解决的问题。

（4）现有方法无法重建非刚性或运动物体

目前所有的三维场景和物体重建方法均针对静止的刚性物体，对于非刚性物体（例如液态物体）和动态物体等一直无法较好地重建。因此，非刚性和运动物体的三维重建问题也将是未来的研究热点。

## 1.3 本文的主要研究内容与组织结构

### 1.3.1 本文的研究内容

通过 1.2.5节中的分析，三维场景和物体的重建问题仍然存在诸多问题和挑战。本文针对 1.2.5节中所提到的前三个问题，开展三维场景和物体的重建关键技术的研究。具体地，本文的主要研究内容将从以下几个关键问题展开：

（1）如何重建不可见的三维结构?

近几年，基于学习的三维重建方法利用从大规模的三维模型数据集学习的形状先验，建立二维图像到三维模型的映射，从而实现三维重建。不同的相机所能提供的信息不尽相同，例如双目相机可以提供视差信息，深度相机可以提供深度信息等，这些信息本身已经一定程度上包含了几何关系。因此，如何针对不同的相机设计有效的三维重建方法，并充分利用相机所能提供的颜色或者空间信息更精准地还原物体的三维结构信息是研究的重点。

（2）如何融合来自不同数据源的特征?

不同的模态的数据源对于不同材质的物体有不同的鲁棒性。例如彩色图像不擅长弱纹理、重复纹理物体的重建，深度图像不擅长不发生反射物体的重建。尽管如此，不同模态的数据信息可以取长补短。因此，如何充分利用这种信息的互

补性，有效融合不同模态数据源的特征成为了需要研究的问题。

（3）如何对场景中的物体分解建模?

为了从场景中对每个物体进行分解建模，本文将设计基于场景语义感知的三维场景重建方案。分解建模后，每个物体的完整三维结构都可以直接从重建后的场景中分离。在该方案中，场景中的目标物体在重建前将会通过视频物体分割（Video Object Segmentation）方法从图像序列中分离；然后该方案将会对每个物体进行重建，并估计物体的位置和位姿，进而完成对三维场景的重建。因此，如何从图像序列中分离目标物体，并在物体重建后组合成三维场景成为了本文需要解决的问题。

### 1.3.2 本文的组织结构

本文的研究目标是多源多视的三维场景和物体重建，旨在让人工智能系统从图像和视频中理解所在场景并重建该场景和其中物体的三维结构，从而更好地与之进行交互。在对研究现状进行分析的基础上，首先，从单源单视的单个三维物体重建问题出发，本文提出了基于几何先验的单目彩色图像三维物体重建、基于深度感知的双目彩色图像三维物体重建和基于网格化残差网络的深度图像三维物体重建，在重建时从可见部分的三维结构推断了不可见部分的三维结构；然后，对于多源多视的单个三维物体重建问题，本文提出了基于多尺度上下文感知融合的多源多视三维物体重建方法，有效地融合了多个单源单视三维物体的重建结果，使得不同数据源和视角的信息得以相互补充；最后，对于多源多视的多个三维物体重建问题，本文提出了基于场景语义感知的多源多视三维场景重建，在完成三维场景重建的同时理解场景的语义信息，使得在重建后可以直接将目标物体从场景中分离。图 1-7展示了本文的主要研究内容和各章节之间的逻辑关系。

首先，单源单视单个三维物体重建的问题的关键在于有效利用已知的先验信息并在重建时推断物体未知部分的三维结构。为了解决这一问题，本文针对单目彩色相机、双目彩色相机、单目深度相机的特性，分别设计了基于几何先验的三维重建、基于深度感知的三维重建和基于网格化残差网络的三维重建。这些方法从大规模的三维模型数据集中学习形状先验，使得模型可以仅使用单源单视的图像即可恢复物体的完整三维结构。另一方面，所提出的方法针对不同相机的特性充分挖掘图像与重建模型间的几何关系，从而更好地恢复物体的细节。

其次，多源多视单个三维物体重建的问题的关键在于设计有效地策略融合多个不同数据源或视角的三维物体的重建结果。一方面，不同模态的数据对于不同

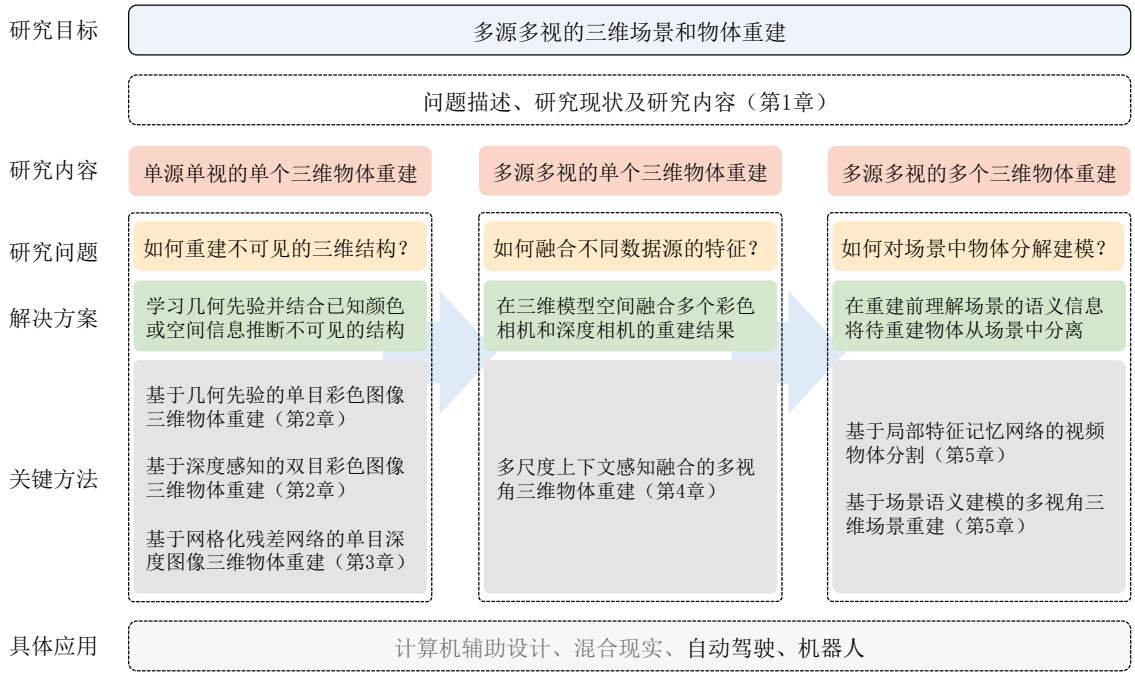



图 1-7 本文的主要研究内容以及各章节之间的逻辑关系

Fig. 1-7 The main contents and relationships among different chapters

材质的物体具有不同的鲁棒性：对于弱纹理、重复纹理的物体，多视角彩色图像难以恢复其三维结构；对于不发生反射的物体，深度图像也无法获取其几何信息。另一方面，受到可见部分的重建结果优于不可见部分重建结果的启发，本文提出了多尺度上下文感知融合模块，对单源单视重建结果中的每个部分质量进行评估，从每个重建结果中选取最佳部分以生成最终的重建结果。所提出的方法在三维模型空间融合多个彩色相机和深度相机的重建结果，使得不同数据源和视角的信息得以相互补充。此外，为了更好的探究影响自然场景重建结果的因素，本文提出了迄今为止最大规模的多视角自然场景的三维物体重建数据集，并改进了在真实场景中三维物体重建的结果。

最后，多源多视多个三维物体重建问题的关键在于充分利用图像序列的信息并推断场景的语义信息并重建其三维结构。为了从图像序列中理解场景中的语义信息，本文提出了基于局部特征记忆网络的视频物体分割方法，从而完成对所在场景的语义理解。为了重建场景中的物体，本文设计了基于场景理解和姿态估计的三维场景重建算法。所提出的方法通过视频物体分割方法将场景中的物体从图像序列中分离，对每个物体进行重建，并估计物体的位置和位姿，进而完成对三维场景的重建。该方法在重建后可以直接从场景中获取每个物体完整的三维结构，实验结果证实了本文所提出方法在真实场景中的实用性和有效性。

# 第 2 章 几何结构感知单源单视彩色图像三维物体重建

## 2.1 引言

从彩色图像中恢复物体完整的三维结构在计算机辅助设计、混合现实、机器人和自动驾驶领域均发挥重要作用。传统的三维重建方法主要使用 Shape from X[1] 或者运动恢复结构从单视角或者多视角图像中恢复物体的三维结构。然而，Shape from X 通常对形状或者环境有非常高的约束条件，因此难以在现实环境中使用；运动恢复结构需要在重建时扫描完整的物体，然而这在一些情况下并不可行。

随着深度学习技术的飞速发展及大规模三维模型数据集（如 ShapeNet[30]）的出现，基于学习的单视角三维重建方法[29,41] 在过去的几年蓬勃发展，并取得了令人瞩目的结果。为了在重建时有效利用已知的先验信息并在重建时挖掘图像和三维物体间的几何关系，根据不同相机类型所采集信息的不同，本文针对市场上主流的单目彩色相机和双目彩色相机分别设计了基于几何先验的单目彩色图像三维物体重建方法和基于深度感知的双目彩色图像三维物体重建方法。

为了在重建时有效利用已有数据的先验知识，基于几何先验的单目彩色图像三维物体重建方法以 ImageNet 上预训练的 VGG-16 作为编码器的骨干网络抽取图像特征，并通过由 3D 转置卷积层组成的解码器将二维图像特征映射至三维模型空间。基于 U-Net[125] 的微调器进一步修正解码器所产生的错误，改善了重建结果的质量。在该过程中，网络从大规模的三维数据集中学习几何先验，建立了从 RGB 空间到三维模型空间的映射关系。

深度信息被证明可以改善重建质量[36,38]。当前双目相机在智能手机和机器人的广泛使用，使得可以在重建时估计出物体的深度。除了深度信息，双目相机可以在重建时探索视图间稠密的特征匹配，这有利于更好地挖掘图像和三维物体间的几何关系。因此，基于深度感知的双目彩色图像三维物体重建方法在重建物体完整三维结构时显式地估计了双目的视差和视图间稠密的特征匹配。另外，由于目前没有针对双目彩色相机的三维物体重建数据集，因此本文使用 ShapeNet 构建了一个新的数据集：StereoShapeNet。该数据集包含 1,052,976 对双目彩色图像，以及对应的深度图和视差图。

本章的贡献可以归纳为以下三点：

• 针对单目彩色相机，提出了基于几何先验的单目彩色图像三维物体重建方法，命名为 Pix2Vox。该方法以 ImageNet 上预训练的 VGG-16 作为编码器的骨干网络，从大规模的三维数据集中学习几何先验，更好地从单视角的图像中恢复物体的完整三维结构。

• 针对双目彩色相机，提出了基于深度感知的双目彩色图像三维物体重建方法。该方法估算了视差并从中推断出物体的 2.5 维结构，并利用两个视图间的特征匹配在重建物体完整三维结构时充分挖掘图像和三维物体间的几何关系。

• 在 ShapeNet、StereoShapeNet、Pix3D 和 Driving 数据集上的实验结果表明，相比于其他基于学习的单视角三维物体重建方法，所提出的方法具有更好的重建性能和泛化性能。

## 2.2 相关工作

早期的工作使用 Shape from X[1] 从单视角图像中恢复物体的三维结构，这些工作通常对形状或者环境有非常高的约束条件，因此难以在现实环境中使用。随着深度学习技术的飞速发展及大规模三维模型数据集的出现，基于学习的单视角三维重建方法在过去的几年蓬勃发展；这些方法不再对形状或者环境有约束，因此受到了广泛关注。3D ShapeNet[30] 利用深度卷积置信网络（Convolutional Deep Belief Network）将输入的深度图像通过吉布斯采样（Gibbs Sampling）不断地预测外形类型和填补未知的体素完成重建。TL-embedding Network[31] 构建了一个自编码器，以 $20 \times 20 \times 20$ 的体素网格作为输入和输出，同时形成一个 64 维的特征向量；在重建时，通过 ConvNets 输入二维图像形成对应的特征向量，并将该特征向量送至解码器输出对应的体素。3D-VAE-GAN[32] 改进了 TL-embedding Network，通过使用变分自编码器得到图像的特征向量，在通过生成对抗网络得到重建的物体。3D-RecGAN++[33] 改进了 3D-VAE-GAN，将单张深度图的三维重建视作形状补全问题，提出了从单张深度图恢复某个物体的完整三维结构的方法。ProjectiveGANs[34] 和 VPL[35] 在生成器中加入了投影模块，将体素投影至特定视角的图像，并通过判别器判断该图像的真实性。为了使所提出的方法有更好的泛化性能，MarrNet[36]、ShapeHD[37] 和 GenRe[38] 从单视角彩色图像中估计深度图像，并使用形状补全解决了真实场景的三维物体重建问题。近几年，基于点云、多边形网格、有向距离场的三维物体重建方案被陆续提出。PSGN[41] 第一次通过卷积神经网络实现从单视角图像到点云的重建，其中而所采用的 HourGlass 结构[42] 可以更好地联合全局

算法 2-1 Pix2Vox-A 算法
Algo.2-1 The Pix2Vox-A algorithm

**Input:** 单视角单目彩色图像 $\mathbf{I} \in \mathbb{R}^{224\times224\times3}$

**Output:** 三维体素模型 $\mathbf{v}^f \in \mathbb{R}^{32\times32\times32}$

1 给定输入图像 $\mathbf{I}$，编码器经过 4 次降采样输出特征图 $\mathbf{F} \in \mathbb{R}^{256\times8\times8}$；

2 调整 $\mathbf{F}$ 的尺寸至 $2048 \times 2^3$；

3 给定特征图 $\mathbf{F}$，解码器经过 4 次上采样输出粗糙重建结果 $\mathbf{v}^c \in \mathbb{R}^{32\times32\times32}$；

4 给定粗糙重建结果 $\mathbf{v}^c$，微调器输出最终重建结果 $\mathbf{v}^r$。

和局部信息。GAL[43] 中提出了几何对抗损失，通过保持预测点云和真实点云在不同二维视角上的几何一致性来从整体上规范生成的点云，同时它要求生成的点云与输入图片的语义相符。Pixel2Mesh[49] 首次借助图卷积神经网络表示多边形网格，利用从输入图像中提取的特征逐步对椭球进行变形从而产生正确的几何形状。OccNet[51] 采用了有向距离场隐式地将三维曲面表示为神经网络分类器的连续决策边界。为了更好地保证重建结果和输入图像的一致性，DISN[52] 估计了相机的位姿，并使用相机位姿将重建结果投影至图像计算损失。相比于 OccNet，DISN 可以更精确地还原三维物体的结构。

## 2.3 基于几何先验的单目彩色图像三维物体重建

### 2.3.1 模型与方法

为了从单张任意视角的彩色图像中恢复物体的完整三维结构，本文提出了基于几何先验的三维物体重建方法，命名为 Pix2Vox。它从大规模三维数据集中学习几何先验，建立了从 RGB 空间到三维模型空间的映射关系。在 Pix2Vox 中，物体三维结构是使用体素表示的。在体素中，0 和 1 分别表示一个体素点是未被填充的和被填充的。Pix2Vox 核心模块如图2-1所示。首先，编码器将每张图像编码成一个特征向量；接着解码器从每个特征向量都恢复出一个粗略的三维体素模型；最后，基于 U-Net 的微调器通过残差连接（Skip Connections）形成一个残差网络，从而用于进一步优化重建结果。所提出的 Pix2Vox 包含 2 个版本：Pix2Vox-F 和 Pix2Vox-A。图2-2展示了 Pix2Vox-F 和 Pix2Vox-A 的网络结构。前者有更少的参数和更快的计算速度，后者拥有更多的参数并可以产生更精确的重建结果。其中，Pix2Vox-A 的算法描述如算法2-1所示。

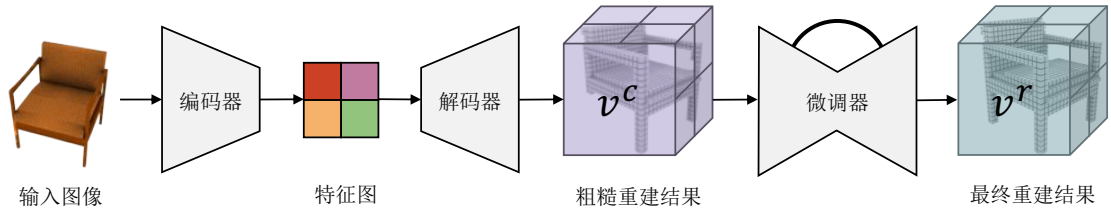



图 2-1 Pix2Vox 的总体框架图

Fig. 2-1 Overview of Pix2Vox

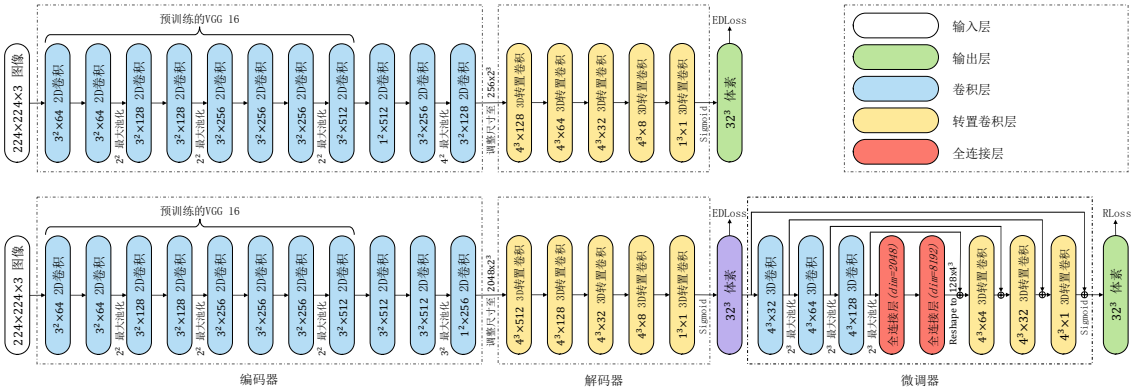



图 2-2 Pix2Vox-F 和 Pix2Vox-A 的网络结构图

Fig. 2-2 The network architecture of Pix2Vox-F and Pix2Vox-A

**（1）编码器**

编码器（Encoder）被用于从图像中抽取特征，并输出特征向量。每张输入图像都会被输入至一个在 ImageNet 上预训练的 VGG-16 网络[126] 的前 9 层卷积层，每个卷积层包含一个 Batch Normalization 和 ReLU 层，在经过这些层后，可以得到一个 $512 \times 28 \times 28$ 的特征向量。这些特征向量被输入至接下来的 3 组卷积层中，每个卷积层包含一个 2D 的卷积层，Batch Normalization 和 ELU 层。在 Pix2Vox-F 中，第一个卷积层的 Kernel Size 是 $1^2$，其他两个卷积层的 Kernel Size 是 $3^2$；卷积层的输出通道数分别是 512，256 和 128。在 Pix2Vox-F 中，这三个卷积层的 Kernel Size 分别为 $3^2$，$3^2$ 和 $1^2$；卷积层的输出通道数分别是 512，512 和 128。在第二个卷积层之后有一个最大池化层，它的 Kernel Size 在 Pix2Vox-F 和 Pix2Vox-A 分别是 $3^2$ 和 $4^2$。最终，编码器在 Pix2Vox-F 和 Pix2Vox-A 分别输出一个长度为 2048 和 16384 的特征向量。

**（2）解码器**

解码器（Decoder）被用于从特征向量中恢复三维模型。在 Pix2Vox-F 和 Pix2Vox-A 中，解码器由 5 个 3D 转置卷积层组成。具体来说，前 4 个转置卷积层中的卷积层的 Kernel Size 是 $4^3$，Stride 是 2，Padding 是 1。除此之外，前 4 层的每个转置卷积层

还包含 Batch Normalization 和 ReLU；在第 5 个转置卷积层中，Batch Normalization 和 ReLU 被 Sigmoid 替代。在 Pix2Vox-F 中，转置卷积层的输出通道数为 128，64，32，8 和 1。在 Pix2Vox-A 中，转置卷积层的输出通道数为 512，128，32，8 和 1。解码器最终会输出一个分辨率为 $32^3$ 的体素模型。

**（3）微调器**

微调器（Refiner）可以看作一个残差网络，用于修正三维体素中被错误恢复的体素值。它的设计基本遵循一个 3D 编码器-解码器的 U-Net 结构。受益于编码器和解码器之间的 U-Net 连接，在融合三维模型中的局部结构可以很好地保留。具体来说，微调器中的编码器包含 3 个 3D 卷积层，每个卷积层中包含 Batch Normalization、ReLU 和 Kernel Size 为 $2^3$ 的 Max Pooling 层。其中的每个 3D 卷积层的 Kernel Size 为 $4^3$，Padding 为 2。这三个卷积层的输出通道数分别为 32，64 和 128。在编码器之后，有 2 个全连接层，维度分别为 2048 和 8192。解码器由 3 个 3D 转置卷积层组成，其中的每个卷积层 Kernel Size 为 $4^3$，Padding 为 2，Stride 为 1。这三个转置卷积层的输出通道数分别为 64，32 和 1。除了最后一个转置卷积层使用了 Sigmoid，其余的转置卷积层中包含 Batch Normalization 和 ReLU。

**（4）损失函数**

二值交叉熵损失（Binary Cross Entropy Loss）用于训练整个网络，它定义为：

$$\ell = \frac{1}{N}\sum_{i=1}^{N}\left[gt_i \log(p_i) + (1 - gt_i)\log(1 - p_i)\right] \tag{2-1}$$

其中，$N$ 表示模型中体素点的数量。$p_i$ 和 $gt_i$ 分别表示预测出的三维结构和 Ground Truth。$\ell$ 的值越小，表明预测结果越接近 Ground Truth。

### 2.3.2 实验结果与分析

**（1）数据集**

ShapeNet[30] 是一个根据 WordNet 类别组织的三维 CAD 模型的集合。本文参考了 3D-R2N2[75] 的实验设置，使用了 ShapeNet 的一个子集，这个子集包含了来自 13 个类别的 44,000 个模型。具体而言，本文使用了 3D-R2N2 在 ShapeNet 上的渲染图像，对于每一个三维模型，它生成了 24 张分辨率为 $137 \times 137$ 的图像。

Pix3D[117] 提供了真实场景的图像和对应的三维模型，这些三维模型与真实图像精准地对应。这个数据集包含了来自 9 个类别的 395 个三维模型。每一个三维模型都对应了在多张在不同真实场景拍摄的图像。本文参考了 Pix3D 的实验设置，

表 2-1 在 ShapeNet 数据集上使用单视角单目彩色图像的三维物体重建的 IoU
Table 2-1 The IoU of 3D object reconstruction from a monocular RGB image of ShapeNet

| 类别 | 3D-R2N2[75] | OGN[29] | DRC[127] | PSGN[41] | Pix2Vox-F | Pix2Vox-A |
|---|---|---|---|---|---|---|
| 飞机 | 0.513 | 0.587 | 0.571 | 0.601 | 0.600 | **0.684** |
| 长椅 | 0.421 | 0.481 | 0.453 | 0.550 | 0.538 | **0.616** |
| 橱柜 | 0.716 | 0.729 | 0.635 | 0.771 | 0.765 | **0.792** |
| 汽车 | 0.798 | 0.828 | 0.755 | 0.831 | 0.837 | **0.854** |
| 椅子 | 0.466 | 0.483 | 0.469 | 0.544 | 0.535 | **0.567** |
| 显示器 | 0.468 | 0.502 | 0.419 | **0.552** | 0.511 | 0.537 |
| 灯具 | 0.381 | 0.398 | 0.415 | **0.462** | 0.435 | 0.443 |
| 扬声器 | 0.662 | 0.637 | 0.609 | **0.737** | 0.707 | 0.714 |
| 枪 | 0.544 | 0.593 | 0.608 | 0.604 | 0.598 | **0.615** |
| 沙发 | 0.628 | 0.646 | 0.606 | 0.708 | 0.687 | **0.709** |
| 桌子 | 0.513 | 0.536 | 0.424 | **0.606** | 0.587 | 0.601 |
| 电话 | 0.661 | 0.702 | 0.413 | 0.749 | 0.770 | **0.776** |
| 船舶 | 0.513 | **0.632** | 0.556 | 0.611 | 0.582 | 0.594 |
| 平均 | 0.560 | 0.596 | 0.545 | 0.640 | 0.634 | **0.661** |

使用了 2,894 未裁剪且未遮挡的真实场景中椅子的图像测试所提出的的方法。

**（2）度量指标**

Intersection over Union (IoU) 被用于评估输出结果的质量。IoU 的阈值被设置为 0.3，将输出的预测概率进行二值化并计算 IoU。更具体地说：

$$\mathrm{IoU} = \frac{\sum_{i,j,k} \mathrm{I}(p_{(i,j,k)} > t)\mathrm{I}(gt_{(i,j,k)})}{\sum_{i,j,k} \mathrm{I}\left[\mathrm{I}(p_{(i,j,k)} > t) + \mathrm{I}(gt_{(i,j,k)})\right]} \tag{2-2}$$

其中，$p_{(i,j,k)}$ 和 $gt_{(i,j,k)}$ 分别表示坐标为 $(i,j,k)$ 的点预测的输出和 Ground Truth。$\mathrm{I}(\cdot)$ 表示指示函数，$t$ 表示二值化的阈值。IoU 值越大表明重建结果的质量越好。

**（3）实现细节**

本文使用 PyTorch[128] 实现了所提出的方法①，并使用了一块 NVIDIA GTX 1080 Ti GPU 训练所提出的神经网络。网络输入图像的分辨率为 $224 \times 224$，输出的体素分辨率为 $32^3$。在训练时，Batch Size 被设置为 64，并使用了 $\beta_1 = 0.9$ 和 $\beta_2 = 0.999$ 的 Adam[129] 优化器。初始的学习率被设置为 $10^{-3}$，并在 150 个 Epoch 之后下降为原来的一半，训练一共持续 250 个 Epoch。

**（4）与现有方法的比较**

为保证对比的公平性，本文在测试时使用了相同的图片（除了 PSGN[41]）。尽管 PSGN 在训练时使用了更多的数据，Pix2Vox-A 依然表现出比它更好的性能。表2-

① 代码已开源：https://github.com/hzxie/Pix2Vox

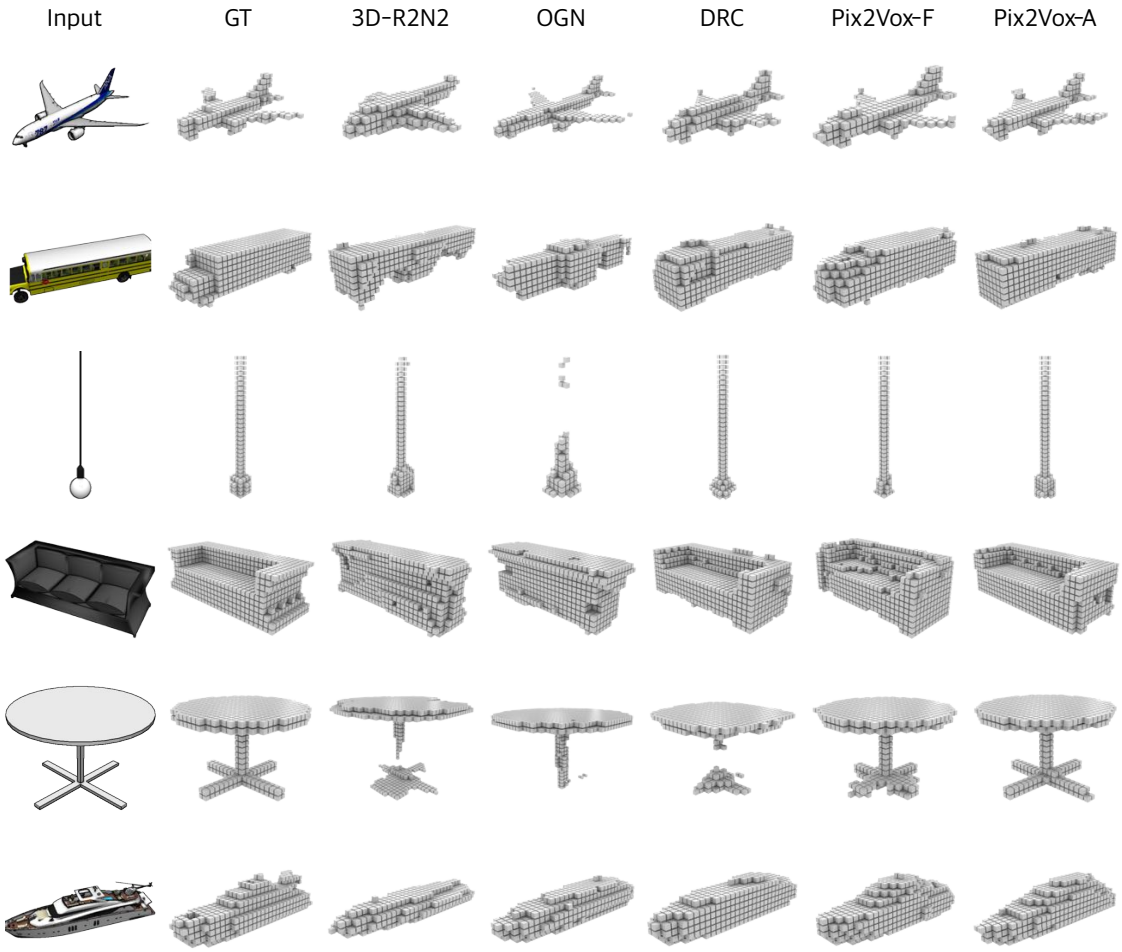



图 2-3 在 ShapeNet 数据集上使用单视角单目彩色图像的三维物体重建的结果

Fig. 2-3 The results of 3D object reconstruction from a monocular RGB image of ShapeNet

1展示了在 ShapeNet 数据集上使用单视角单目彩色图像恢复物体完整三维结构的结果。单视角重建实验结果表明，Pix2Vox-F 和 Pix2Vox-A 都显著地超过了其他方法。相比于 3D-R2N2，Pix2Vox-A 在 IoU 上有 18% 的性能提升。图2-3展示了一些有代表性的重建结果。Pix2Vox-F 和 Pix2Vox-A 都可以更好地恢复出物体的纤细部位，如台灯和桌子腿。和 Pix2Vox-F 相比，Pix2Vox-A 拥有更高维度的特征向量和更多的参数，也拥有更好的重建性能。

为了进一步验证所提出的方法在真实场景中的性能，本文在 Pix3D 数据集上进行了实验。由于 ShapeNet 不包含自然背景，因此并不能很好地泛化至真实场景拍摄的图像。Render for CNN[130] 被用于重新渲染 ShapeNet 中椅子类别的三维模型，并为每个三维模型生成了 60 张图像。此外，每张图像在训练时还使用了随机颜色和随机亮度扰动，并使用二维边界框对图像进行裁剪。在测试时，输入图像

表 2-2 在 Pix3D 数据集上使用单视角单目彩色图像的三维物体重建的 IoU
Table 2-2 The IoU of 3D object reconstruction from a monocular RGB image of the Pix3D dataset

| 方法 | IoU |
|---|---|
| 3D-R2N2[75] | 0.136 |
| DRC[127] | 0.265 |
| Pix3D (w/o Pose)[117] | 0.267 |
| Pix3D (w/ Pose)[117] | 0.282 |
| Pix2Vox-F | 0.271 |
| Pix2Vox-A | **0.288** |


图 2-4 在 Pix3D 数据集上使用单视角单目彩色图像的三维物体重建的结果
Fig. 2-4 The results of 3D object reconstruction from a monocular RGB image of Pix3D

被缩放至合适的尺寸。表 2-2和图 2-4展示了所提出的方法在 Pix3D 数据集上的重建结果。和其他方法相比，所提出的 Pix2Vox-A 方法可以更好地恢复真实场景中物体的三维结构。

图 2-5展示了各个方法在 ShapeNet 测试集上的推理时间、模型大小以及 IoU 的对比。其中，Pix2Vox-F 的参数量相比于 3D-R2N2 减少了 80%。此外，Pix2Vox-F

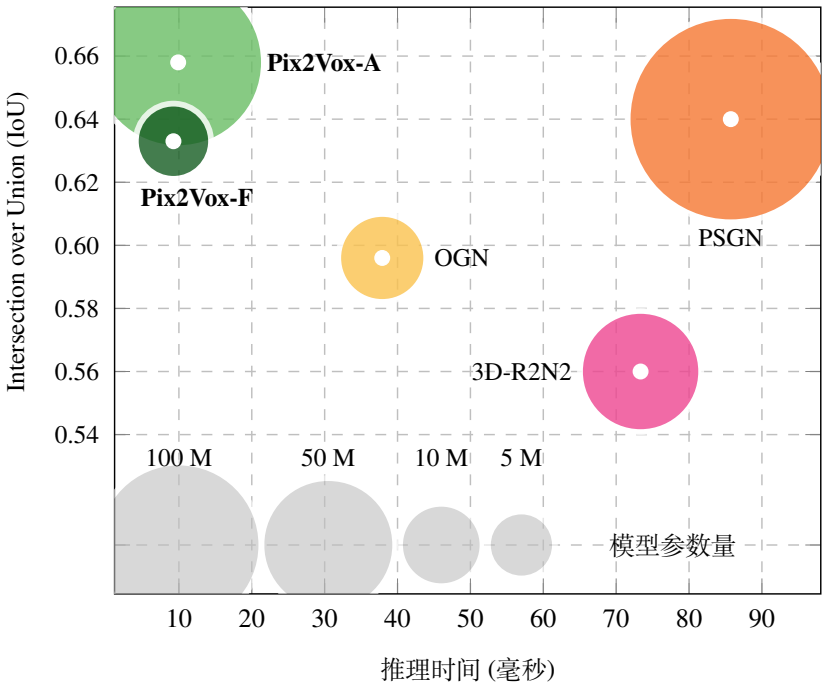



图 2-5 各个方法在 ShapeNet 数据集上的推理时间、模型参数量、IoU 的对比

Fig. 2-5 The comparison of forward inference time, model size, and IoU of all methods on ShapeNet

和 Pix2Vox-A 的推理速度大约是 3D-R2N2 的 8 倍，这些推理时间是在一台配备了一块 NVIDIA GTX 1080 Ti GPU 的台式机上测试得到的。

## 2.4 基于深度感知的双目彩色图像三维物体重建

### 2.4.1 模型与方法

为了从双目彩色图像中恢复物体的完整三维结构，本文提出了基于深度感知的三维物体重建方法。所提出的方法在重建物体完整三维结构时显式地估计了双目的视差和视图间稠密的特征匹配。视差图提供了物体 2.5 维的结构，因此可以更好地恢复物体三维结构的细节；视图间稠密的特征匹配可以更好地提供视图间的特征匹配关系，从而帮助推断物体的三维结构。它包含两个版本：*Stereo2Voxel* 和 *Stereo2Point*，分别用于从恢复单视角的双目彩色图像物体的三维体素和点云。它们都包含 3 个子网络：DispNet-B，CorrNet 和 RecNet，分别用于估计视差，寻找稠密的特征匹配关系和重建三维模型（如图2-6所示）。首先 DispNet-B 从一对双目图像中估计出两个视图的视差图，估计得到的视差图和原始 RGB 图像连接可得 RGB-D 图像。所得到的两个视图的 RGB-D 图像被送至 RecNet 的编码器中。CorrNet 被用于从两个视图中寻找稠密的特征匹配关系。最后，RecNet 的解码器被

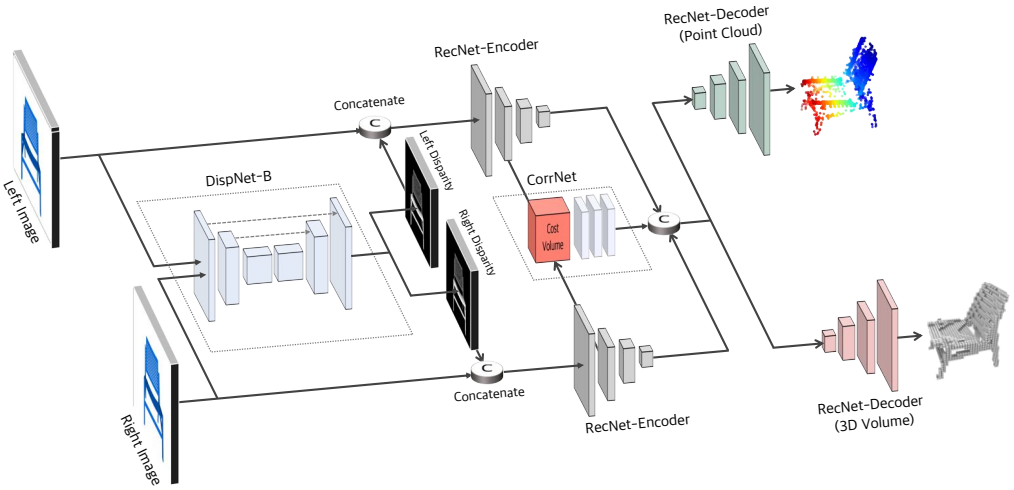



图 2-6 Stereo2Voxel 和 Stereo2Point 的总体框架图

Fig. 2-6 Overview of Stereo2Voxel and Stereo2Point

算法 2-2 Stereo2Voxel 算法

Algo.2-2 The Stereo2Voxel algorithm

**Input:** 单视角双目彩色图像 $\mathbf{I}^L, \mathbf{I}^R \in \mathbb{R}^{137\times137\times3}$

**Output:** 三维体素模型 $\hat{\mathbf{V}} \in \mathbb{R}^{32\times32\times32}$

1 给定输入双目图像 $\mathbf{I}^L, \mathbf{I}^R$，DispNet-B 输出视差图 $\hat{\mathbf{D}}^L, \hat{\mathbf{D}}^R \in \mathbb{R}^{137\times137\times3}$；

2 在通道维度拼接 $\mathbf{I}^L$ 和 $\hat{\mathbf{D}}^L$ 生成图像 $\mathbf{M}^L \in \mathbb{R}^{137\times137\times4}$；

3 在通道维度拼接 $\mathbf{I}^R$ 和 $\hat{\mathbf{D}}^R$ 生成图像 $\mathbf{M}^R \in \mathbb{R}^{137\times137\times4}$；

4 给定 $\mathbf{M}^L$，RecNet 编码器的第二、四个卷积层生成特征 $\mathbf{F}_2^L \in \mathbb{R}^{128\times34\times34}$ 和 $\mathbf{F}_4^L \in \mathbb{R}^{8192}$；

5 给定 $\mathbf{M}^R$，RecNet 编码器的第二、四个卷积层生成特征 $\mathbf{F}_2^R \in \mathbb{R}^{128\times34\times34}$ 和 $\mathbf{F}_4^R \in \mathbb{R}^{8192}$；

6 给定 $\mathbf{F}_2^L$ 和 $\mathbf{F}_2^R$，CorrNet 生成视图相关性向量 $\mathbf{C} \in \mathbb{R}^{4096}$；

7 拼接特征 $\mathbf{C}$，$\mathbf{F}_4^L$ 和 $\mathbf{F}_4^R$，生成特征 $\mathbf{F}_f \in \mathbb{R}^{20480}$，并调整尺寸至 $320 \times 4^3$；

8 给定特征 $\mathbf{F}_f$，RecNet 体素解码器通过 3 次上采样输出体素模型 $\hat{\mathbf{V}}$。

用于从特征向量中生成一个物体的三维体素或点云。Stereo2Voxel 和 Stereo2Point 的算法描述分别如算法2-2和2-3所示。

**（1）视差估计网络：DispNet-B**

包含编码器-解码器结构的 DispNet-B 被用于计算双目彩色图像的视差，如图2-7 (a) 所示。DispNet-B 中的编码器生成原图大小 $\frac{1}{8} \times \frac{1}{8}$ 的特征图。然后，DispNet-B 中的解码器通过 3 个转置卷积生成和输入图像相同大小的视差图。DispNet-B 同时

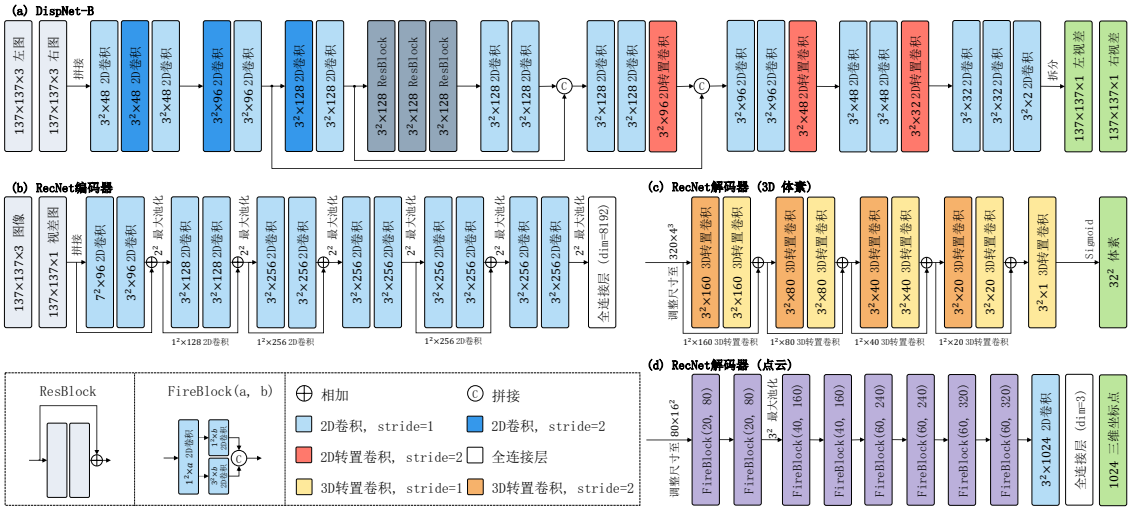



图 2-7 Stereo2Voxel 和 Stereo2Point 的网络结构图

Fig. 2-7 The network architecture of Stereo2Voxel and Stereo2Point

算法 2-3 Stereo2Point 算法

Algo.2-3 The Stereo2Point algorithm

**Input:** 单视角双目彩色图像 $\mathbf{I}^L, \mathbf{I}^R \in \mathbb{R}^{137\times137\times3}$

**Output:** 三维点云模型 $\hat{\mathbf{P}} \in \mathbb{R}^{1024\times3}$

1 给定输入双目图像 $\mathbf{I}^L$, 给定输入双目图像 $\mathbf{I}^L, \mathbf{I}^R$，DispNet-B 输出视差图 $\hat{\mathbf{D}}^L, \hat{\mathbf{D}}^R \in \mathbb{R}^{137\times137\times3}$；

2 在通道维度拼接 $\mathbf{I}^L$ 和 $\hat{\mathbf{D}}^L$ 生成图像 $\mathbf{M}^L \in \mathbb{R}^{137\times137\times4}$；

3 在通道维度拼接 $\mathbf{I}^R$ 和 $\hat{\mathbf{D}}^R$ 生成图像 $\mathbf{M}^R \in \mathbb{R}^{137\times137\times4}$；

4 给定 $\mathbf{M}^L$，RecNet 编码器的第二、四个卷积层生成特征 $\mathbf{F}_2^L \in \mathbb{R}^{128\times34\times34}$ 和 $\mathbf{F}_4^L \in \mathbb{R}^{8192}$；

5 给定 $\mathbf{M}^R$，RecNet 编码器的第二、四个卷积层生成特征 $\mathbf{F}_2^R \in \mathbb{R}^{128\times34\times34}$ 和 $\mathbf{F}_4^R \in \mathbb{R}^{8192}$；

6 给定 $\mathbf{F}_2^L$ 和 $\mathbf{F}_2^R$，CorrNet 生成视图相关性向量 $\mathbf{C} \in \mathbb{R}^{4096}$；

7 拼接特征 $\mathbf{C}$，$\mathbf{F}_4^L$ 和 $\mathbf{F}_4^R$，生成特征 $\mathbf{F}_f \in \mathbb{R}^{20480}$，并调整尺寸至 $80\times16^2$；

8 给定特征 $\mathbf{F}_f$，RecNet 点云解码器输出点云模型 $\hat{\mathbf{P}}$。

估计左右两个视图的视差，因为这相比单视图视差估计可以取得更好的结果[131]。此外，DispNet-B 的参数量只有 DispNet[61] 的 6%，因为它的网络中使用了更小的通道数。因此，DispNet-B 的效率非常高，前向传播速度大约是 DispNet 的 4 倍。

**（2）三维重建网络：RecNet**

RecNet 可以从输入的单视角双目彩色图像中生成三维体素或点云。卷积层之

间的残差连接（Residual Connections）可以加速网络的收敛[132]，受到这点的启发，RecNet 中的编码器使用了残差块（Residual Blocks）构建。部分编码器的残差块中引入了 $1\times1$ 的卷积层，用于解决卷积前后通道数变化的问题。根据三维表达形式的不同，RecNet 的解码器有两个版本：Stereo2Voxel 和 Stereo2Point，分别用于生成三维体素和点云，如图2-7(c) 和 (d) 所示。RecNet 解码器的输入包含三个部分：RecNet 编码器输出的左图和右图的特征向量以及 CorrNet 输出的特征向量，这三个特征向量被拼接成一个长度为 20480 的特征向量。

在 RecNet 中，生成三维体素的解码器包含 9 个转置卷积层，将特征图上采样至 $32^3$ 的尺寸。最后一个特征图会被传递至一个 Sigmoid 层，并输出三维体素中某个体素被填充的概率。和 RecNet 的编码器相似，其中也使用了残差链接以提高效率。生成点云的解码器由 8 个 Fire 块（Fire Block）[133] 和全连接层组成。和 PSGN[41] 一样，RecNet 最终生成一个大小为 $1024\times3$ 的矩阵，表示 1024 个点的三维坐标。为了减小解码器的参数量，若干个 Fire 块被使用以替换传统的转置卷积层。每个 Fire 块包含一个 Squeeze 卷积层，其中只包含 $1\times1$ 的卷积；其结果被作为 $1\times1$ 和 $3\times3$ 卷积层的输入。相比于 $3\times3$ 卷积层，采用这种设计的 RecNet 结构会包含更少的参数量。相比于 PSGN，RecNet 的参数量只有其 1/3。

**（3）视图关联网络：CorrNet**

如上所述，CorrNet 被用于左右视图的稠密特征匹配。稠密的特征匹配是通过一个 Cost Volume 计算的，其中 Cost Volume 保留了双目图像中所有的几何信息。和 GC-Net[134] 一样，Cost Volume 的尺寸是高度 × 宽度 ×(视差最大值 / 偏移)× 通道数，它是通过将左右视图每次移动一定的偏移堆叠产生的。所生成的 Cost Volume 可以让网络对左右视图的特征进行稠密的特征匹配。3D-CNN 被引入用于进一步提取特征的匹配关，。3D-CN 由 9 个通道数为 128 的 3D 卷积层组成，每个卷积层中都包含 Batch Normalization 和 ReLU 非线性层。第一个卷积层的 Kernel Size 为 $1\times1\times1$，除了第一个卷积层之外的其他卷积层的 Kernel Size 均为 $3\times3\times3$。网络的最后包含一个 Kernel Size 为 $1\times1\times1$ 的 3D 卷积层和一个 Kernel Size 为 $1\times1$ 的 2D 卷积层，这两层的输出通道数均为 1。最终的输出结果被送至一个全连接层并得到一个长度为 4096 维的向量。

**（4）损失函数**

训练过程使用了 3 个损失函数，分别用于监督视差、三维体素和点云。均方误差（Mean Square Error）损失函数被用于来度量所估计的视差和 Ground Truth 之

间的误差。它的定义如下：

$$\mathcal{L}_{disp} = \frac{1}{HW} \sum \left[ ||\hat{\mathbf{D}}^L - \mathbf{D}^L||^2 + ||\hat{\mathbf{D}}^R - \mathbf{D}^\mathbf{R}||^2 \right] \tag{2-3}$$

其中，$H$ 和 $W$ 分别表示视差图的高度和宽度。$\mathbf{D}^L$ 和 $\mathbf{D}^R$ 分别表示左右视图视差的 Ground Truth。$\hat{\mathbf{D}}^L$ 和 $\hat{\mathbf{D}}^R$ 分别表示左右视图视差的估计值。3D Binary Cross Entropy 被用于度量所生成的三维体素和 Ground Truth 的差异。它可以定义为：

$$\mathcal{L}_{vol} = \frac{1}{n_{vox}} \sum \left[ \mathbf{V} \log(\hat{\mathbf{V}}) + (1 - \mathbf{V}) \log(1 - \hat{\mathbf{V}}) \right] \tag{2-4}$$

其中，$n_{vox}$ 表示模型中体素点的数量。$\mathbf{V}$ 和 $\hat{\mathbf{V}}$ 分别表示三维体素的 Ground Truth 和估计值。和 PSGN[41] 一样，倒角距离（Chamfer Distance）被用于度量所生成的点云和 Ground Truth 的差异。它可以形式化地描述为：

$$\mathcal{L}_{cd} = \frac{1}{n_{gt}} \sum_{p \in \mathbf{P}} \min_{q \in \hat{\mathbf{P}}} ||p - q||_2^2 + \frac{1}{n_p} \sum_{p \in \hat{\mathbf{P}}} \min_{q \in \mathbf{P}} ||p - q||_2^2 \tag{2-5}$$

其中 $p$ 和 $q$ 分别为 $\mathbf{P}$ 和 $\hat{\mathbf{P}}$ 的一个点，$\mathbf{P} = \{(x_i, y_i, z_i)\}_{i=1}^{n_{gt}}$ 和 $\hat{\mathbf{P}} = \{(x_i, y_i, z_i)\}_{i=1}^{n_p}$ 分别表示 Ground Truth 和重建得到的点云。$n_{gt}$ 和 $n_p$ 分别表示 Ground Truth 和重建结果的点云中点的数量。

### 2.4.2 实验结果与分析

**（1）数据集**

StereoShapeNet 是本文使用 Blender ①从 ShapeNet 生成的一个大规模的双目图像三维物体重建数据集，包含了 1,052,976 对双目图像，以及两个视图的视差和深度图，和对应的三维模型，命名为 *StereoShapeNet*，如图 2-8所示。在生成数据的过程中，方位角（azimuth）被设定为 $\theta_{az} \in [0°, 360°)$，俯仰角（elevation）被设定为 $\theta_{el} \in [-30°, 30°]$。在 Blender 中，相机的焦距被设定为 35 毫米，双目相机的基线被设定为 130 毫米。生成的图像分辨率为 $224 \times 224$。和 3D-R2N2[75] 一样，本文为来自 ShapeNet 中 13 个主要类别的 44,000 个模型的每个模型生成了 24 张图像。

Driving 数据集[61] 中的自然场景是为了模拟 KITTI[106] 中的真实场景中行驶的汽车拍摄图像而构建的。构建 Driving 数据集时，场景中的三维模型来自于 ShapeNet 以及 3D Warehouse②。相机的基线被设定为 1 个 Blender 单位长度，其中一辆汽车的宽度约为 2 个 Blender 单位长度，这个设定和 KITTI 中汽车的宽度为

① https://www.blender.org

② https://3dwarehouse.sketchup.com/

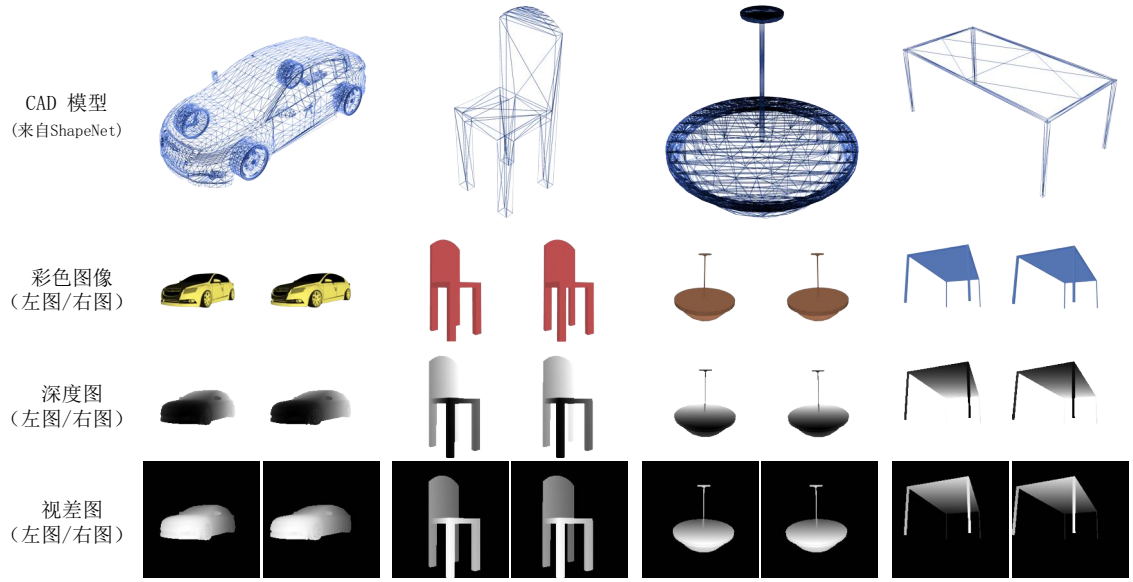



图 2-8 StereoShapeNet 数据集中图像和对应 CAD 模型的示例

Fig. 2-8 Examples of the images and the corresponding CAD models in StereoShapeNet

186 厘米，相机的基线为 54 厘米相仿。

**（2）度量指标**

为了评估所提出的方法的重建性能，本文在 *StereoShapeNet* 上和现有方法进行了对比。Intersection over Union (IoU) 被用于评估三维体素的重建质量。IoU 中的阈值被设置为 0.4，用于二值化输出的预测概率。其定义如公式2-2所示。IoU 的值越大，表明体素的重建结果和 Ground Truth 越接近。倒角距离被用于评估点云的重建质量，其定义如公式2-5所示。倒角距离的值越小，表明点云重建结果和 Ground Truth 越接近。

**（3）实现细节**

本文使用 PyTorch[128] 实现了所提出的方法①，并使用了使用了一块 NVIDIA GTX 1080 Ti GPU 训练所提出的神经网络。网络输入图像的分辨率为 $137 \times 137$，输出的体素分辨率为 $32^3$。和 PSGN[41] 一样，$n_p$ 和 $n_{gt}$ 的值分别被设置为 1024 和 16,384。在训练时，Batch Size 被设置为 20，并使用了 $\beta_1 = 0.9$ 和 $\beta_2 = 0.999$ 的 Adam[129] 优化器。初始的学习率被设置为 $10^{-4}$，并在 300 个 Epoch 之后下降为原来的一半，训练一共持续 500 个 Epoch。

**（4）与现有方法的比较**

所有对比的方法都在 *StereoShapeNet* 数据集上进行了微调（Fine-tune），并

① 代码已开源：https://github.com/hzxie/Stereo-3D-Reconstruction

表 2-3 在 StereoShapeNet 上使用单视角双目彩色图像的三维重建结果
Table 2-3 The 3D object reconstruction results from stereo RGB images of StereoShapeNet

| 类别 | 三维体素重建 (IoU, $32^3$ 分辨率) | | | | | 点云重建 (CD $\times 10^{-3}$, 1024 个点) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Matryoshka | Matryoshka* | Pix2Vox | LSM | Stereo2Voxel | PSGN | PSGN* | AtlasNet | AtlasNet* | Stereo2Point |
| 飞机 | 0.557 | 0.535 | 0.686 | 0.621 | **0.709** | 0.826 | 0.699 | 0.807 | 0.796 | **0.534** |
| 长椅 | 0.524 | 0.473 | 0.566 | 0.517 | **0.622** | 1.789 | 1.695 | 1.996 | 1.796 | **1.182** |
| 橱柜 | 0.766 | 0.763 | 0.754 | 0.691 | **0.784** | 2.360 | 1.853 | 1.756 | 1.692 | **1.229** |
| 汽车 | 0.827 | 0.810 | 0.811 | 0.796 | **0.830** | 1.295 | 0.882 | 1.045 | 1.036 | **0.779** |
| 椅子 | 0.559 | 0.514 | 0.604 | 0.595 | **0.669** | 2.004 | 1.594 | 1.837 | 1.858 | **1.267** |
| 显示器 | 0.635 | 0.614 | 0.586 | 0.547 | **0.692** | 2.815 | 2.238 | 2.386 | 2.146 | **1.356** |
| 灯具 | 0.424 | 0.411 | 0.449 | 0.469 | **0.521** | 3.973 | 3.038 | 4.142 | 4.118 | **3.001** |
| 扬声器 | 0.697 | 0.727 | 0.658 | 0.670 | **0.701** | 3.868 | 2.691 | 2.839 | 2.869 | **2.124** |
| 枪 | 0.540 | 0.557 | 0.652 | 0.682 | **0.690** | 0.790 | 0.763 | 0.818 | 0.874 | **0.524** |
| 沙发 | 0.702 | 0.679 | 0.714 | 0.651 | **0.770** | 2.625 | 2.086 | 1.664 | 1.656 | **1.199** |
| 桌子 | 0.559 | 0.503 | 0.570 | 0.566 | **0.635** | 1.889 | 1.500 | 1.892 | 1.916 | **1.337** |
| 电话 | 0.759 | 0.847 | 0.831 | 0.694 | **0.866** | 1.445 | 1.158 | 1.156 | 1.250 | **0.896** |
| 船舶 | 0.587 | 0.595 | 0.558 | 0.592 | **0.645** | 2.029 | 1.495 | 1.712 | 1.524 | **1.027** |
| 平均 | 0.626 | 0.603 | 0.652 | 0.632 | **0.702** | 1.916 | 1.493 | 1.704 | 1.689 | **1.185** |

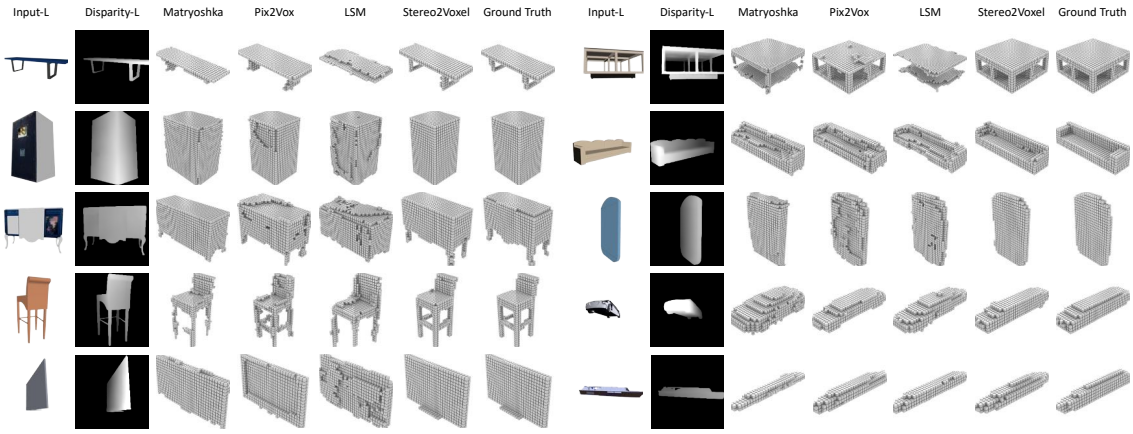



图 2-9 在 StereoShapeNet 数据集上使用单视角双目彩色图像的三维体素重建结果
Fig. 2-9 The results of 3D volumes reconstruction from a pair of stereo RGB images of ShapeNet

用相同的测试图片比对这些方法的重建结果。Stereo2Voxel 和一个多视图重建方法 Pix2Vox-A、一个单视图重建方法 Matryoshka Network[135] 进行了对比。Stereo2Voxel 还和一个 Multi-view Stereo 的方法 LSM[76] 进行了对比，以进一步体现所提出的方法的卓越的重建能力。Stereo2Point 和 PSGN[41] 以及 AtlasNet[136] 进行了对比。单视图三维物体重建方法只使用左视图作为输入。为了和单视图重建方法进行更加公平的对比，这些单视图重建方法被拓展为以堆叠的左右视图作为网络的输入，这些拓展后的方法被命名为 Matryoshka*，PSGN* 和 AtlasNet*。表2-3列出了在 *StereoShapeNet* 上的重建结果。实验结果表明，Stereo2Voxel 和 Stereo2Point 的性能超越了所有单视图和多视图的重建方法。相比 Multi-view Stereo 的方法 LSM，Stereo2Voxel 依然取得了更好的重建性能，尽管 LSM 使用了相机外参作为额外的输入。图2-9和

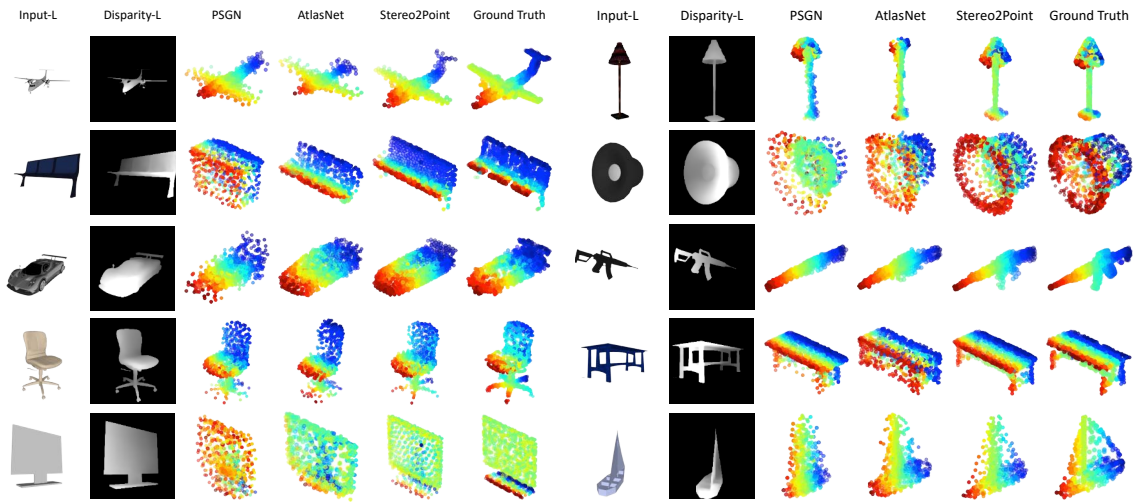



图 2-10 在 StereoShapeNet 数据集上使用单视角双目彩色图像的点云重建结果

Fig. 2-10 The results of point cloud reconstruction from a pair of stereo RGB images of ShapeNet

图2-10展示了这些方法在 *StereoShapeNet* 数据集上的重建结果。相比于其他方法，Stereo2Voxel 和 Stereo2Point 可以更好地恢复出物体的细节（如桌腿和凳腿）。

为了验证所提出的方法在自然图像上的性能，Stereo2Voxel 和 Stereo2Point 在 Driving 数据集上和 Matryoshka、Pix2Vox、PSGN、AltasNet 等方法进行了对比。所有的方法均使用了 StereoShapeNet 数据集中的汽车类别进行了微调，并在在训练时随机选择 SUN 数据集[137] 中的图像作为背景，同时使用了随机颜色和亮度作为数据增强。为了更好地让所提出的方法泛化至自然场景中，FlyingThings3D[61] 被用于预训练 DispNet-B。在测试中，本文从 Driving 数据集中选择了未截断和未遮挡的图像，使用二维边界框对图像中最大的汽车对象进行裁剪，并将其缩放至网络的输入大小。图 2-11展现了 Driving 数据集上一些比较有代表性的重建结果。除了 Pix2Vox，其他对比的方法对于这三组图片会产生相似的重建结果，这近一步证实了现有的单视角重建的方法在重建时不探索重建后的物体和输入图像间的几何关系，从而更倾向于输出一个平均的形状以减少重建误差[138]。由于自然场景中背景和光照变化通常较为复杂，因此从自然场景中恢复物体的三维结构通常充满挑战。尽管如此，和其他方法相比，所提出的 Stereo2Voxel 和 Stereo2Point 可以更好地恢复物体的骨架结构。

**（5）消融实验**

本节验证了 Stereo2Voxel 和 Stereo2Point 中两个关键组件的有效性：DispNet-B 和 CorrNet。

**DispNet-B** 在 Stereo2Voxel 和 Stereo2Point 中被用于从双目彩色图像中恢复左

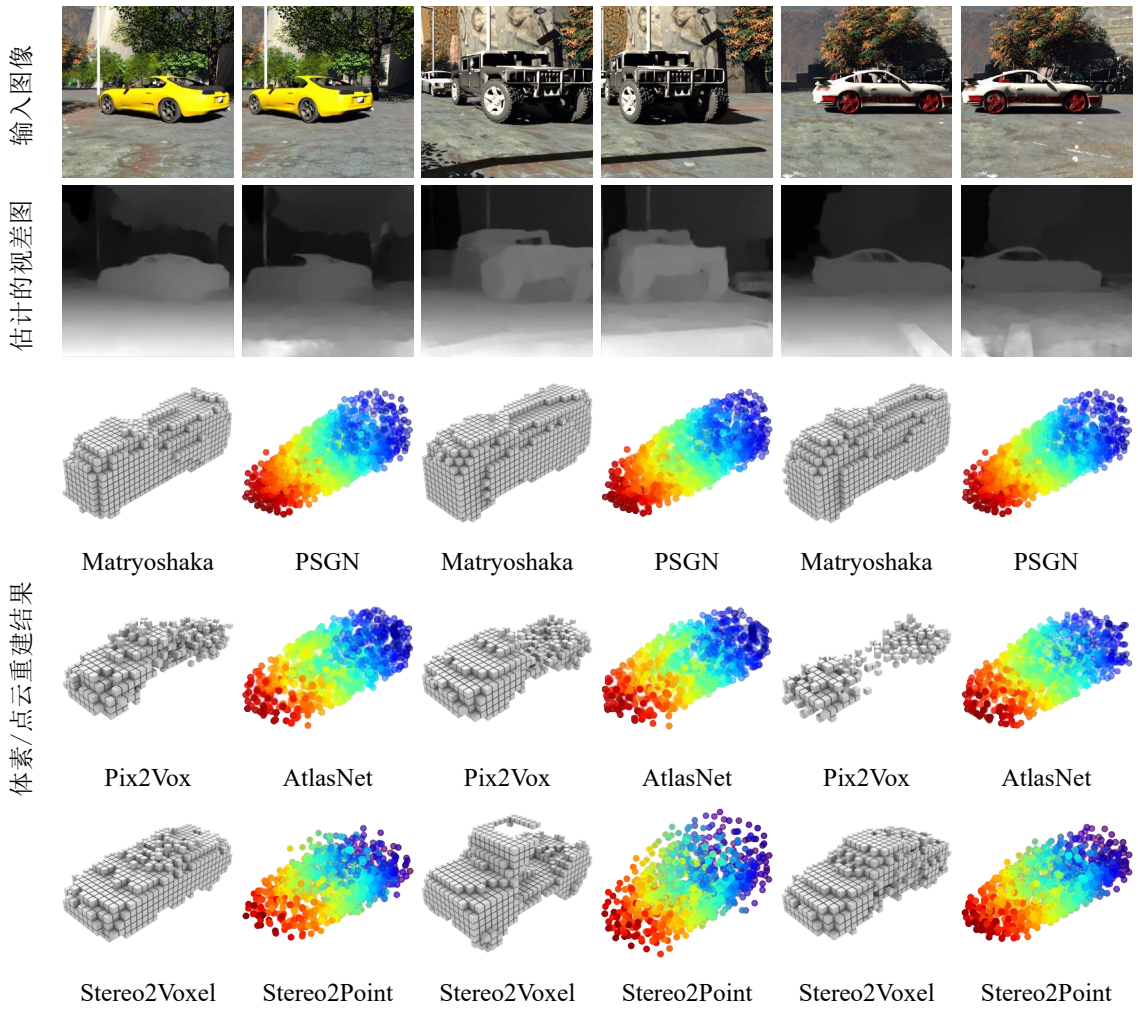



图 2-11 在 Driving 数据集上使用单视角双目彩色图像的三维重建结果

Fig. 2-11 The results of 3D object reconstruction from a pair of stereo images of Driving

右图的视差图。为了证明视差在双目三维物体重建中的重要性，DispNet-B 被从 Stereo2Voxel 和 Stereo2Point 中移除。在这种设置下，双目彩色图像直接被输入至 RecNet 的编码器中。表 2-4的结果表明，移除 DispNet-B 后，Stereo2Point 所生成点云的倒角距离增加了 16.4%，而 Stereo2Voxel 所生成体素的 IoU 下降至 0.678。

为了进一步对比 DispNet-B 和其他视差估计方法（包括 DispNet[61]、GA-Net[139] 和 AANet[140]）的性能，本文比较了它们的参数量、推理时间以及在 Flying Things 3D 子集和 StereoShapeNet 上的终点误差（End-Point-Error，EPE），其中 Flying Things 3D 子集仅包含在 Cleanpass 中视差值小于 96 个像素的图像。如表 2-5所示，所提出的 DispNet-B 的终点误差和 DispNet[61] 是可比较的，并略低于 GA-Net[139]。然而，DispNet-B 的参数量仅为 DispNet[61] 和 GA-Net[139] 的 6% 和 38%。本文在同一台配备 NVIDIA GTX 1080 Ti GPU 的电脑上对比了这些方法的推理时间。其结果

表 2-4 移除 DispNet-B 和 CorrNet 前后的性能对比
Table 2-4 The comparison before and after removing DispNet-B and CorrNet

| DispNet-B | CorrNet | Stereo2Voxel (IoU) | Stereo2Point (CD $\times 10^{-3}$) |
|---|---|---|---|
| ✓ | ✓ | **0.702** | **1.185** |
| ✓ | × | 0.690 | 1.284 |
| × | ✓ | 0.678 | 1.379 |
| × | × | 0.651 | 1.570 |

表 2-5 不同方法参数量、推理时间和终点误差的对比
Table 2-5 The comparison of the numbers of parameters, inference time, and the endpoint error (EPE)

| 方法 | DispNet-B | DispNet[61] | GA-Net[139] | AANet[140] |
|---|---|---|---|---|
| 参数量 (M) | **2.54** | 39.45 | 6.58 | 4.03 |
| 推理时间 (秒) | **0.018/对** | 0.063 | 6.999 | 0.068 |
| FlyingThings 3D (EPE) | 1.292 | 1.157 | 0.515 | **0.477** |
| StereoShapeNet (EPE) | 0.096 | 0.092 | 0.089 | **0.085** |

表 2-6 使用不同方法估计的视差图对于重建性能的影响
Table 2-6 The comparison of reconstruction results with the different disparity maps

| 方法 | Stereo2Voxel (IoU) | Stereo2Point (CD $\times 10^{-3}$) |
|---|---|---|
| SGBM[60] | 0.680 | 1.282 |
| DispNet-B | 0.702 | 1.185 |
| DispNet[61] | 0.702 | 1.184 |
| GA-Net[139] | 0.705 | 1.178 |
| AANet[140] | **0.706** | **1.170** |

表明，DispNet-B 的速度分别是 DispNet[61] 和 GA-Net[139] 的 7 倍和 778 倍，并且 DispNet-B 可以在一次推理中同时完成对于左右两个视图的视差估计。表 2-6展示了视差图的准确度对于重建结果的影响，其结果表明，视差图质量越高，重建结果会略有提高。

**CorrNet** 在 Stereo2Voxel 和 Stereo2Point 中被用于查找双目图像中特征的匹配关系。为了验证 CorrNet 的有效性，CorrNet 在 Stereo2Voxel 和 Stereo2Point 中被移除。表 2-4的结果表明，移除 CorrNet 后，Stereo2Point 所生成点云的倒角距离由 1.185 上升至 1.284，而 Stereo2Voxel 所生成体素的 IoU 则由 0.702 下降至 0.690。同时移除 DispNet-B 和 CorrNet 将使得重建性能进一步下降：Stereo2Point 所生成点云的倒角距离上升了 32.5%, 而 Stereo2Voxel 所生成体素的 IoU 下降了 7.3%。

## 2.5 本章小结

本章针对单目彩色相机和双目彩色相机提出了单源单视三维物体重建方法。对于单目彩色相机，本文提出了基于几何先验的单目彩色图像三维物体重建方法，方法被命名为 Pix2Vox-F 和 Pix2Vox-A；其中 Pix2Vox-F 具有比现有方法都更快的速度，而 Pix2Vox-A 具有比现有方法都更好的性能。由于目前双目彩色相机在智能手机和机器人上被广泛使用，本文针对双目彩色相机提出了基于深度感知的双目彩色图像三维物体重建方法，使得在重建物体完整三维结构时充分利用已知的 2.5 维结构推断未知的三维结构；方法被命名为 Stereo2Voxel 和 Stereo2Point，被分别用于重建体素和点云。所提出的方法可以解决 Shape from X 无法恢复物体不可见部分三维结构的问题。在 ShapeNet、StereoShapeNet、Pix3D 和 Driving 数据集上的实验结果表明，相比于其他基于学习的单视角三维物体重建方法，所提出的方法具有更好的重建性能和泛化性能。

# 第 3 章 几何结构感知单源单视深度图像三维物体重建

## 3.1 引言

随着计算机视觉技术的逐步发展，采用深度相机进行物体识别和场景建模的相关应用越来越多。相比于提供颜色信息的彩色相机，深度相机可以获得每个点到深度相机所在的垂直平面的距离值，提供三维点的空间坐标信息。假定像素坐标为 $(u, v)$ 的点在三维空间中的坐标为 $(x, y, z)$，则有：

$$\begin{aligned} x &= \frac{uz}{f_x} \\ y &= \frac{vz}{f_y} \end{aligned} \tag{3-1}$$

其中 $f_x$ 和 $f_y$ 为相机焦距。如图 3-1所示，深度图像的场景可以通过公式 3-1反投影至可见部分的三维结构（即 2.5 维结构）。尽管 2D 卷积神经网络在图像领域取得了巨大突破，但直接将其应用至深度图像难以显式对物体的 2.5 维结构进行建模，造成物体几何结构的信息丢失。因此，基于深度图像的三维重建技术应该转换为点云补全问题，即通过 2.5 维结构推测物体完整的三维结构。

在过去的几年里，大量工作使用卷积神经网络从彩色图像恢复物体的体素结构。因为点云数据是不规则且无序的，因此无法像图像数据一样直接使用卷积神经网络。现有的大部分方法[141–147] 通过体素化的方法将点云转换为体素并对其应

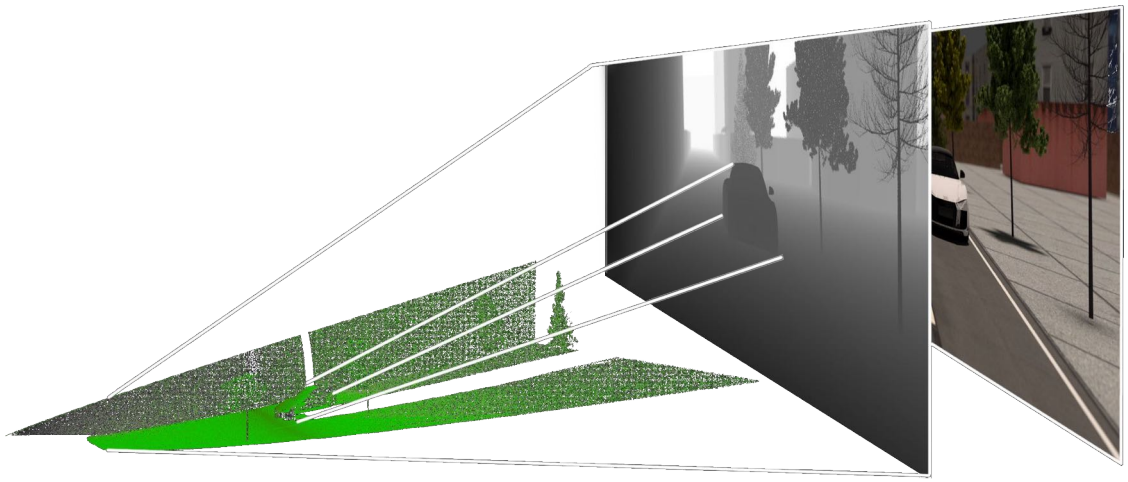


图 3-1 深度图像反投影示意图
Fig. 3-1 The illustration of unprojecting a depth image

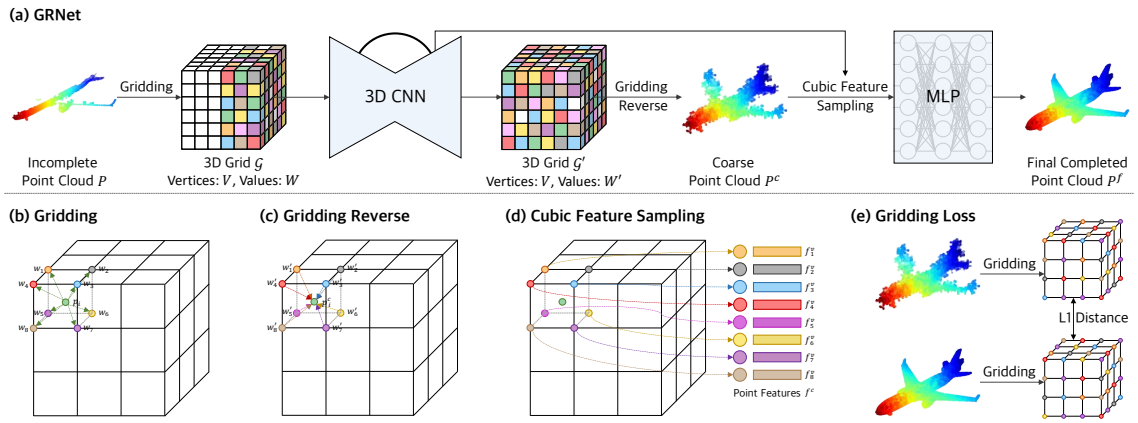



图 3-2 GRNet 的总体框架图

Fig. 3-2 Overview of GRNet

用 3D 卷积神经网络，然而体素化将会不可逆的丢失物体的集合信息。另外一些方法[44,45,148] 使用了多层感知机直接处理点云数据。然而，这些方法通常通过最大池化聚集全部点的信息，然而这种方式并不能充分考虑点与点之间连接性及其上下文信息。最近，也有一些工作[149,150] 尝试使用图神经网络（Graph Convolutional Network, GCN）建立点云中点和点的连接性。建立图依赖最近邻算法（K-nearest neighbor, KNN），然而最近邻算法对于点的密度较为敏感[151]。在点云分割领域，一些工作使用更通用的卷积对点云的空间关系进行显式建模。SPLATNet[152] 和 InterpConv[153] 分别在高维的 Lattice 空间和相邻点插值得到的三维网格中进行卷积操作。然而这些方法有一个强的假设：点云中点的坐标在卷积前后是不改变的，然而对于点云补全任务而言，这并不适用。

为了解决上述问题，3D Grid 被引入作为中间表示以规则化无序的点云，从而对点云的几何结构和上下文信息进行显式建模。基于 3D Grid，本文进一步提出了网格化残差网络（Gridding Residual Network, GRNet），如图 3-2所示。除了 3D 卷积神经网络（3D CNN）和多层感知机（MLP），本文还提出了 3 个可微分的操作：网格化（Gridding）、逆网格化（Gridding Reverse）和立方级特征采样（Cubic Feature Sampling）。网格化残差网络是通过如下步骤从 2.5 维结构生成物体完整的三维结构的。首先，对于点云中的每一个点，它都会落在一个 3D Grid Cell 中，而每个 3D Grid Cell 包含 8 个顶点，网格化通过一个三线性插值函数将这个点插值到这 8 个顶点上，从而显式地对点云的几何关系进行了建模。接着，包含残差连接的 3D 卷积神经网络被用于学习上下文感知和空间感知的特征，并用于补全输入点云中缺失的部分。然后，逆网格化为每个 3D Grid Cell 生成一个点，这个点的坐标是通过

这个 3D Grid Cell 顶点的权值和坐标加权得到的，这个操作实现了 3D Grid 到点云的转换。最后，立方级特征采样将点云所在的 3D Grid Cell 8 个顶点的特征抽取出来，并输入至多层感知机中生成完整点云。

本章的贡献可以归纳为以下三点：

• 本文首次引入了 3D Grid 作为点云的中间表示，从而将无序的点云规则化为一个规则的结构，同时保留了点云的结构信息和上下文信息。

• 本文提出了针对点云补全任务的 GRNet，并设计了 3 个可微分的操作：网格化、逆网格化和立方级特征采样。

• 在 ShapeNet、Completion3D 和 KITTI 数据集上的实验结果表明，所提出的 GRNet 均超越了现有的方法。

## 3.2 相关工作

根据点云补全和重建方法的网络结构，现有的方法可以被大致归纳为三类：基于多层感知机的方法、基于图的方法和基于卷积的方法。

**（1）基于多层感知机的方法**

由于先驱工作 PointNet[154] 的简洁性和有效性，有一系列的工作使用多层感知机实现点云的理解[155,156] 与重建[44,148]。这些方法使用多个多层感知机独立处理点云中的每个点，并使用一个对称的函数（如最大池化）将这些特征聚集。然而这些方法并没有充分考虑点云的几何结构。PointNet++[157] 和 TopNet[45] 使用一个层级的结构考虑点云的集合信息。为了减少多层感知机造成的结构信息丢失，AtlasNet[136] 和 MSN[158] 通过估计一组表面的参数用于恢复某个物体完整的点云。

**（2）基于图的方法**

将点云中的每个点看作图中的顶点，基于图的方法将为相邻的点生成一条边，从而表征点云中点和点之间的关系。这些方法通常在相邻点上进行图卷积，并且使用池化生成一个新的图，同时聚集相邻点的特征。和基于多层感知机的方法相比，基于图的方法考虑了点云的几何关系。DGCNN[149] 在特征空间构建了一个图，并在网络的每一层动态更新这个图。LDGCNN[159] 移除了 DGCNN 中的变换网络（Transformation Network）并且将不同层级的特征连接起来，从而减少了模型的大小并提高了性能。受到 DGCNN 的启发，Hassani 等人[160] 引入了多尺度的图卷积网络学习点和形状的特征，并使用自监督的策略学习分类和重建。DCG[150] 在 DGCNN 的基础上编码局部的连接，并逐渐由粗糙恢复精细的点云结构。

算法 3-1 GRNet 算法
Algo.3-1 The GRNet algorithm

**Input:** 不完整点云模型 $\mathbf{P} \in \mathbb{R}^{n\times 3}$，其中 n 为输入点的数量

**Output:** 完整点云模型 $\mathbf{P}^f \in \mathbb{R}^{16384\times 3}$

1 给定输入点云 $\mathbf{P}$，网格化根据公式 3-3生成 3D Grid $\mathcal{G} =< V, W >$；

2 给定 $W$，三维卷积神经网络输出 $W'$ 和前 3 个卷积层输出的特征集合 $\mathcal{F} = \{\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3\}$；

3 给定 3D Grid $\mathcal{G}' =< V, W' >$，逆网格化根据公式 3-9生成粗糙点云 $\mathbf{P}^c \in \mathbb{R}^{2048\times 3}$；

4 给定 $\mathbf{P}^c$ 和 $\mathcal{F}$，立方级特征采样根据公式 3-14为每个 $\mathbf{P}^c$ 中的每个点寻找对应的特征 $\mathbf{F}^c \in \mathbb{R}^{2048\times 1792}$；

5 给定 $\mathbf{P}^c$ 和 $\mathbf{F}^c$，多层感知机输出最终的点云 $\mathbf{P}^f$。

**（3）基于卷积的方法**

早期的工作主要是用三维卷积网络在体素上进行 3D 卷积[141,142,161]。然而将点云转化为体素将引入不可逆的量化误差，从而丢失点云的细节[162]。目前没有直接在不规则点云使用卷积进行点云补全的工作。在现有点云理解的工作中[153,163–166]，一些工作将卷积应用于点云离散化生成的三维网格（3D Grid）中。Hua 等人[163]定义了在规则三维网格上的卷积操作，并对落入同一个网格中的点使用相同的权重。PointCNN[166] 通过 $\chi$ 变换实现了满足排列不变性的卷积。除了将卷积神经网络应用于离散空间，最近的一些工作[147,151,152,167–171] 直接将卷积应用于连续空间。Thomas 等人[151] 在 KPConv 中同时使用刚性和可变形的卷积算子，这意味着它可以针对输入点云的不同区域调整其内核的形状。和基于图的方法相比，基于卷积的方法对于点云中密度的变化更加的有效和鲁棒[153]。

## 3.3 基于网格化残差网络的单目深度图像三维物体重建

### 3.3.1 模型与方法

为了在点云补全的过程中更好地对点云的几何结构和上下文信息进行显式建模，本文提出了基于网格化残差网络的三维物体重建方法，命名为 GRNet。所提出的 GRNet 通过 Coarse-to-Fine 的策略从不完整的点云恢复完整点云。如图3-2所示，它有 5 个模块组成：网格化，3D 卷积神经网络，逆网格化，立方级特征采样

和多层感知机。给定一个不完整的点云 $P$ 作为输入，网格化将会生成一个 3D Grid $\mathcal{G} =< V, W >$，其中 $V$ 和 $W$ 分别表示 $\mathcal{G}$ 的顶点集和值集。接着，$W$ 被送入 3D 卷积神经网络并输出 $W'$。再接着，逆网格化根据 3D Grid $\mathcal{G}' =< V, W' >$ 生成一个粗略的点云 $P^c$。然后，立方级特征采样从 $P^c$ 提取对应的特征 $F^c$。最后多层感知机 $P^c$ 及其对应的特征 $F^c$ 作为输出并输出最终的点云 $P^f$。GRNet 的算法描述如算法 3-1所示。

**（1）网格化**

在 GRNet 中，3D Grid 被引入将无序的点云规则化到一个有序的结构上，并进一步提出了网格化。具体来说，点云 $P = \{p_i\}_{i=1}^{n}$ 被转换为一个规则的 3D Grid $\mathcal{G} =< V, W >$，其中 $p_i \in \mathbb{R}^3, V = \{v_i\}_{i=1}^{N^3}$，$W = \{w_i\}_{i=1}^{N^3}, v_i \in \{(-\frac{N}{2}, -\frac{N}{2}, -\frac{N}{2}), \ldots, (\frac{N}{2} - 1, \frac{N}{2} - 1, \frac{N}{2} - 1)\}$，$w_i \in \mathbb{R}$，$n$ 表示点云 $P$ 中点的个数，$N$ 表示 3D Grid $\mathcal{G}$ 的分辨率。在这个操作中，点云空间结构被完整地保留。如图3-2 (b) 所示，3D Grid 中的每一个小立方体为一个 Cell。显然地，每个 Cell 包含 8 个顶点。对于 3D Grid $\mathcal{G}$ 中的每个 Cell 的顶点 $v_i = (x_i^v, y_i^v, z_i^v)$，其邻域点集 $\mathcal{N}(v_i)$ 为这个顶点相邻的 8 个 Cell 中所包含的点。邻域点集 $\mathcal{N}(v_i)$ 中的点 $p = (x, y, z) \in \mathcal{N}(v_i)$ 的满足条件：(1) $p \in P$，(2) $x_i^v - 1 < x < x_i^v + 1$，(3) $y_i^v - 1 < y < y_i^v + 1$，(4) $z_i^v - 1 < z < z_i^v + 1$。在体素化过程中，顶点 $v_i$ 的值 $w_i$ 可通过如下公式得到：

$$w_i = \begin{cases} 0 & \forall p \notin \mathcal{N}(v_i) \\ 1 & \exists p \in \mathcal{N}(v_i) \end{cases} \tag{3-2}$$

然而，体素化过程中会引入量化误差，从而造成重建结果中细节的缺失。此外，体素化是不可微分的，所以无法用于点云重建中。如图3-2 (b) 所示，给定一个顶点 $v_i$ 及其邻域点集 $\mathcal{N}(v_i)$ 网格化将通过如下公式计算顶点 $v_i$ 的值 $w_i$：

$$w_i = \sum_{p \in \mathcal{N}(v_i)} \frac{w(v_i, p)}{|\mathcal{N}(v_i)|} \tag{3-3}$$

其中，$|\mathcal{N}(v_i)|$ 表示顶点 $v_i$ 对应邻接点集点的个数。特别地，$w_i = 0$ 当 $|\mathcal{N}(v_i)| = 0$。其中 $w(v_i, p)$ 可以定义为

$$w(v_i, p) = (1 - |x_i^v - x|)(1 - |y_i^v - y|)(1 - |z_i^v - z|) \tag{3-4}$$

根据公式 3-3和3-4可知，在反向传播过程中，对应坐标点对于 $w(v_i, p)$ 的梯度为：

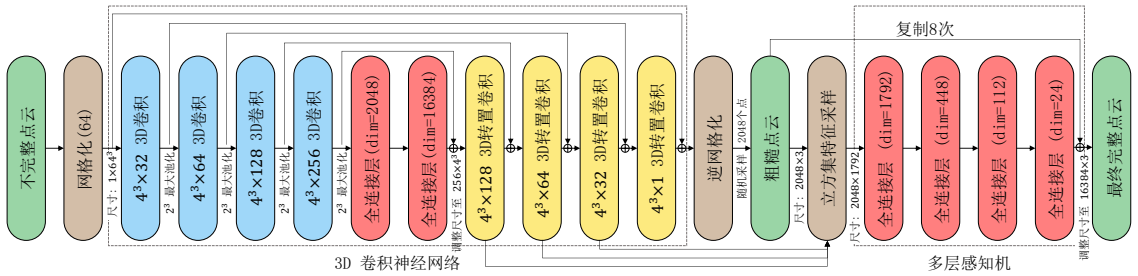



图 3-3 GRNet 的网络结构图

Fig. 3-3 The network architecture of GRNet

$$\frac{\partial w_i}{\partial x}=\begin{cases}-\frac{1}{|\mathcal{N}(v_i)|}\sum_{p\in\mathcal{N}(v_i)}(1-|y_i^v-y|)(1-|z_i^v-z|), & x>x_i^v\\ \frac{1}{|\mathcal{N}(v_i)|}\sum_{p\in\mathcal{N}(v_i)}(1-|y_i^v-y|)(1-|z_i^v-z|), & x\le x_i^v\end{cases} \tag{3-5}$$

其中 $x$ 和 $x_i^v$ 分别表示点 $p$ 和顶点 $v_i$ 在 x 轴的坐标。类似可得：

$$\frac{\partial w_i}{\partial y}=\begin{cases}-\frac{1}{|\mathcal{N}(v_i)|}\sum_{p\in\mathcal{N}(v_i)}(1-|x_i^v-x|)(1-|z_i^v-z|), & y>y_i^v\\ \frac{1}{|\mathcal{N}(v_i)|}\sum_{p\in\mathcal{N}(v_i)}(1-|x_i^v-x|)(1-|z_i^v-z|), & y\le y_i^v\end{cases} \tag{3-6}$$

$$\frac{\partial w_i}{\partial z}=\begin{cases}-\frac{1}{|\mathcal{N}(v_i)|}\sum_{p\in\mathcal{N}(v_i)}(1-|x_i^v-x|)(1-|y_i^v-y|), & z>z_i^v\\ \frac{1}{|\mathcal{N}(v_i)|}\sum_{p\in\mathcal{N}(v_i)}(1-|x_i^v-x|)(1-|y_i^v-y|), & z\le z_i^v\end{cases} \tag{3-7}$$

其中 $y$ 和 $y_i^v$ 分别表示点 $p$ 和顶点 $v_i$ 在 y 轴的坐标；$z$ 和 $z_i^v$ 分别表示点 $p$ 和顶点 $v_i$ 在 z 轴的坐标。

**（2）3D 卷积神经网络**

含有残差连接（Skip Connections）的 3D 卷积神经网络被用于实现补全点云的缺失部分，它的设计主要遵循 U-Net 的思想。给定 $W$，3D 卷积神经网络可以形式化表示为：

$$W'=\mathrm{3DCNN}(W) \tag{3-8}$$

其中 $W'=\{w_i'|w_i'\in\mathbb{R}\}_{i=1}^{N^3}$。

如图3-3所示，3D 卷积神经网络的编码器包含 4 个卷积层，每个卷积层的 Kernel Size 为 $4^3$，Padding 为 2，并且跟随着 Batch Normalization 和 Leaky ReLU 激活函数和一个 Kernel Size 为 $2^3$ 的 Max Pooling 层。这些卷积层的输出通道数分别为 32，64，128 和 256。编码器之后包含 2 个用于建立全局的特征关系的全连接层，输出向量的维度分别为 2048 和 16384。3D 卷积神经网络的解码器包含 4 个转置卷积层，

每个转置卷积层的 Kernel Size 为 $4^3$，Padding 为 2，并且跟随着 Batch Normalization 和 ReLU 激活函数。

**（3）逆网格化**

如图3-2 (c) 所示，逆网格化被用于从 3D Grid $\mathcal{G}' =< V, W' >$ 生成粗糙点云 $P^c = \{p_i^c\}_{i=1}^m$，其中 $p_i^c \in \mathbb{R}^3$，$m$ 表示粗糙点云中点的数量。对于每一个 3D Grid Cell，逆网格化将通过加权这个 Cell 中 8 个顶点的值和坐标的方式生成一个点 $p_i^c$，可形式化地描述为：

$$p_i^c = \frac{\sum_{\theta \in \Theta^i} w'_\theta v_\theta}{\sum_{\theta \in \Theta^i} w'_\theta} \tag{3-9}$$

其中，$\{v_\theta | \theta \in \Theta^i\}$ 和 $\{w'_\theta | \theta \in \Theta^i\}$ 分别表示这 8 个顶点的值集和顶点集，$\Theta^i = \{\theta_j^i\}_{j=1}^8$ 表示第 $i$ 个 3D Grid Cell 中顶点的索引。特别地，对于满足条件 $\sum_{\theta \in \Theta^i} w'_\theta = 0$ 的 Cell，逆网格化将不生成点 $p_i^c$。

令 $x_i^c$ 为 $p_i^c$ 在 x 轴的坐标，可形式化描述为：

$$x_i^c = \frac{\sum_{\theta \in \Theta^i} w'_\theta x_\theta^v}{\sum_{\theta \in \Theta^i} w'_\theta} \tag{3-10}$$

在反向传播计算梯度时，$x_i^c$ 对 $w'_\theta$ 的偏导数可以通过如下公式计算：

$$\begin{aligned} \frac{\partial x_i^c}{\partial w'_\theta} &= \frac{x_\theta^v}{\sum_{\theta \in \Theta^i} w'_\theta} - \frac{\sum_{\theta \in \Theta^i} w'_\theta x_\theta^v}{\left(\sum_{\theta \in \Theta^i} w'_\theta\right)^2} \\ &= \frac{x_\theta^v}{\sum_{\theta \in \Theta^i} w'_\theta} - \frac{1}{\sum_{\theta \in \Theta^i} w'_\theta} \cdot x_i^c \\ &= \frac{x_\theta^v - x_i^c}{\sum_{\theta \in \Theta^i} w'_\theta} \end{aligned} \tag{3-11}$$

类似可得：

$$\frac{\partial y_i^c}{\partial w'_\theta} = \frac{y_\theta^v - y_i^c}{\sum_{\theta \in \Theta^i} w'_\theta} \tag{3-12}$$

$$\frac{\partial z_i^c}{\partial w'_\theta} = \frac{z_\theta^v - z_i^c}{\sum_{\theta \in \Theta^i} w'_\theta} \tag{3-13}$$

**（4）立方级特征采样**

为了获取点云的上下文信息，立方级特征采样为粗糙点云 $P^c$ 生成对应的特征 $F^c = \{f^c\}_{i=1}^m$，以助于后续的多层感知机恢复点云的细节，如图3-2 (d) 所示。令 $\mathcal{F} = \{f_1^v, f_2^v, \ldots, f_{t^3}^v\}$ 表示 3D 卷积神经网络的特征图，其中 $f_i^v \in \mathbb{R}^c$，$t^3$ 表示特征

图的大小。对于粗糙点云 $P^c$ 中的每一个点 $p_i^c$，它的特征 $f_i^c$ 可以通过如下公式计算得到：

$$f_i^c = [f_{\theta_1^i}^v, f_{\theta_2^i}^v, \ldots, f_{\theta_8^i}^v] \tag{3-14}$$

其中 $[\cdot]$ 表示拼接（Concatenation）操作。$\{f_{\theta_j^i}^v\}_{j=1}^8$ 表示 $p_i^c$ 所在的 3D Grid Cell 的 8 个顶点的特征。在 GRNet 中，立方级特征采样被应用于 3D 卷积神经网络的前 3 个转置卷积层的特征图上。为了减少特征冗余，立方级特征采样仅从 $P^c$ 随机选择 2048 个点，并生成大小为 $2048 \times 1792$ 的特征图。

在反向传播计算梯度时，$f_{\theta_j^i}^v$ 的偏导数可以通过如下公式求得：

$$\frac{\partial f_{i,j}^c}{\partial f_{\theta_j^i}^v} = 1 \tag{3-15}$$

其中 $j \in \{1, 2, \ldots, 8\}$，$f_{i,j}^c$ 表示 $f_i^c$ 中的第 $j$ 个元素。

由于 $\lfloor \cdot \rfloor$ 和 $\lceil \cdot \rceil$ 是不可微分的，因此 $x_i^c$、$y_i^c$、$z_i^c$ 0 的偏导数为 0[172]，可以形式化描述为：

$$\frac{\partial f_{i,j}^c}{\partial x_i^c} = 0 \tag{3-16}$$

$$\frac{\partial f_{i,j}^c}{\partial y_i^c} = 0 \tag{3-17}$$

$$\frac{\partial f_{i,j}^c}{\partial z_i^c} = 0 \tag{3-18}$$

**（5）多层感知机**

多层感知机被用于从粗糙点云恢复物体的细节。对于粗糙点云 $P^c$ 中的每一个点，多层感知机会估计出 8 个偏移量，这个偏移量表示最终的点云相对于这个粗糙点云中的点的偏移。多层感知机将粗糙点云 $P^c$ 和对应的特征 $F^c$ 作为输入，并输出最终的完整点云 $P^f = \{p_i^f\}_{i=1}^k$。该过程可以形式化描述为：

$$P^f = \mathrm{MLP}(F^c) + \mathrm{Tile}(P^c, r) \tag{3-19}$$

其中，$p_i^f \in \mathbb{R}^3$，$k$ 表示 $P^f$ 中点的数量。Tile 重复 $r$ 次 $P^c$ 生成一个大小为 $rm \times 3$ 的矩阵。在 GRNet 中，$r$ 被设定为 8。多层感知机包含 4 个的输出长度分别为 1792，448，112 和 24 的全连接层，输出包含 16384 个点的点云。

**（6）网格化损失**

现有的方法在训练神经网络时主要使用倒角距离（Chamfer Distance）[41] 作为损失函数。这个损失函数惩罚了在预测结果中远离 Ground Truth 的点，然而这并不能保证所预测的点可以保留物体的几何结构。因此，很多网络倾向于输出一个平均的形状以减小倒角距离，但这会导致重建结果中细节的丢失[43,52]。由于点云的无序性，点云补全的方法很难使用图像中的 L1/L2 损失以及体素中的二值交叉熵。而在网格化的帮助下，无序的点云可被映射至规则的 3D Grid，如图3-2 (e) 所示。因此，基于网格化的损失函数被提出，命名为网格化损失，它定义为两个 3D Grid 值集的 L1 距离。令 $\mathcal{G}_{pred} =< V^{pred}, W^{pred} >$ 和 $\mathcal{G}_{gt} =< V^{gt}, W^{gt} >$ 分别表示通过网格化从预测和 Ground Truth 生成的 3D Grid，其中 $W^{pred} \in \mathbb{R}^{N_G^3}, W^{gt} \in \mathbb{R}^{N_G^3}$，$N_G$ 表示这两个 3D Grid 的分辨率。则，网格化损失可以定义为：

$$\mathcal{L}_{Gridding}(W^{pred}, W^{gt}) = \frac{1}{N_G^3} \sum ||W^{pred} - W^{gt}|| \tag{3-20}$$

### 3.3.2 实验结果与分析

**（1）数据集**

ShapeNet 针对点云补全的数据集是由 PCN[44] 渲染 ShapeNet[30] 数据集生成的，其中包含了来自 8 个类别的 30,974 个三维模型。Ground Truth 的点云是通过均匀采样多边形网格上的点生成的，每个模型包含 16,384 个点。输入的不完整点云是通过反投影深度图像至三维空间得到的。为了更公平地对比所有的方法，本文使用了和 PCN 一致的训练集/验证集/测试集划分。

Completion3D Benchmark[45] 的训练集和验证集分别包含 28,974 和 800 个样本。和 PCN 生成的 ShapeNet 不同，每个 Ground Truth 的点云中仅包含 2,048 个点。

KITTI[173] 包含真实场景中的激光雷达扫描序列。该序列是由型号为 Velodyne HDL-64E 的 64 线激光雷达以每秒 10 帧扫描频率获得的。本节的实验使用了由 PCN[44] 预处理的数据。具体而言，每一帧里的汽车都使用了三维边界框将它从激光雷达扫描中抽取出来，从而生成 2,401 个不完整的点云。这些不完整的点云通常高度稀疏，并且不包含完整点云的 Ground Truth。

**（3）实现细节**

本文使用 PyTorch[128] 和 CUDA 实现了所提出的方法①，并使用了两块 NVIDIA

① 代码已开源：https://github.com/hzxie/GRNet

表 3-1 在 ShapeNet 上使用单视角深度图像的三维物体重建的倒角距离（$\times 10^{-3}$）
Table 3-1 The CD ($\times 10^{-3}$) of 3D object reconstruction from a depth image of ShapeNet

| 方法 | 飞机 | 橱柜 | 汽车 | 椅子 | 灯具 | 沙发 | 桌子 | 船舶 | 平均 |
|---|---|---|---|---|---|---|---|---|---|
| AtlasNet[136] | 1.753 | 5.101 | 3.237 | 5.226 | 6.342 | 5.990 | 4.359 | 4.177 | 4.523 |
| PCN[44] | **1.400** | 4.450 | **2.445** | 4.838 | 6.238 | 5.129 | 3.569 | 4.062 | 4.016 |
| FoldingNet[174] | 3.151 | 7.943 | 4.676 | 9.225 | 9.234 | 8.895 | 6.691 | 7.325 | 7.142 |
| TopNet[45] | 2.152 | 5.623 | 3.513 | 6.346 | 7.502 | 6.949 | 4.784 | 4.359 | 5.154 |
| MSN[158] | 1.543 | 7.249 | 4.711 | 4.539 | 6.479 | 5.894 | 3.797 | 3.853 | 4.758 |
| GRNet | 1.531 | **3.620** | 2.752 | **2.945** | **2.649** | **3.613** | **2.552** | **2.122** | **2.723** |

表 3-2 在 ShapeNet 上使用单视角深度图像的三维物体重建的 F-Score@1%
Table 3-2 The F-Score@1% of 3D object reconstruction from a depth image of ShapeNet

| 方法 | 飞机 | 橱柜 | 汽车 | 椅子 | 灯具 | 沙发 | 桌子 | 船舶 | 平均 |
|---|---|---|---|---|---|---|---|---|---|
| AtlasNet[136] | 0.845 | 0.552 | 0.630 | 0.552 | 0.565 | 0.500 | 0.660 | 0.624 | 0.616 |
| PCN[44] | 0.881 | **0.651** | **0.725** | 0.625 | 0.638 | 0.581 | 0.765 | 0.697 | 0.695 |
| FoldingNet[174] | 0.642 | 0.237 | 0.382 | 0.236 | 0.219 | 0.197 | 0.361 | 0.299 | 0.322 |
| TopNet[45] | 0.771 | 0.404 | 0.544 | 0.413 | 0.408 | 0.350 | 0.572 | 0.560 | 0.503 |
| MSN[158] | **0.885** | 0.644 | 0.665 | 0.657 | 0.699 | 0.604 | **0.782** | 0.708 | 0.705 |
| GRNet | 0.843 | 0.618 | 0.682 | **0.673** | **0.761** | **0.605** | 0.751 | **0.750** | **0.708** |

GTX TITAN Xp GPU 训练所提出的神经网络。在训练时，Batch Size 被设置为 32，并使用了 $\beta_1 = 0.9$ 和 $\beta_2 = 0.999$ 的 Adam[129] 优化器。初始的学习率被设置为 $10^{-4}$，并在 50 个 Epoch 之后下降为原来的一半，训练一共持续 150 个 Epoch。

**（2）度量指标**

与 PSGN[41] 和 TopNet[45] 相同，本文使用了 Chamfer Distance 作为度量指标之一，它可以定义为：

$$\mathrm{CD} = \frac{1}{n_{\mathcal{T}}} \sum_{t \in \mathcal{T}} \min_{r \in \mathcal{R}} ||t - r||_2^2 + \frac{1}{n_{\mathcal{R}}} \sum_{r \in \mathcal{R}} \min_{t \in \mathcal{T}} ||t - r||_2^2 \tag{3-21}$$

其中 $\mathcal{T} = \{(x_i, y_i, z_i)\}_{i=1}^{n_{\mathcal{T}}}$ 和 $\mathcal{R} = \{(x_i, y_i, z_i)\}_{i=1}^{n_{\mathcal{R}}}$ 分别表示预测输出和 Ground Truth。

正如 Tatarchenko 等人[138] 所指出的，Chamfer Distance 可能在一些场景中并不能很好地反映重建结果的质量。遵循 Tatarchenko 等人[138] 的建议，本文使用 F-Score@1% 作为额外的度量指标。它的定义如下：

$$\text{F-Score}(d) = \frac{2P(d)R(d)}{P(d) + R(d)} \tag{3-22}$$

其中 $P(d)$ 和 $R(d)$ 分别表示给定距离阈值 $d$ 的精度和召回率，它们可定义为：

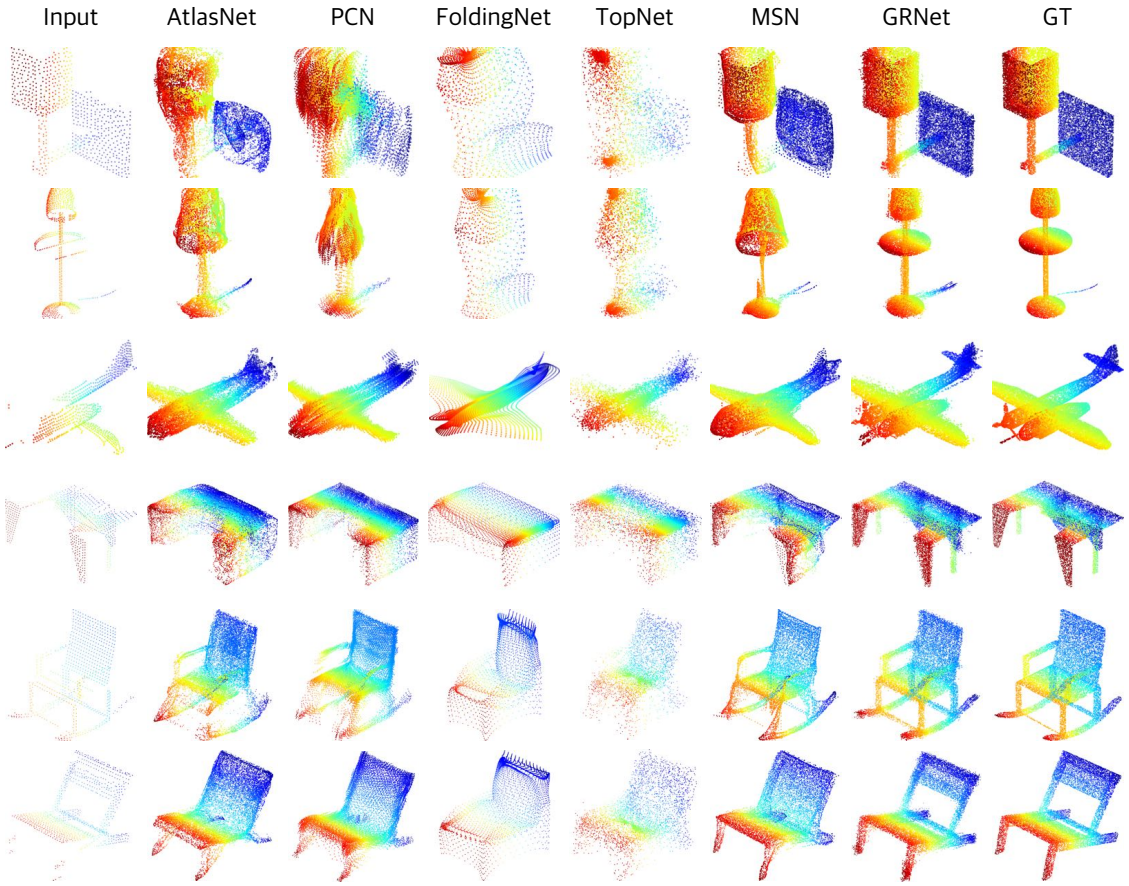



图 3-4 在 ShapeNet 数据集上使用单视角深度图像的三维物体重建结果

Fig. 3-4 The results of 3D object reconstruction from a depth image of ShapeNet

$$P(d) = \frac{1}{n_{\mathcal{R}}} \sum_{r \in \mathcal{R}} \left[ \min_{t \in \mathcal{T}} ||t - r|| < d \right] \tag{3-23}$$

$$R(d) = \frac{1}{n_{\mathcal{T}}} \sum_{t \in \mathcal{T}} \left[ \min_{r \in \mathcal{R}} ||t - r|| < d \right] \tag{3-24}$$

**（4）与现有方法的比较**

本文在 ShapeNet 数据集上和其他方法进行了对比。为了公平对比所有的方法，所有方法均生成了包含 16,384 个点的点云。AtlasNet[136] 通过生成一组参数化表示的表面以表示点云，并在这些表面上均匀采样 16,384 个点。FoldingNet[174] 作为 PCN[44] 的基线方法，将 $128 \times 128$ 的 2D Grid 映射至至 3D Grid。TopNet[45] 使用了一个树状结构的解码器考虑了点云的拓扑结构，由于这种结构的可扩展性，可以调整节点的数量和特征向量的大小使得其生成 16,384 个点。MSN[158] 通过由粗糙到精细的策略生成了包含 8,192 个点的点云；为了获得包含 16,384 个点的点云，可以取两次的推理结果的并集。表 3-1和3-2的结果表明，所提出的 GRNet 在倒角距离

表 3-3 在 Completion 3D 上使用单视角深度图像的三维物体重建的倒角距离（$\times 10^{-4}$）
Table 3-3 The CD ($\times 10^{-4}$) of 3D object reconstruction from a depth image of Completion 3D

| 方法 | 飞机 | 橱柜 | 汽车 | 椅子 | 灯具 | 沙发 | 桌子 | 船舶 | 平均 |
|---|---|---|---|---|---|---|---|---|---|
| AtlasNet[136] | 10.36 | 23.40 | 13.40 | 24.16 | 20.24 | 20.82 | 17.52 | 11.62 | 17.77 |
| FoldingNet[174] | 12.83 | 23.01 | 14.88 | 25.69 | 21.79 | 21.31 | 20.71 | 11.51 | 19.07 |
| PCN[44] | 9.79 | 22.70 | 12.43 | 25.14 | 22.72 | 20.26 | 20.27 | 11.73 | 18.22 |
| TopNet[45] | 7.32 | 18.77 | 12.88 | 19.82 | 14.60 | 16.29 | 14.89 | 8.82 | 14.25 |
| SA-Net[47] | 5.27 | **14.45** | **7.78** | 13.67 | 13.53 | **14.22** | 11.75 | 8.84 | 11.22 |
| SoftPoolNet[175] | **4.89** | 18.86 | 10.17 | 15.22 | 12.34 | 14.87 | 11.84 | 6.48 | 11.90 |
| GRNet | 6.13 | 16.90 | 8.27 | **12.23** | **10.22** | 14.93 | **10.08** | **5.86** | **10.64** |

和 F-Score@1% 的度量指标上均超越了所有的对比方法。图 3-4展示了在 ShapeNet 上使用单视角深度图像的三维重建结果。相比于其他方法，GRNet 可以更好地恢复物体的细节（如椅子和台灯）。

除了 ShapeNet，本文还在 Completion 3D Benchmark 上和其他方法进行了对比。验证集上最佳性能的模型被用于预测测试集中的 1,184 个物体的完整三维结构。由于 Completion 3D Benchmark 要求点云仅包含 2,048 个点，因此 GRNet 输出的点云被进行了降采样。在线排行榜的结果①如表 3-3所示。可见，所提出的 GRNet 显著超越了其他方法，并位居第一名。

为了进一步验证所提出的方法对于真实场景中激光雷达扫描的有效性，本文尝试补全了 KITTI 数据集中不完整的汽车。ShapeNet 数据集的生成是通过将深度图像反投影至三维空间，而 KITTI 数据集中的激光雷达扫描中的点云可能会非常稀疏。为了更加公平地与其他方法对比，所有对比的方法都在 ShapeNetCars 数据集（即 ShapeNet 中的汽车类别）上进行了微调（Fine-tune）。在测试时，输入的点云均通过三维边界框进行了归一化，从而使得在 ShapeNetCars 上训练的模型可以充分利用所学到的先验知识。因为 KITTI 数据集并没有提供 Ground Truth，所有方法的点云补全结果均使用一致性（Consistency）和均匀性（Uniformity）进行度量。一致性（Consistency）是在 PCN 中使用的，它定义为在不同帧之间同一辆车平均的倒角距离。令 $\mathcal{R}_{t_i}^{j}$ 表示第 $j$ 辆车在 $t_i$ 时刻的补全结果，则一致性可以形式化描述为：

$$\text{Consistency} = \frac{1}{n_f - 1} \sum_{i=2}^{n_f} \text{CD}(\mathcal{R}_{t_{i-1}}^{j}, \mathcal{R}_{t_i}^{j}) \tag{3-25}$$

PU-GAN[176] 中使用均匀性度量点云分布的均匀程度，它可以形式化描述为：

① https://completion3d.stanford.edu/results

表 3-4 在 KITTI 数据集上使用激光雷达扫描点云的三维物体重建结果
Table 3-4 The of 3D object reconstruction from a single-view depth image of Completion 3D

| 方法 | 一致性 ($\times 10^{-3}$) | 不同 $p$ 值下的均匀性 | | | | |
|---|---|---|---|---|---|---|
| | | 0.4% | 0.6% | 0.8% | 1.0% | 1.2% |
| AtlasNet[136] | 0.700 | 1.146 | 1.005 | 0.874 | 0.761 | 0.686 |
| PCN[44] | 1.557 | 3.662 | 5.812 | 7.710 | 9.331 | 10.823 |
| FoldingNet[174] | 1.053 | 1.245 | 1.303 | 1.262 | 1.162 | 1.063 |
| TopNet[45] | 0.568 | 1.353 | 1.326 | 1.219 | 1.073 | 0.950 |
| MSN[158] | 1.951 | 0.822 | 0.675 | 0.523 | 0.462 | 0.383 |
| GRNet | **0.313** | **0.632** | **0.572** | **0.489** | **0.410** | **0.352** |

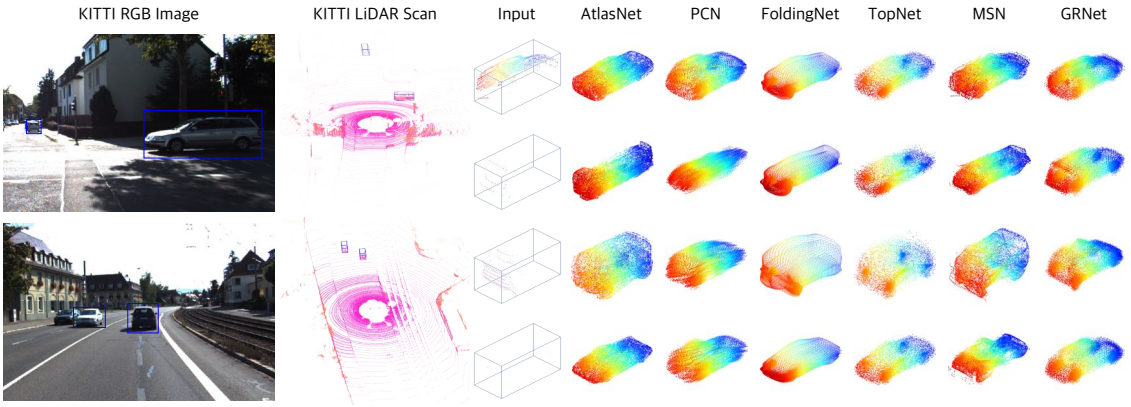



图 3-5 在 KITTI 数据集上使用激光雷达扫描点云的三维物体重建结果
Fig.3-5 The results of 3D object reconstruction from LiDAR scans of the KITTI dataset

$$\text{Uniformity}(p) = \frac{1}{M}\sum_{i=1}^{M} \text{U}_{\text{imbalance}}(S_i)\text{U}_{\text{clutter}}(S_i) \tag{3-26}$$

其中 $S_i(i = 1, 2, \ldots, M)$ 表示从 $\mathcal{R}$ 中通过最远点采样法（Farthest Point Sampling）和半径为 $\sqrt{p}$ 的球查询（Ball Query）获得的一组点的集合。$\text{U}_{\text{imbalance}}$ 和 $\text{U}_{\text{clutter}}$ 分别考虑了全局和局部分布的均匀性，它们分别定义为：

$$\text{U}_{\text{imbalance}}(S_i) = \frac{(|S_i| - \hat{n})^2}{\hat{n}} \tag{3-27}$$

其中 $\hat{n} = p|\mathcal{R}|$ 表示 $S_i$ 中期望的点的数量；

$$\text{U}_{\text{clutter}}(S_i) = \frac{1}{|S_i|}\sum_{j=1}^{|S_i|} \frac{(d_{i,j} - \hat{d})^2}{\hat{d}} \tag{3-28}$$

其中 $d_{i,j}$ 表示相对于 $S_i$ 中第 $j$ 个点的最近距离，$\hat{d}$ 在 $S_i$ 服从均匀分布时近似等于 $\sqrt{\frac{2\pi p}{|S_i|\sqrt{3}}}$[176]。表 3-4展示了对于 KITTI 数据集中激光雷达扫描的点云补全结果。这些实验结果表明 GRNet 在一致性和均匀性上超越了其他的方法。得益于网格化和

表 3-5 网格化在不同分辨率下的倒角距离、F-Score@1%、参数量和反向传播时间
Table 3-5 The Chamfer Distance, F-Score@1%, numbers of parameters, and backward time on ShapeNet with different resolutions of 3D grids generated by Gridding

| 分辨率 | CD ($\times 10^{-4}$) | | F-Score@1% | | 参数量 | 反向传播时间 |
|---|---|---|---|---|---|---|
| | 粗糙 | 精细 | 粗糙 | 精细 | (M) | (毫秒) |
| $32^3$ | 23.339 | 5.943 | 0.329 | 0.549 | 69.54 | 64 |
| $64^3$ | **11.259** | **2.723** | 0.340 | 0.708 | 76.70 | 100 |
| $128^3$ | 12.383 | 2.732 | **0.366** | **0.712** | 76.77 | 302 |

表 3-6 立方级特征采样在以不同的特征图作为输入所产生点云的倒角距离、F-Score@1%、多层感知机的参数量和反向传播时间
Table 3-6 The Chamfer Distance, F-Score@1%, and numbers of parameters of MLPs on ShapeNet with different features maps feeding into Cubic Feature Sampling

| 特征图的尺寸 | | | CD | F-Score | 参数量 | 反向传播时间 |
|---|---|---|---|---|---|---|
| $128 \times 8^3$ | $64 \times 16^3$ | $32 \times 32^3$ | ($\times 10^{-4}$) | @1% | (M) | (毫秒) |
| | | | 11.375 | 0.343 | 0 | 72 |
| | | ✓ | 2.922 | 0.640 | 0.11 | 80 |
| | ✓ | ✓ | 2.805 | 0.686 | 0.96 | 88 |
| ✓ | ✓ | ✓ | **2.723** | **0.708** | 4.07 | 100 |

表 3-7 网格化损失在不同分辨率下的倒角距离、F-Score@1%、参数量和反向传播时间
Table 3-7 The Chamfer Distance and F-Score@1% on ShapeNet with different resolutions of 3D grids generated by Gridding Loss

| 分辨率 | CD ($\times 10^{-4}$) | | F-Score@1% | | 反向传播时间 |
|---|---|---|---|---|---|
| | 粗糙 | 精细 | 粗糙 | 精细 | (毫秒) |
| 未使用 | 11.259 | 4.460 | 0.340 | 0.624 | 86 |
| $64^3$ | 10.275 | 3.427 | 0.364 | 0.672 | 92 |
| $128^3$ | **9.324** | **2.723** | **0.386** | **0.708** | 100 |

逆网格化，GRNet 可以对于空间结构更加敏感，因此可以更好地保持两帧之间的一致性。如图 3-5所示，输入的高度稀疏点云已经很难识别出其中的物体；相比之下，完整的点云可以提供更多的三维结构信息。此外，实验结果也表明本节所提出的方法可以产生更合理的形状补全结果。

**（5）消融实验**

GRNet 的性能提升主要来自于网格化、立方级特征采样和网格化损失。为了进一步证明各个组件的有效性，本文使用不同的参数对这些组件进行了消融实验。

**网格化**在 GRNet 中被用于将无序的点云映射至有序的 3D Grid。表 3-5展示了

由网格化生成的不同分辨率的 3D Grid。可见，最终补全点云的 F-Score 随着 3D Grid 分辨率的增加而增加。然而，参数量和反向传播时间也随着分辨率的增加而增加。为了在效率和精度上达到平衡，所提出的 GRNet 决定使网格化生成分辨率为 $64^3$ 的 3D Grid。

**立方级特征采样**被用于从 3D Grid 的特征图中抽取点云的上下文信息。为了验证立方级特征采样的有效性，本文对比了使用不同的特征图和不使用特征图的结果。根据表 3-6的实验结果，立方级特征采样显著提升了点云补全结果的质量。此外，随着不同尺度特征图的增多，补全结果的质量会不断改善，并且不会显著提升参数量和反向传播的时间。

**网格化损失**被用于在训练时改善点云的细节。本文进一步验证了网格化损失的有效性，如表 3-7所示。当移除网格化损失后，倒角距离和 F-Score 的都会有明显的下降。将网格化损失中 3D Grid 的分辨率由 $64^3$ 提升至 $128^3$ 时，倒角距离和 F-Score 分别有 25.9% 和 5.4% 的提升。

## 3.4 本章小结

本章提出了针对深度图像的三维物体重建方法，命名为 GRNet。由于现有基于 2D 卷积神经网络的方法和基于多层感知机的方法无法显式地对点云的几何结构和上下文信息进行建模，从而造成几何结构的丢失。为了解决这个问题，本文尝试将卷积操作应用至不规则的点云数据中，并保留其结构和上下文信息。为了实现这个目标，本文将 3D Grid 作为点云的中间表示，从而将无序的点云规则化至一个有序的结构；同时，本文提出了可微分的网格化和逆网格化实现点云和 3D Grid 间的相互转换。本文还提出了可微分的立方级特征采样，为了更好的获取点云的上下文信息。基于网格化，本文设计了网格化损失，以解决现有的损失函数无法恢复物体细节的问题。在 ShapeNet、Completion3D 和 KITTI 数据集的实验结果均表明，GRNet 在重建的精度上均超越了现有的方法。

# 第 4 章 多尺度上下文感知融合多源多视三维物体重建

## 4.1 引言

从多个视角或数据源恢复物体完整的三维结构对机器人、三维建模、物体识别和医疗等领域均有重要价值。运动恢复结构[5]和即时定位和地图重建（Simultaneous Localization and Mapping，SLAM）[177]这些传统方法主要依赖相邻帧之间的特征匹配并借助多视角几何恢复物体的三维结构。尽管这些方法在过去几十年中被广泛使用并取得了令人满意的结果，但它们均只能从彩色或者深度图像中恢复三维结构，难以充分利用多数据源、多视角的图像。然而彩色图像的多视角特征匹配对于弱纹理或重复纹理的物体上会失败，深度图像也无法获取不发生反射物体的几何结构。相比之下，同时使用彩色图像和深度图像的特征可以使得不同数据源的信息相互补充，从而提高重建结果的质量。

近几年来，基于深度学习的多视角三维重建方法不需要特征匹配准即可恢复物体完整的三维结构，并在技术上取得了关键性的突破。代表性的方法包括 3D-R2N2[75]、LSM[76]、DeepMVS[178]、RayNet[179] 和 AttSets[77]。3D-R2N2 和 LSM 借助循环神经网络将多视角三维重建问题转换为序列学习问题：它们使用了循环神经网络融合多个从编码器提取的特征图，随着输入图像的增多，融合后的特征图被逐渐改善。然而基于循环神经网络的方法存在三个问题：

- 由于循环神经网络不满足排列不变性[180]，当改变输入图像的顺序后，循环神经网络无法给出一致的输出。
- 由于循环神经网络的长时记忆丢失的问题[181]，输入网络的早期特征会被其遗忘，从而导致循环神经网络无法充分利用输入图像进行重建。
- 由于循环神经网络当前时刻的输出依赖上一时刻的输入[182]，因此它无法并行计算，从而导致计算效率较低。

为了解决循环神经网络所带来的问题，DeepMVS 在三维重建时使用最大池化（Max Pooling）聚合一组从无序图像中提取的特征；而 RayNet 使用了平均池化（Average Pooling）聚合多个来自于不同体素的特征。虽然最大池化和平均池化一定程度上解决了循环神经网络的上述问题，但是它们只抓住了特征中的平均值和最大值，而丢失了更多可能有用的信息。AttSets 使用了注意力聚合模块（Attentional

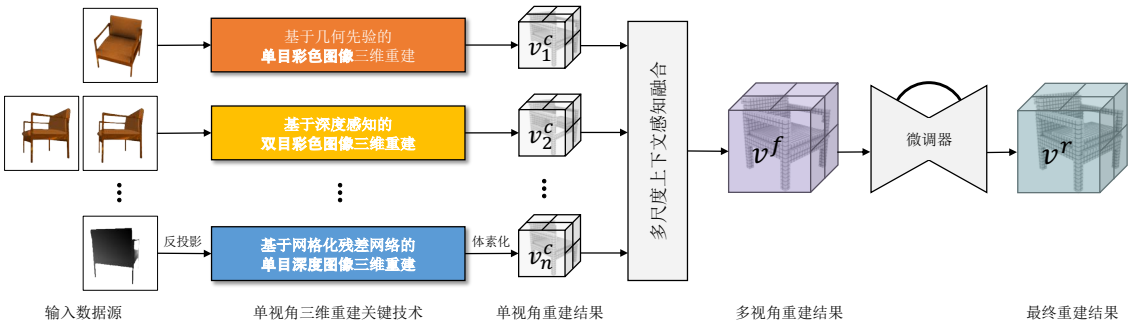



图 4-1 Pix2Vox++ 的总体框架图

Fig.4-1 Overview of Pix2Vox++

Aggregation Module）对每个特征预测一个注意力得分，并通过这个得分聚合输入的特征。然而真实场景的图像中往往会包含较复杂的背景，从而给特征聚合过程带来极大的挑战。

为了解决这些问题，本文提出了 Pix2Vox++，可以从多个视角或数据源拍摄的多模态图像融合对应的重建结果，并生成最终的物体重建结果，框架如图 4-1所示。本文第2章和第3章的工作，使得可以从彩色图像和深度图像分别恢复物体的完整三维结构。而在 Pix2Vox++ 中所提出的多尺度上下文感知融合可以从不同数据源或视角的重建结果中选择重建质量较高的部分，并生成融合后的结果。这样的设计充分利用了所有视角和数据源的信息并且消除了计算效率低下和长时记忆丢失的问题。融合后使用了微调器对融合后的结果进行修正，并生成最终的重建结果。

本章的贡献可以被归纳为以下三点：

• 本文为多数据源和多视角三维物体重建设计了统一的框架，命名为 Pix2Vox++。它可以从任意多个数据源和输入视角恢复物体的完整的三维结构。

• 本文提出了多尺度上下文感知融合，它可以自适应地从多个来自于不同数据源和视角的重建结果中选择高质量的重建部件，通过融合这些部件生成该物体完整的三维结构。

• 在 ShapeNet、Pix3D 和 Things3D 数据集上的重建结果表明，所提出的方法在精度和性能上均超越了现有的方法。

## 4.2 相关工作

传统的三维重建方法（如运动恢复结构、即时定位和地图重建）需要一组彩色图像作为输入，这些方法通过特征匹配并最小化反投影误差恢复物体的三维结构[183]。然而，当多个视角的基线相差非常大时，特征匹配将会变得非常困难；并

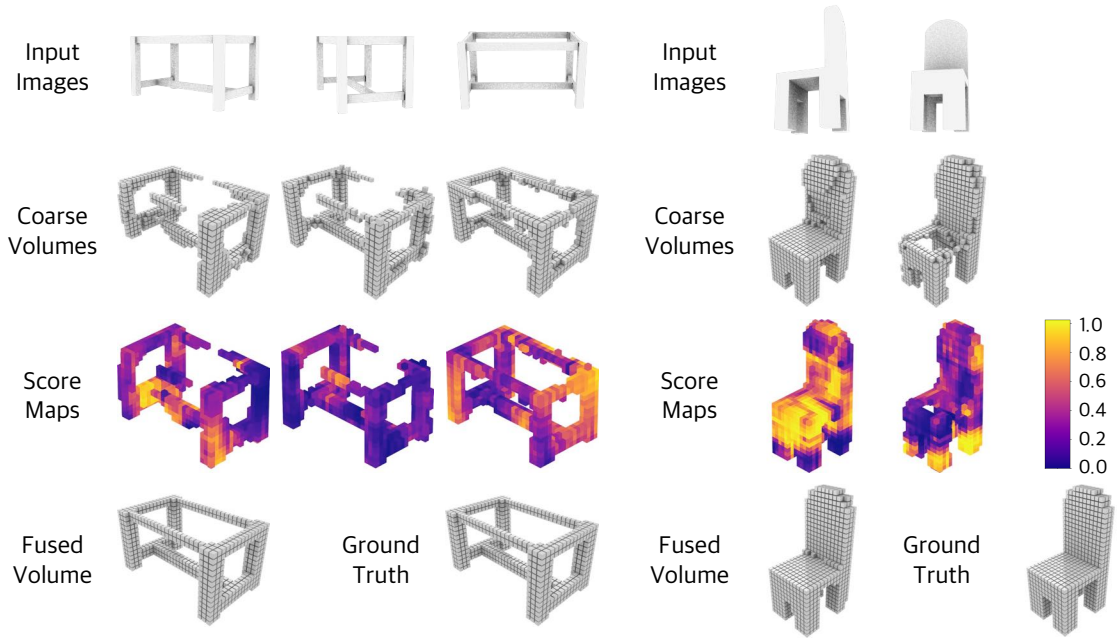



图 4-2 多尺度上下文感知融合中得分图的可视化结果

Fig.4-2 Visualization of the score maps in the multi-scale context-aware fusion module

且在一些情况下，无法在重建前完整地扫描整个物体[33]。近几年来，深度学习网络在图像领域取得了巨大的成功，因此也被用于多视角三维重建解决上述问题。3D-R2N2[75] 和 LSM[76] 均使用了循环神经网络，导致在重建时不满足排列不变性并且不能充分利用输入图像序列中包含的有效信息。DeepMVS[178] 和 RayNet[179] 分别使用最大池化和平均池化聚合特征。AttSets[77] 使用了注意力聚合模块融合来自于多个图像的特征。然而这些方法都没有充分利用输入的特征，而是采用最大池化和平均池化等方式选取了很少部分的特征，从而无法生成高质量的重建结果。除了基于体素的重建方法，近期的多视角三维重建工作也尝试直接重建点云和多边形网格。Pixel2Mesh++[78] 通过图神经网络利用跨视角的信息恢复物体的多边形网格结构。Lin 等人[79] 通过优化物体的多边形网格以实现多视图测光的一致性，同时以先验形状约束多边形网格形变。上述方法均使用了相机外参作为输入，然而当视图之间的基线相差非常大时，获取相机外参将会充满挑战。

## 4.3 多尺度上下文感知融合多源多视三维物体重建模型与方法

为了将多个视角或数据源对应的重建结果融合成一个三维模型，本文提出了多尺度上下文感知融合的三维物体重建方法，命名为 Pix2Vox++。所提出的 Pix2Vox++ 包含两个模块：多尺度上下文感知融合和微调器。首先，多尺度上下

算法 4-1 Pix2Vox++ 算法
Algo.4-1 The Pix2Vox++ algorithm

**Input:** 从不同视角或数据源重建的体素模型集合 $\mathcal{V} = \{v_r^c\}_{r=1}^n$ 和对应上下文集合 $C = \{c_r\}_{r=1}^n$，其中 $v_r^c \in \mathbb{R}^{t^3}$，$f_r \in \mathbb{R}^{9\times t^3}$，t 为体素的分辨率

**Output:** 多视角重建体素模型 $v^r \in \mathbb{R}^{t^3}$

1 给定上下文集合 $C$，上下文评分网络生成得分集合 $\mathcal{M} = \{m_r | m_r \in \mathbb{R}^{t^3}\}_{r=1}^n$；

2 给定得分集合 $\mathcal{M}$，根据公式 4-1归一化评分，生成归一化的得分集合 $\mathcal{S} = \{s_r | s_r \in \mathbb{R}^{t^3}\}_{r=1}^n$；

3 给定体素模型集合 $\mathcal{V}$ 和归一化的得分集合 $\mathcal{S}$，根据公式 4-2生成加权融合后的三维体素模型 $v^f \in \mathbb{R}^{t^3}$；

4 给定融合后的三维体素模型 $v^f \in \mathbb{R}^{t^3}$，微调器生成最终的体素模型 $v^r$。

文感知融合对每个三维体素部件的重建质量进行评分，并生成对应得分 $s_r (r = 1, 2, \ldots, n)$，并根据部件的评分从不同粗糙三维体素重建 $v_r^c (r = 1, 2, \ldots, n)$ 中选择重建质量较高的部件，融合成三维体素 $v_f$。微调器通过残差连接形成一个残差网络，从而用于进一步优化重建结果 $v_f$，生成 $v_r$。Pix2Vox++ 的算法描述如算法 4-1所示。

**（1）多尺度上下文感知融合**

对于同一个物体，不同视角所观察到的部位是不同的。对于不可见的部位，它的重建质量会略低于可见的部位。受到这点的启发，本文提出了多尺度上下文感知融合，它可以从多个三维体素中选择高质量的重建部位，这些部位被用于组建成一个融合的三维模型。如图4-2所示，对于重建质量较高的部件，多尺度上下文感知融合将会产生更高的得分，这可以显著降低某个视角中错误重建对于最终结果的影响。图中的虚线框展现了对于体素 $v_1^c$ 得分的计算过程，其他体素的得分也遵循相同的方式计算。如图4-3所示，给定一个或者多个三维体素模型，多尺度上下文感知融合模块可以为体素中的每一个点输出一个得分，这些得分会被用于加权，以融合多个三维体素的值。在这个过程中，不同三维体素的空间约束得以保留，使得 Pix2Vox++ 可以更好地利用多个视图的信息恢复物体的三维结构。另一方面，更深的卷积卷积层拥有更大的感受野，可以帮助探索体素中更多的上下文信息。然而更深的卷积层丢失了物体的细节，所提出的多尺度上下文感知引入了浅层网络的特征图，将不同尺度的特征图叠加起来；这样既扩大了感受野，也保留了物体细节。

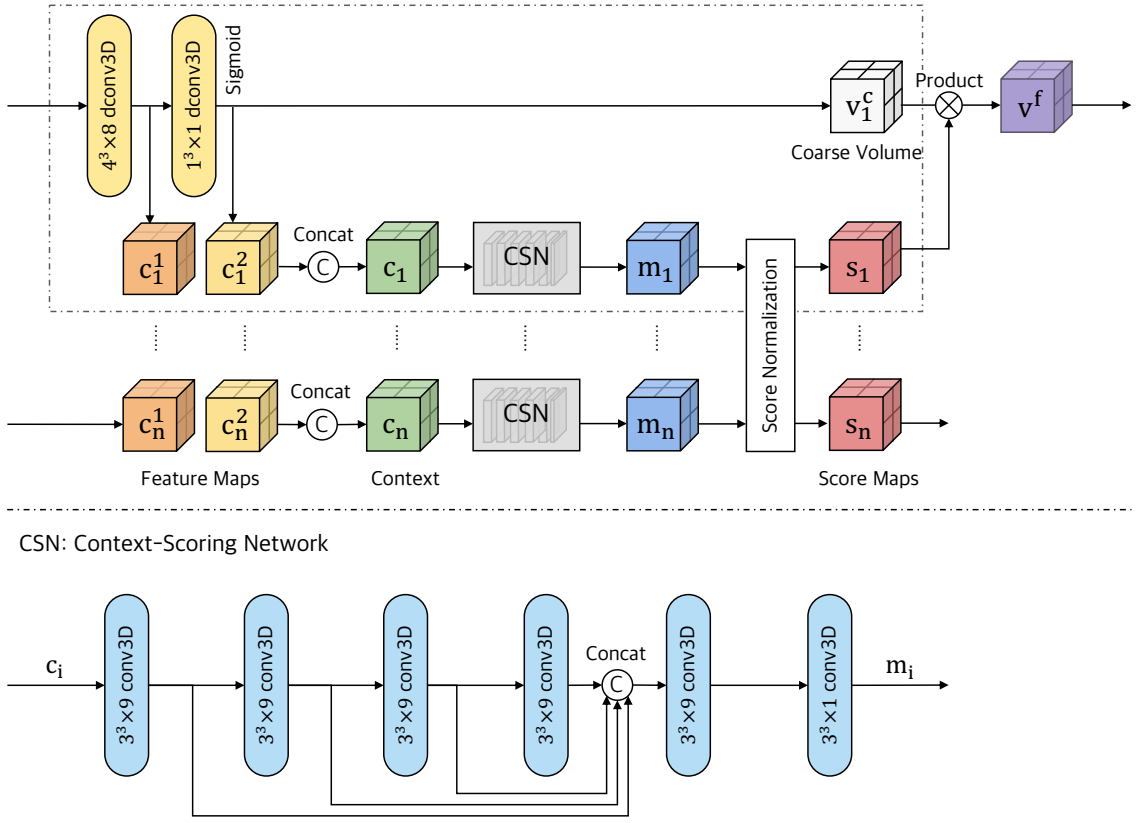



图 4-3 多尺度上下文感知融合的总体框架图和网络结构图

Fig.4-3 Overview and the network architecture of the multi-scale context-aware fusion module

具体来说，多尺度上下文感知融合模块将第 $r$ 个视图生成的三维模型 $v_r^c$ 对应的上下文 $c_r$ 作为输入，并通过上下文评分网络（Context Scoring Network）为第 $r$ 个模型的上下文生成得分 $m_r$。其中上下文由解码器最后两层的输出拼接（Concatenation）而成。上下文评分网络由 5 个 3D 卷积块组成，每个卷积块中包含 Batch Normalization 和 Leaky ReLU，其中的每个卷积层的 Kernel Size 为 $3^3$，Padding 为 1。前 4 个卷积层的输出通道数为 9，并且它们的输出在拼接后输入至最后一个输出通道数为 1 的卷积层。之后，多尺度上下文感知融合模块对每个 $c_r$ 的输出 $m_r$ 使用 Softmax 函数进行归一化。第 $r$ 个体素中坐标为 $(i, j, k)$ 的得分 $s_r^{(i,j,k)}$ 可以通过如下公式计算得到：

$$s_r^{(i,j,k)} = \frac{\exp\left(m_r^{(i,j,k)}\right)}{\sum_{p=1}^{n} \exp\left(m_p^{(i,j,k)}\right)} \tag{4-1}$$

其中 $n$ 表示输入图片的数量。最终的加权输出结果 $v^f$ 可以由每个点的得分和对应位置的体素值加权得到：

$$v^f = \sum_{r=1}^{n} s_r v_r^c \tag{4-2}$$

**（2）微调器**

微调器可以被视作一个残差网络，被用于进一步修正多尺度上下文感知融合所生成的不正确的融合结果。它的设计基本遵循一个 3D 编码器-解码器的 U-Net 结构[125]。受益于编码器和解码器之间的 U-Net 连接，在融合三维模型中的局部结构可以很好地保留。

对于生成分辨率为 $32^3$ 体素的微调器，该微调器中的编码器包含 3 个 3D 卷积块，每个卷积块中包含 Batch Normalization、ReLU 和 Kernel Size 为 $2^3$ 的 Max Pooling 层。其中的每个 3D 卷积层的 Kernel Size 为 $4^3$，Padding 为 2。这三个卷积层的输出通道数分别为 32，64 和 128。在编码器之后，有 2 个全连接层，维度分别为 2048 和 8192。解码器由 3 个 3D 转置卷积层组成，其中的每个卷积层 Kernel Size 为 $4^3$，Padding 为 2，Stride 为 1。这三个转置卷积层的输出通道数分别为 64，32 和 1。除了最后一个转置卷积层使用了 Sigmoid，其余的转置卷积层中包含 Batch Normalization 和 ReLU。

对于生成分辨率为 $64^3$ 体素的微调器，该微调器中的编码器包含 4 个 3D 卷积块，每个卷积块中包含 Batch Normalization、ReLU 和 Kernel Size 为 $2^3$ 的 Max Pooling 层。其中的每个 3D 卷积层的 Kernel Size 为 $4^3$，Padding 为 2。这四个卷积层的输出通道数分别为 16, 32，64 和 128。解码器由 4 个 3D 转置卷积层组成，其中的每个卷积层 Kernel Size 为 $4^3$，Padding 为 2，Stride 为 1。这四个转置卷积层的输出通道数分别为 64，32, 16 和 1。除了最后一个转置卷积层使用了 Sigmoid，其余的转置卷积层中包含 Batch Normalization 和 ReLU。

对于生成分辨率为 $128^3$ 体素的微调器，该微调器中的编码器包含 5 个 3D 卷积块，每个卷积块中包含 Batch Normalization、ReLU 和 Kernel Size 为 $2^3$ 的 Max Pooling 层。其中的每个 3D 卷积层的 Kernel Size 为 $4^3$，Padding 为 2。这五个卷积层的输出通道数分别为 8, 16, 32，64 和 128。解码器由 5 个 3D 转置卷积层组成，其中的每个卷积层 Kernel Size 为 $4^3$，Padding 为 2，Stride 为 1。这五个转置卷积层的输出通道数分别为 64，32, 16, 8 和 1。除了最后一个转置卷积层使用了 Sigmoid，其余的转置卷积层中包含 Batch Normalization 和 ReLU。

**（3）损失函数**

训练网络使用了二值交叉熵损失（Binary Cross Entropy Loss），它定义为：

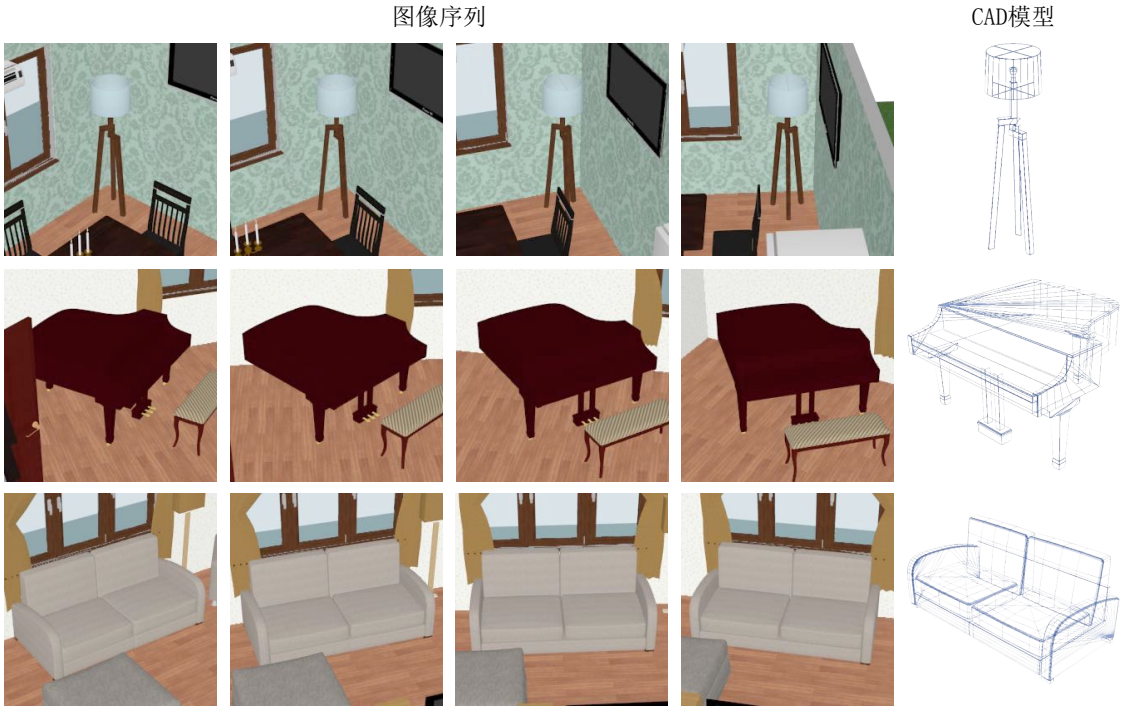



图 4-4 Things3D 数据集中图像和对应 CAD 模型的示例

Fig.4-4 Examples of the images and the corresponding CAD models in Things3D

$$\ell = \frac{1}{N}\sum_{i=1}^{N}[gt_i \log(p_i) + (1 - gt_i)\log(1 - p_i)] \tag{4-3}$$

其中，$N$ 表示模型中体素点的数量。$p_i$ 和 $gt_i$ 分别表示预测出的三维结构和 Ground Truth。$\ell$ 的值越小，表明预测结果越接近 Ground Truth。

## 4.4 实验结果与分析

**（1）数据集**

ShapeNet[30] 是一个根据 WordNet①类别组织的三维 CAD 模型的集合。本节的实验参考了 3D-R2N2[75] 的实验设置，使用了 ShapeNet 的一个子集，这个子集包含了来自 13 个类别的 44,000 个模型。每个模型均包含 24 张分辨率为 $137 \times 137$ 的彩色图像。

Pix3D[117] 提供了真实场景的图像和对应的三维模型，这些三维模型与真实图像精准地对应。这个数据集包含了来自 9 个类别的 395 个三维模型。每一个三维模型都对应了在多张在不同真实场景拍摄的图像。本节的实验参考了 Pix3D 的实验设置，使用了 2,894 未裁剪且未遮挡的真实场景中椅子的图像测试所提出的方法。

① https://wordnet.princeton.edu/

图 4-5 Things3D 数据集中场景的示例
Fig.4-5 The examples of the scene in the Things3D dataset

Things3D 数据集包含 28 万个物体的 168 万张图像，数据集中的部分样本图 4-4所示。这个数据集是本文基于 SUNCG[112] 数据集提出的，以解决目前没有大规模自然场景三维重建数据集的问题。SUNCG 数据集包含 3.9 万个室内场景，但这些场景中仅包含 2,000 种三维模型。为了增加数据集中三维模型的多样性，SUNCG 场景中的原有模型被 ShapeNet 数据集中的同类别模型替换，在替换后确保模型的长宽高均小于原有模型，替换后的场景如图 4-5所示。对于场景中的每一个物体，相机以以 30 度的俯仰角围绕物体旋转，生成 24 张图像。相机到物体的距离被设置为 10 个单位长度，相机的焦距为 96 毫米。当渲染得到的图像中的物体有超过 12.5% 的像素被遮挡时，这张图像将被舍弃。图像的大小被设定为 $256 \times 256$。整个渲染的过程使用了 15 台服务器运行了 32 天，每台服务器包含 4 个 Intel Xeon E5-2682 v4@2.50GHz 的 CPU 和 256GB 的内存。

**（2）度量指标**

Intersection over Union (IoU) 被用于评估输出结果的质量。IoU 的阈值被设置为 0.3，将输出的预测概率进行二值化并计算 IoU。其定义如公式2-2所示。越大的 IoU 表明越好的体素重建结果。

**（3）实现细节**

本文使用 PyTorch[128] 实现了所提出的方法①，并使用了一块 NVIDIA GTX 1080

① 代码已开源：https://gitlab.com/hzxie/Pix2Vox

表 4-1 在 ShapeNet 数据集上使用多视角图像生成分辨率为 $32^3$ 体素重建的 IoU
Table 4-1 The IoU of multi-view 3D volume reconstruction on ShapeNet at $32^3$ resolution

| 视角数量 | 1 | 2 | 3 | 4 | 5 | 8 | 12 | 16 | 20 |
|---|---|---|---|---|---|---|---|---|---|
| 3D-R2N2[75] | 0.560 | 0.603 | 0.617 | 0.625 | 0.634 | 0.635 | 0.636 | 0.636 | 0.636 |
| AttSets[77] | 0.642 | 0.662 | 0.670 | 0.675 | 0.677 | 0.685 | 0.688 | 0.692 | 0.693 |
| Pix2Vox++ | **0.670** | **0.695** | **0.704** | **0.708** | **0.711** | **0.715** | **0.717** | **0.718** | **0.719** |

表 4-2 在 Things3D 数据集上使用多视角图像生成分辨率为 $32^3$ 体素重建的 IoU
Table 4-2 The IoU of multi-view 3D volume reconstruction on Things3D at $32^3$ resolution

| 视角数量 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| 3D-R2N2[75] | 0.307 | 0.316 | 0.322 | 0.325 | 0.329 | 0.331 | 0.332 | 0.334 |
| AttSets[77] | 0.402 | 0.415 | 0.422 | 0.427 | 0.429 | 0.431 | 0.433 | 0.434 |
| Pix2Vox++ | **0.428** | **0.444** | **0.452** | **0.456** | **0.460** | **0.462** | **0.465** | **0.467** |

Ti GPU 训练所提出的神经网络。网络输入图像的分辨率为 $224 \times 224$，输出的体素分辨率为 $32^3$。在训练时，Batch Size 被设置为 64，并使用了 $\beta_1 = 0.9$ 和 $\beta_2 = 0.999$ 的 Adam[129] 优化器。初始的学习率被设置为 $10^{-3}$，并在 150 个 Epoch 之后下降为原来的一半，训练一共持续 250 个 Epoch。

**（4）与现有方法的比较**

为了验证 Pix2Vox++ 在多视图三维重建上的性能，本文将其和 3D-R2N2[75] 以及 AttSets[77] 在 ShapeNet 上进行了对比。表 4-1的结果表明，Pix2Vox++ 在所有的输入视图数量上超越了这两个对比方法。图 4-6展示了使用 3 个视角的彩色图像进行重建的定性结果，该结果也表明，所提出的 Pix2Vox++ 可以更好地恢复物体的细节。图 4-2展示了多视图重建时得分图的可视化结果，对于最右侧的椅子而言，坐凳的重建质量相对较差，因此对应位置将会有较低的得分。这种评分机制可以有效消除低质量重建对于最终融合结果的影响。表 4-3对比了参数量、显存占用和运行时间。其中，运行时间是在同一台配备 NVIDIA GTX 1080 Ti GPU 的机器上测试得到的。实验结果表明，Pix2Vox++ 在多视图重建上的推理速度均快于 3D-R2N2 和 AttSets。

为了进一步评估所提出的方法对于自然场景图像上的性能，Pix2Vox++ 在 Things3D 上和 3D-R2N2 以及 AttSets 进行了对比。表 4-2的结果表明，对于自然场景的数据，Pix2Vox++ 在所有的输入视图数量上超越了 3D-R2N2 和 AttSets。图 4-7也表明，Pix2Vox++ 相比于 3D-R2N2 和 AttSets 可以更好地重建自然场景中的物体。另外，实验结果表明，Things3D 数据集可以提高模型在真实场景的泛化

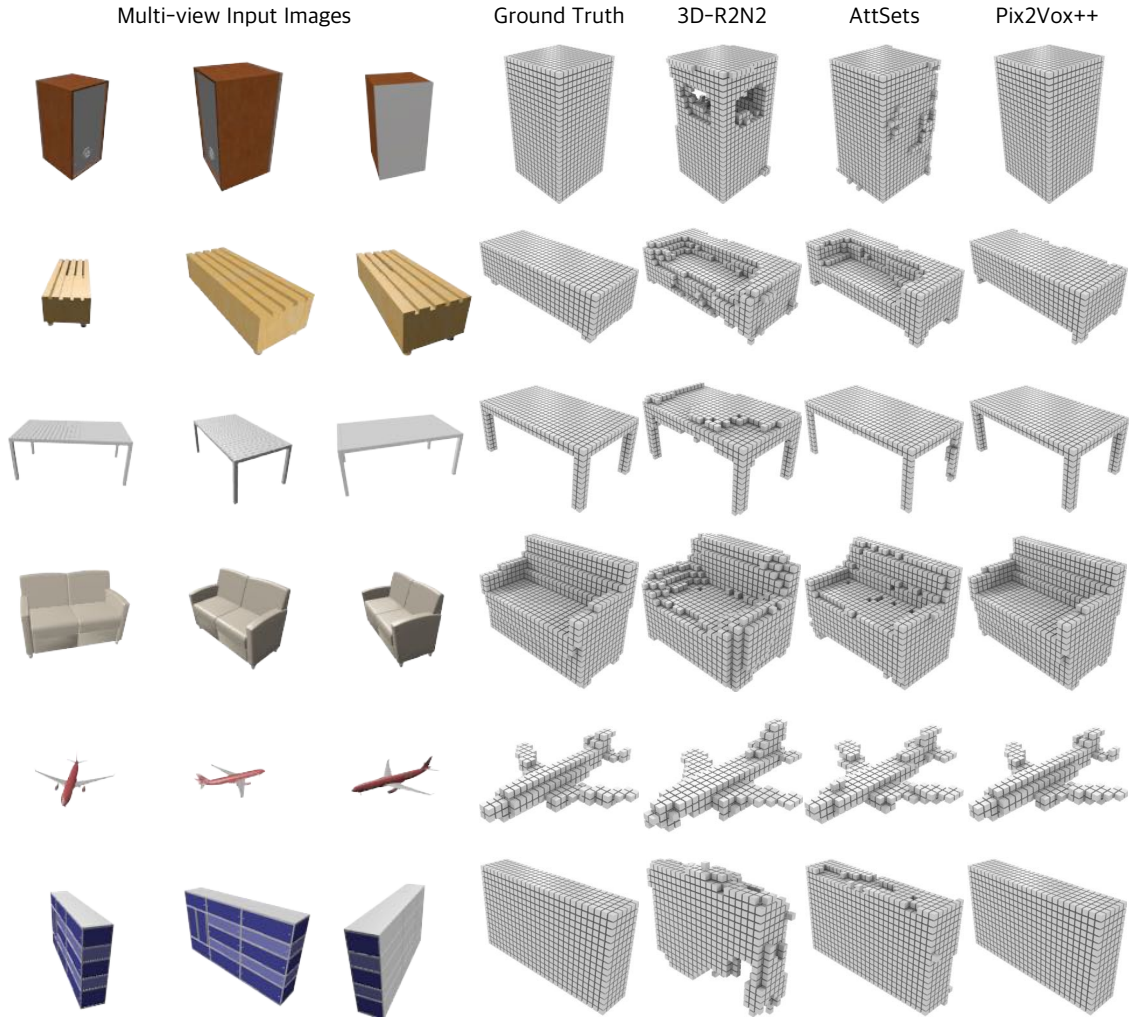



图 4-6 在 ShapeNet 数据集上使用多视角图像生成分辨率为 $32^3$ 体素的重建结果
Fig.4-6 Multi-view 3D volume reconstruction on ShapeNet at $32^3$ resolution

表 4-3 在 ShapeNet 数据集上参数量、内存占用和推理时间的对比
Table 4-3 The comparison of #parameters, memory footprint, and inference time on ShapeNet

| 方法 | 3D-R2N2[75] | AttSets[77] | Pix2Vox++ |
|---|---|---|---|
| 参数量 (M) | 35.97 | 17.71 | 96.31 |
| 显存占用 (MB) | 1407 | 3911 | 2411 |
| 推理时间 (毫秒) | | | |
| 1 视图 | 78.86 | 26.32 | 10.64 |
| 2 视图 | 112.27 | 47.62 | 17.51 |
| 4 视图 | 116.68 | 52.63 | 29.88 |
| 8 视图 | 122.04 | 58.83 | 56.52 |

能力。表 4-4对比了所提出的模型使用不同的训练集在 Pix3D 上的测试结果，其

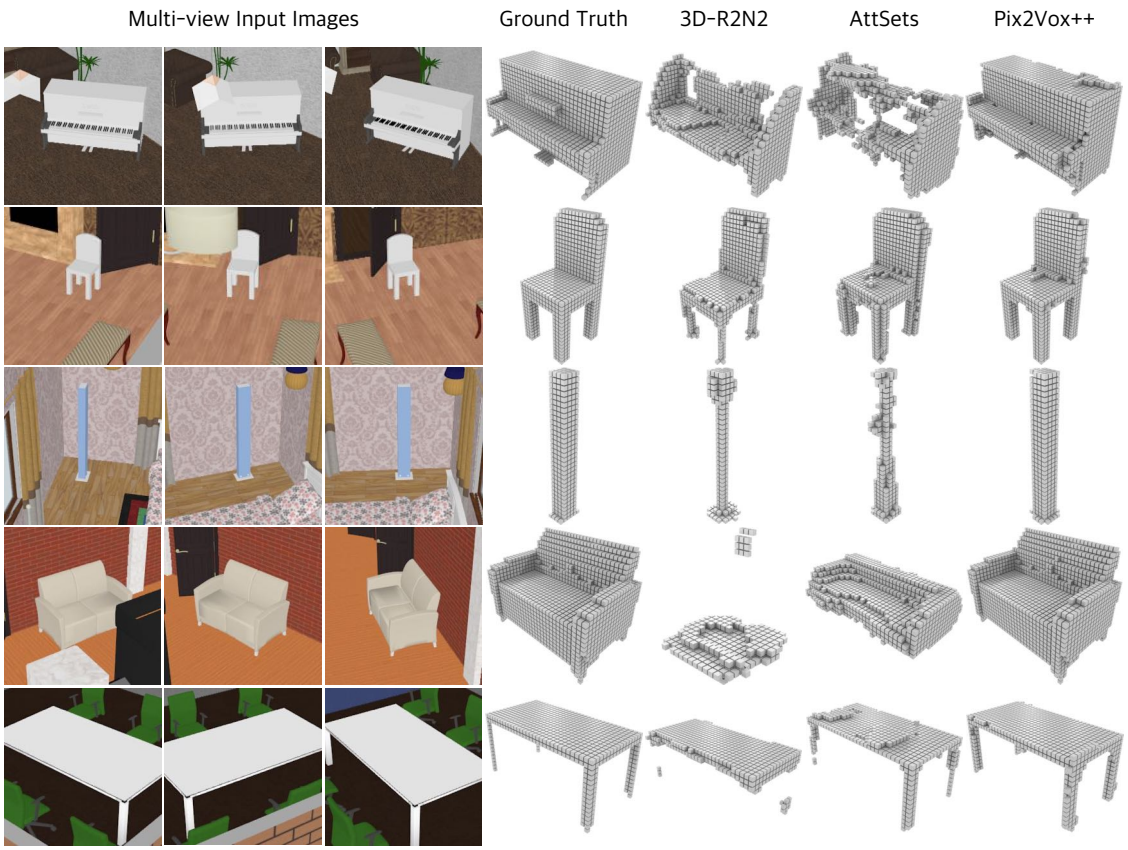



图 4-7 在 Things3D 数据集上使用多视角图像生成分辨率为 $32^3$ 体素的重建结果
Fig.4-7 Multi-view 3D volume reconstruction on Things3D at $32^3$ resolution

表 4-4 在 Pix3D 数据集上使用不同训练数据的三维体素重建的 IoU 的对比
Table 4-4 The IoU of 3D volume reconstruction results on Pix3D with different training data

| 方法 | IoU |
|---|---|
| 3D-R2N2[75] | 0.136 |
| DRC[127] | 0.265 |
| Pix3D (w/o Pose)[117] | 0.267 |
| Pix3D (w/ Pose)[117] | 0.282 |
| Pix2Vox++ (训练集：*ShapeNet-Chairs*) | 0.204 |
| Pix2Vox++ (训练集：*Things3D-Chairs*) | 0.269 |
| Pix2Vox++ (训练集：*ShapeNet-Chairs-Rfc*) | 0.292 |
| Pix2Vox++ (训练集：*Things3D-Chairs-Rfc*) | **0.324** |

中 ShapeNet-Chairs 和 Things3D-Chairs 分别代表 ShapeNet 和 Things3D 中的椅子类别的数据，ShapeNet-Chairs-Rfc 和 Things3D-Chairs-Rfc 代表根据 Pix3D 的数据分布在 ShapeNet 和 Things3D 中的椅子类别的数据上使用 Render for CNN[130] 重新渲染得到的数据。这些结果表明，相比于 ShapeNet，Things3D 可以更好地帮助

表 4-5 在 ShapeNet-Cars 数据集上使用多视角图像生成分辨率为 $64^3$ 和 $128^3$ 体素重建的 IoU
Table 4-5 The IoU of multi-view 3D object reconstruction on ShapeNet-Cars at $64^3$ and $128^3$ resolutions

| 视角数量 | 1 | 2 | 4 | 8 |
|---|---|---|---|---|
| **分辨率:** $64^3$ | | | | |
| OGN[29] | 0.771 | N/A | N/A | N/A |
| Matryoshka[135] | 0.784 | N/A | N/A | N/A |
| Pix2Vox++ | **0.803** | **0.813** | **0.815** | **0.819** |
| **分辨率:** $128^3$ | | | | |
| OGN[29] | 0.782 | N/A | N/A | N/A |
| Matryoshka[135] | 0.794 | N/A | N/A | N/A |
| Pix2Vox++ | **0.826** | **0.837** | **0.841** | **0.843** |

所提出的模型泛化至真实场景的数据集；另外在 Render for CNN 的帮助下，在 Things3D-Chairs-Rfc 上训练的模型可以在 Pix3 D 数据集上取得最好的性能。

之前的实验主要对比了生成分辨率为 $32^3$ 体素的三维重建结果，而在这种低分辨率下的体素丢失了物体的细节。为了进一步验证 Pix2Vox++ 对于高分辨率体素的三维重建结果，本文将其与 Matryoshka Networks[135] 和 OGN[29] 进行了对比。本实验遵循了 OGN 中的实验设置，从 ShapeNet-Cars（即 ShapeNet 中的车辆类别）数据集中分别从彩色图像中预测分辨率为 $64^3$ 和 $128^3$ 的体素。在计算 IoU 时，重建结果被上采样至 $256^3$，并与该分辨率的 Ground Truth 计算 IoU。为了确保对比的公平性，实验使用了 OGN 的数据集划分并使用了该方法所提供的 Ground Truth。表 4-5展示了 Pix2Vox++ 与其他方法的定量结果。该结果表明，相比于 Matryoshka Networks 和 OGN，Pix2Vox++ 在从单视角生成分辨率为 $64^3$ 和 $128^3$ 的体素可以取得更好的结果。此外，表 4-5也给出了 Pix2Vox++ 在 $64^3$ 和 $128^3$ 分辨率下多视角三维物体重建结果。

**（5）消融实验**

微调器被用于修正多尺度上下文感知融合所生成的不正确的融合结果。当从 Pix2Vox++ 中移除微调器后，单视角三维物体重建的 IoU 由 0.670 降低至 0.658。图 4-8展示了在不同输入视角数量下使用微调器和不使用微调器的性能对比。从图中可知，在其他视角的数量下，移除微调器也会造成明显的性能下降。

多尺度上下文感知融合在融合多个视角/多数据源重建结果发挥重要的作用。为了验证其有效性，本文将它与其他融合方式做了对比，包括均值融合，基于循环神经网络的融合[75]，注意力聚合[77] 以及不使用多尺度的上下文感知融合。均值

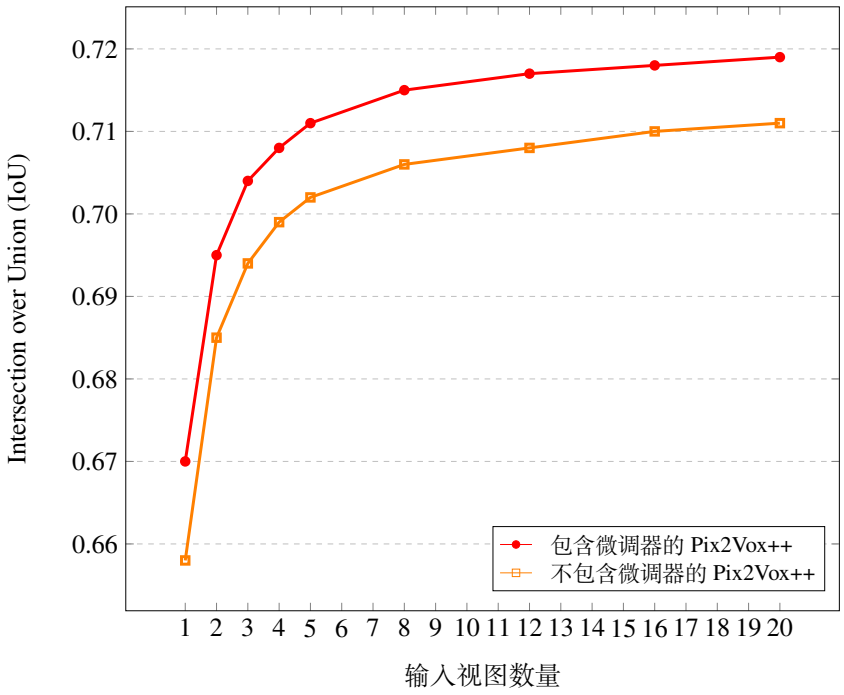



图 4-8 在 ShapeNet 数据集上不同输入视图数量性能的对比

Fig.4-8 The comparison of the performance in different numbers of views on ShapeNet

表 4-6 在 ShapeNet 数据集上使用多视角图像生成分辨率为 $32^3$ 体素的重建 IoU

Table 4-6 The IoU of multi-view 3D volume reconstruction on ShapeNet at $32^3$ resolution

| 视角数量 | 1 | 2 | 3 | 4 | 5 | 8 | 12 | 16 | 20 |
|---|---|---|---|---|---|---|---|---|---|
| Pix2Vox++ † | **0.670** | 0.680 | 0.690 | 0.695 | 0.699 | 0.703 | 0.704 | 0.705 | 0.706 |
| Pix2Vox++ ‡ | **0.670** | 0.690 | 0.699 | 0.702 | 0.706 | 0.710 | 0.712 | 0.713 | 0.714 |
| Pix2Vox++-R2N2 | 0.663 | 0.672 | 0.680 | 0.684 | 0.686 | 0.688 | 0.689 | 0.689 | 0.690 |
| Pix2Vox++-AttSets | 0.638 | 0.675 | 0.689 | 0.696 | 0.701 | 0.707 | 0.710 | 0.713 | 0.713 |
| Pix2Vox++ | **0.670** | **0.695** | **0.704** | **0.708** | **0.711** | **0.715** | **0.717** | **0.718** | **0.719** |

融合是指在最终的重建体素 $v^f$ 中，坐标为 $(i,j,k)$ 的体素值由各个视图产生的体素在坐标为 $(i,j,k)$ 体素值求均值获得。这个过程可以形式化地描述为：

$$v^f_{(i,j,k)} = \frac{1}{n}\sum_{r=1}^{n} v^r_{(i,j,k)} \tag{4-4}$$

如表 4-6所示，当多尺度上下文感知融合被均值融合替代之后，Pix2Vox++ 的重建性能有所下降。这主要是由于均值融合容易受到重建结果极值的影响。为了和基于循环神经网络的融合进行对比，Pix2Vox++ 中的多尺度上下文感知融合被替换为 3D-R2N2 中的循环神经网络融合来自不同视图的重建结果。为了满足循环神经网络的输入维度，网络中增加了一个大小为 1,024 的全连接层，并将这个方法命名

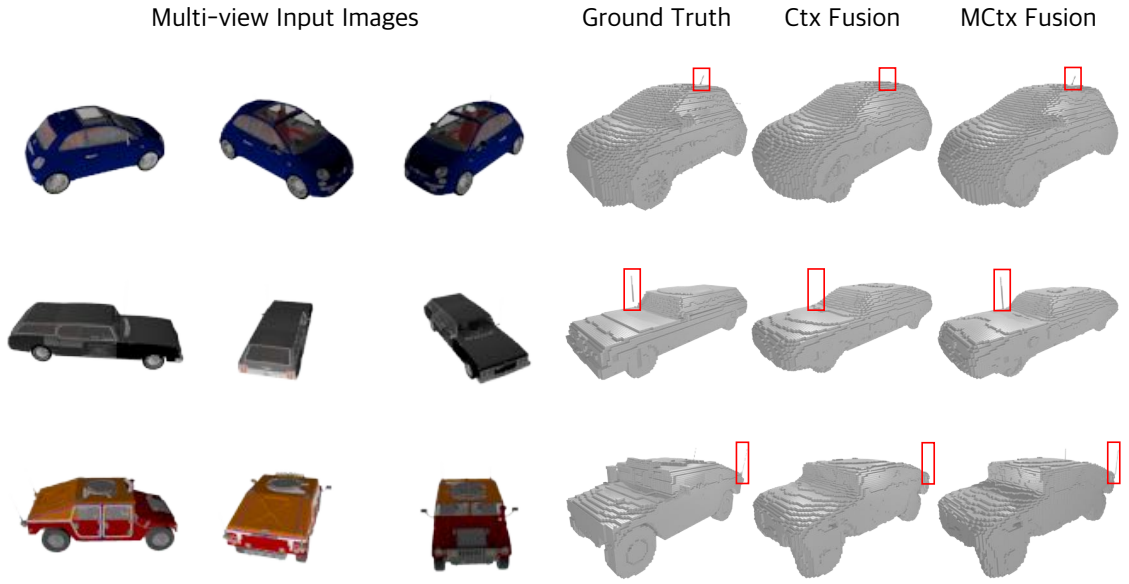



图 4-9 在 ShapeNet-Cars 数据集上使用多视角图像生成分辨率为 $128^3$ 体素的重建结果
Fig.4-9 The multi-view 3D volume reconstruction on ShapeNet-Cars at $128^3$ resolution

为 Pix2Vox++-R2N2。如表 4-6所示，Pix2Vox++ 在所有的输入视图数量上超越了 Pix2Vox++-R2N2。为了和注意力聚合方法进行对比，Pix2Vox++ 中的多尺度上下文感知融合被替换为 AttSets 中所提出的注意力聚合融合来自多个视图的重建结果，并将这个方法命名为 Pix2Vox++-AttSets。如表 4-6所示，Pix2Vox++ 在所有的输入视图数量上超越了 Pix2Vox++-AttSets。为了验证在上下文感知融合中多尺度特征的重要性，Pix2Vox++ 中的多尺度上下文感知融合被替换为不使用多尺度特征的上下文感知融合方法，并将这个方法命名为 Pix2Vox++‡。如表 4-6所示，Pix2Vox++ 相比于 Pix2Vox++‡ 在 IoU 上有 1% 的提升。图 4-9对比了多尺度上下文感知融合和上下文感知融合在重建高分辨率体素的结果，其中 “Ctx Fusion’’ 和 “MCtx Fusion’’ 分别表示使用了上下文感知融合和多尺度的上下感知融合的重建结果。相比于上下文感知融合，多尺度上下文感知融合在重建时可以更好地恢复物体的细节，因为它同时考虑了不同尺度的特征，而底层特征可以很好地保留物体的细节。

## 4.5 本章小结

本章提出了多源多视三维物体重建方法，命名为 Pix2Vox++。不同视角可以观察到物体不同的部件，从而可见部位的重建质量高于不可见部位的重建质量。受到这一点的启发，本文提出了多尺度上下文感知融合，用于融合来自于多个数据源、多个视角的重建结果。所提出的方法使得彩色图像和深度图像的信息可以相

互补充，可以更好地重建弱纹理、重复纹理和不发生反射的物体。相比于现有的基于循环神经网络的多视角三维重建方法，所提出的方法解决了循环神经网络不满足排列不变性和长时记忆丢失的问题；相比于基于池化的多视角三维重建方法，所提出的方法可以更加充分地利用来自于图像的信息。此外，本文构建了目前最大规模包含自然场景的多视角三维重建数据集——Things3D，该数据集可以显著提升现有方法在真实场景数据集上的泛化能力。在 ShapeNet、Pix3D 和 Things3D 数据集的实验结果均表明，Pix2Vox++ 在重建速度和精度上均超越了现有的方法。

# 第 5 章 基于场景语义感知的多源多视三维场景重建

## 5.1 引言

从单张图像中理解并重建三维场景一直是计算机视觉领域长期研究的问题。在过去的几十年中，无语义的三维场景重建方法（比如运动恢复结构[5] 和即时定位和地图重建[177]）取得了令人满意的结果，然而这些方法在重建场景时无法感知环境的语义信息。因此重建后的物体和背景融为一体，难以将物体从重建场景中分离。另外，这些方法仅能重建物体可见部分的三维结构；当重建物体的完整三维结构时，需要扫描物体的完整图像，然而这在一些情况下并不是可行的[33]。尽管在最近几年，基于三维物体重建的三维场景理解算法被提出[100,102]，但基本上所有的三维场景理解算法仅能以单个视角的图像作为输入。而从单视角图像中恢复物体的三维结构时一个非适定性问题（Ill-posed Problem）：一张图像理论上可以对应无数个三维结构[41]。

为了解决这些问题，本文提出了基于场景语义建模的多源多视三维场景重建方法，命名为 URNet。该方法使用视频物体分割算法估计物体的掩膜，将物体从图像序列中分离，再使用多尺度上下文感知融合的多源多视三维物体重建方法重建该物体完整的三维结构。重建后，Marching Cubes[184] 算法被用于从体素模型生成多边形网格模型。在重建完成后，URNet 估计了物体的位姿，并将多个重建的物体进行组合，从而完成对三维场景的重建，如图 5-1所示。所提出的方法在重建时完成了对场景的理解，从而使得重建后的物体可以容易地从重建的场景中分离。

现有的视频物体分割使用了掩膜传播（Mask Propagation）或者特征匹配的策略估计物体在视频序列中的掩膜。早期基于掩膜传播的方法[185–187] 使用光流将物体的掩膜从上一帧传递至下一帧，并使用一个全卷积网络（Fully Convolutional Network）完善估计的物体掩膜。然而基于掩膜传播的方法很容易造成误差累积，特别是当目标物体被遮挡或是漂移时。近几年，基于特征匹配的方法[188–193] 使用全局-全局的特征匹配计算过去帧和当前帧特征相似度和匹配关系。在这些方法中，基于 STM（Space-Time Memory）的方法[189–191] 将过去帧记忆在网络中，从而更好地应对物体的遮挡和漂移。然而，这些方法也记忆了目标物体之外区域的特征，从而导致了对相似物体的错配，也造成了更高的计算复杂度。为了解决这个问题，

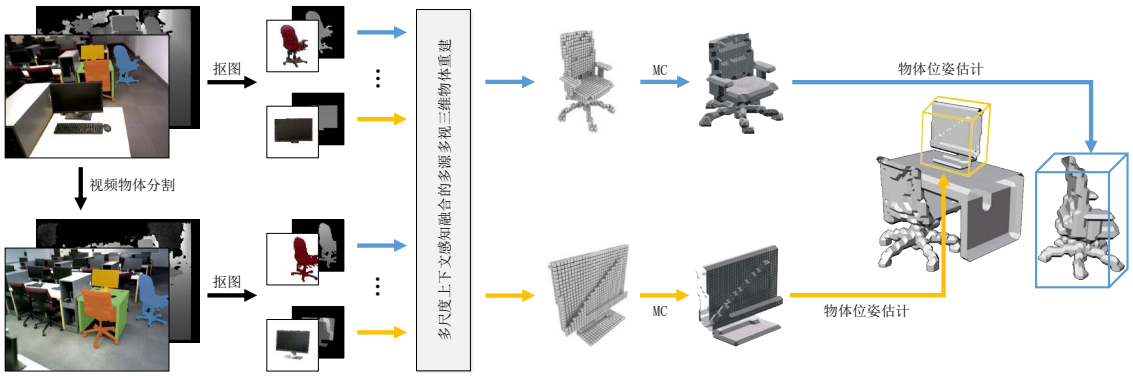



图 5-1 URNet 的总体框架图
Fig.5-1 Overview of URNet

本文提出了 RMNet，仅提取目标物体所在区域的特征，并在该区域内进行特征匹配，从而显著减少了对相似物体的错配，并提高了计算效率。

本章的贡献可以被归纳为以下三点：

• 本文提出了视频物体分割方法，命名为 RMNet。该方法仅在包含目标物体的区域内进行记忆和匹配，有效消除了相似物体的错配并降低了计算复杂度。

• 本文提出了基于场景语义建模的多源多视三维场景重建方法，命名为 URNet。该方法估计出目标物体的掩膜，恢复目标物体完整的三维结构，并估计该物体的位姿，进而恢复场景的三维结构。

• 在 DAVIS 和 YouTube-VOS 数据集上，所提出的 RMNet 取得了更好的精确度和更快的推理速度；在 SUN3D 数据集和实地拍摄的视频上，所提出的 URNet 可以生成更高质量的三维场景重建结果，证明了所提出方法的有效性。

## 5.2 相关工作

### 5.2.1 视频物体分割

**（1）基于传播的方法**

早期的视频物体分割方法将这个任务视为时序上的标签传播问题。ObjectFlow[194]、SegFlow[195] 和 DVSNet[196] 同时估计视频中物体的掩膜和光流。为了更好地泛化至具体的物体实例，MaskTrack[187] 针对第一帧微调了整个网络。MaskRNN[185] 首先估计出物体的边界框和光流，再根据物体的边界框和光流预测多个物体的掩膜。CINN[197] 使用了马尔科夫随机场建立了视频帧之间像素的时空关系。DyeNet[198] 和 PReMVOS[186] 在考虑时间上标签的传播同时加入了重识别（Re-identification）。FAVOS[199] 为每个物体的部位生成一个边界框，并使用目标跟

踪的策略在整个视频中跟踪这些边界框，并以此进一步为每一帧推断物体的掩膜。LucidTracker[200] 通过合成和测试数据集同领域的数据去训练特定的视频语义分割网络，该方法可以使用比原有方法 1/1000 至 1/20 的训练数据。SAT[201] 利用视频帧内在联系产生更为适配的跟踪搜索区域和生成更稳定的目标分割结果。尽管这些方法取得了较为满意的结果，但是这些方法对于遮挡和漂移并不鲁棒，从而造成传播过程中的误差累积[202]。

**（2）基于特征匹配的方法**

为了更好地应对遮挡和漂移，近期的方法使用特征匹配的方法在视频中寻找相似外观的物体。OSVOS[203] 通过微调网络的方法将通用的语义信息迁移至视频物体分割任务中。RGMP[204] 使用上一帧估计的掩膜作为当前帧的先验知识，和当前帧一起用于估计当前帧的分割结果。PML[188] 将分割视作像素级标签检索问题，首先使用三元组损失（Triplet Loss）通过最近邻匹配学习了像素级别的特征表示，再根据最近邻算法决定分割的结果。VideoMatch[205] 使用了软匹配（Soft Matching）机制估算不同帧之间的相似性得分。像 PML 和 VideoMatch 一样，FEELVOS[192] 使用学习的特征和最近邻匹配，但是将此机制用作卷积网络的内部指导，而不是将其用于最终的分割决策。这使能够使用标准交叉熵损失以端到端的方式学习特征。基于 FEELVOS，CFBI[193] 同时显式地考虑了前景物体和背景物体的匹配。STM[190] 利用了一个记忆网络将过去的多帧和当前帧进行特征匹配，该网络的性能超越了过去所有的方法。在 STM 的基础上，KMN[191] 引入了一个高斯核以减少错误的特征匹配。EGMN[189] 在 STM 的基础上使用了图神经网络，将每一帧视作节点，将视图间的关系视作边。然而，现有基于特征匹配的方法在不包含目标物体的区域中也进行了特征匹配，因此无法有效区分外观相似的物体。

### 5.2.2 三维场景理解与重建

理解并重建场景中的三维物体是计算机图形学和计算机视觉领域长久研究的问题。3D-SIS[91] 使用了语义分割算法获得场景的语义信息。它借助卷积神经网络融合了来自彩色和深度图像的信息，实现了多视角 RGB-D 图像的实例级体素语义分割。这些多视角的图像特征被反投影至一个更大的三维网格（3D Grid）中，从而聚合了来自不同视角信息。随着大规模三维数据集的普及，越来越多的方法[92–97]得以使用“检测-检索”的策略恢复场景中的物体的完整三维结构。然而三维模型数据库无法穷举真实世界中全部的三维模型，从而导致这些方法无法重建不包含

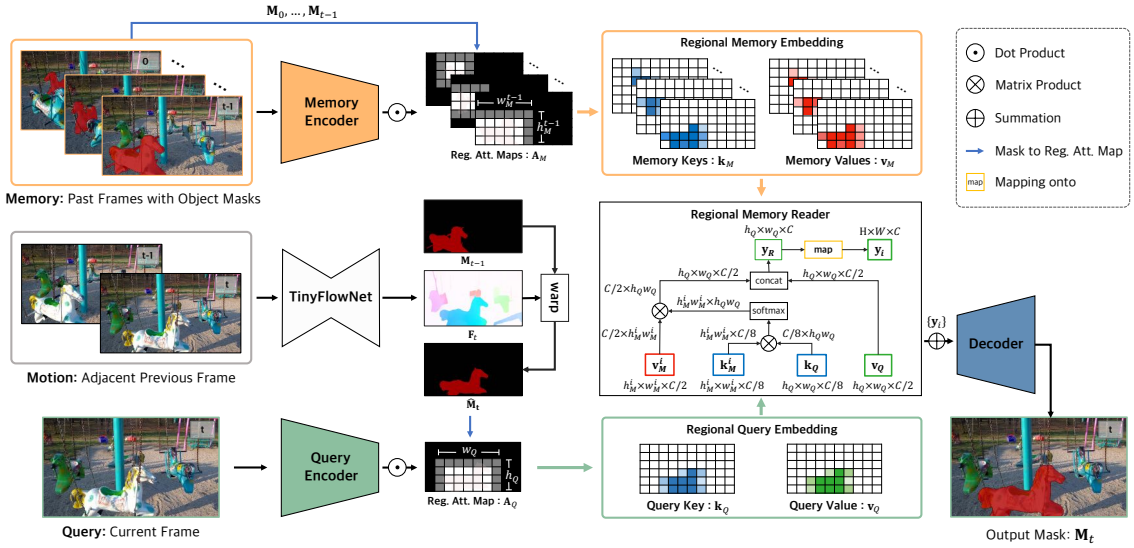



图 5-2 RMNet 的总体框架图
Fig.5-2 Overview of RMNet

在其中的物体的三维结构。为了解决这个问题，一些方法在推理时不再依赖三维模型数据集，直接识别场景中的物体，并借助神经网络从大量数据中学习的先验知识重建物体的三维结构。作为这个方向的先驱者，Voxlets[99] 通过随机森林从单张深度图像中推断场景中物体的完整的三维体素结构。Mesh R-CNN[100] 利用目标实例分割任务的通用 Mask R-CNN[101] 获取场景中包含的物体，并用另一个分支预测该物体的三维结构。Total3DUnderstanding[102] 和 FroDo[103] 作为近期的工作，均使用了二维目标检测获取场景中物体的二维边界框，并使用了三维目标检测获取物体的位姿。它们在使用二维边界框裁剪图片后，将包含物体的图片输入至重建网络恢复物体的完整三维结构，再根据估计所得到的物体位姿将重建后的物体放回至场景中。CoReNet[104] 中提出的光线跟踪残差连接（Ray-traced Skip Connections）使得图像中所有的二维信息都可以被准确的映射至三维空间中，使得该网络同时重建单张彩色图像中全部的物体。然而，这些方法仅能以单视角的图像作为输入，无法利用多视角图像消除单视角三维重建所造成的不适定问题。

## 5.3 基于局部特征记忆网络的视频物体分割

### 5.3.1 模型与方法

为了从图像序列中分离目标物体，并且对相似的目标物体具有更好的鲁棒性，本文提出了基于局部特征记忆网络的视频物体分割方法，命名为 RMNet。所提出的 RMNet 的总体框架图如图 5-2所示。和 STM[190] 一样，当前帧被视作查询帧

算法 5-1 RMNet 算法
Algo.5-1 The RMNet algorithm

**Input:** 图像序列 $\mathcal{I} = \{\mathbf{I}_i\}_{i=1}^{n}$ 和第一帧目标物体的掩膜 $\mathbf{M}_1$

**Output:** 图像序列的目标物体掩膜 $\mathcal{M} = \{\mathbf{M}_i\}_{i=2}^{n}$

将 $\mathbf{I}_1$ 和 $\mathbf{M}_1$ 拼接后输入记忆编码器，生成全局特征键值对 $\mathbf{k}_M^G$ 和 $\mathbf{v}_M^G$；

通过公式 5-1估计目标物体的局部注意力图 $\mathbf{A}_1$；

将 $\mathbf{A}_1$ 点乘至 $\tilde{\mathbf{k}}_M^1$ 和 $\tilde{\mathbf{v}}_M^1$ 获得局部记忆特征 $\mathbf{k}_M^1$ 和 $\mathbf{v}_M^1$；

**for** *t = 2 to n* **do**

  给定输入帧 $\mathbf{I}_{t-1}$ 和 $\mathbf{I}_t$，TinyFlowNet 估算后向光流 $\mathbf{F}_t$；

  给定 $\mathbf{F}_t$ 和 $\mathbf{M}_{t-1}$，通过 Backward Warp 估计当前帧掩膜 $\hat{\mathbf{M}}_t$；

  通过公式 5-1估计当前帧目标物体的局部注意力图 $\mathbf{A}_t$；

  给定 $\mathbf{I}_t$，查询编码器生成全局特征键值对 $\tilde{\mathbf{k}}_Q^t$ 和 $\tilde{\mathbf{v}}_Q^t$；

  将 $\mathbf{A}_t$ 点乘至 $\tilde{\mathbf{k}}_Q^t$ 和 $\tilde{\mathbf{v}}_Q^t$ 获得局部查询特征 $\mathbf{k}_Q^t$ 和 $\mathbf{v}_Q^t$；

  给定 $\{\mathbf{k}_M^i\}_{i=1}^{t-1}$、$\{\mathbf{v}_M^i\}_{i=1}^{t-1}$、$\mathbf{k}_Q^t$、$\mathbf{v}_Q^t$，局部记忆阅读器根据公式 5-5生成关联性特征 $\mathbf{Y}^t$；

  给定 $\mathbf{Y}^t$，解码器输出当前帧目标物体的掩膜 $\mathbf{M}_t$

**end**

（Query Frame），过去帧以及估计的物体掩膜被用于记忆帧（Memory Frame）存储于记忆网络中。对于 STM 中的记忆帧和查询帧，整张图像的特征都被存储下来；而 RMNet 仅使用目标物体所在区域的特征，并为记忆帧和查询帧中的目标物体分别生成对应的局部记忆特征和局部查询特征。局部记忆特征和局部查询特征是通过从记忆编码器（Memory Encoder）和查询编码器（Query Encoder）抽取的特征表示（Feature Embedding）和局部注意力图（Regional Attention Map）点乘得到的。局部记忆特征和局部查询特征都包含一个局部键（Regional Key）和局部值（Regional Value）。

在 STM 中，时空记忆阅读器（Space-Time Memory Reader）被用于在记忆帧和查询帧中进行全局的特征匹配。而在 RMNet 中，局部记忆阅读器（Regional Memory Reader）仅在包含目标物体的局部记忆特征和局部查询特征进行局部特征匹配，不仅减少了对相似物体的错配，而且提高了特征匹配效率。局部记忆阅读器的输出被输入至解码器（Decoder）生成目标物体在查询帧的掩膜。所提出的 RMNet 算法流程如算法 5-1所示。

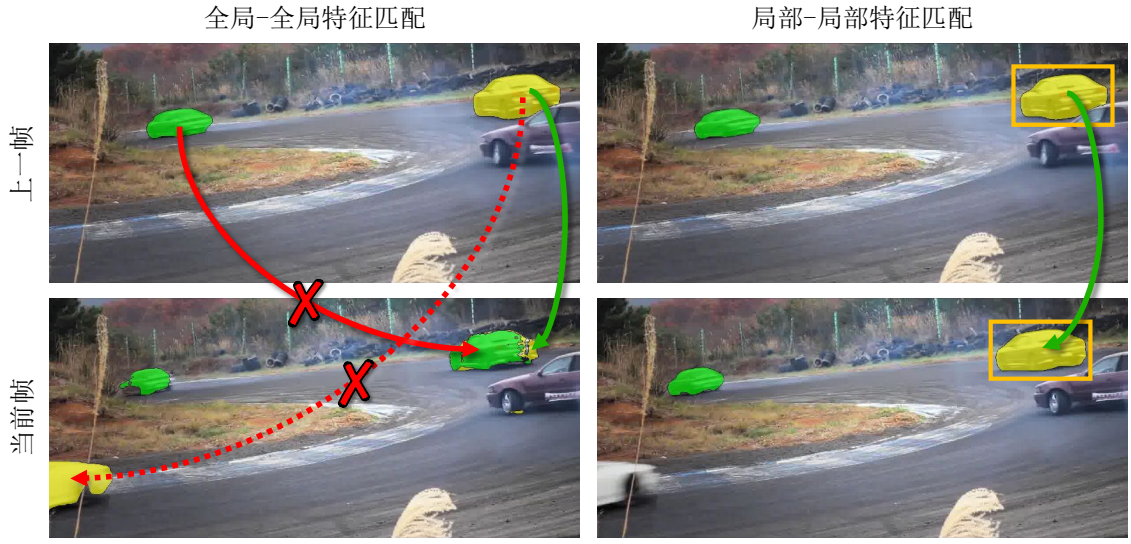



图 5-3 全局-全局匹配和局部-局部匹配的正确和错误的特征匹配

Fig.5-3 The feature matchings in the global-to-global matching and local-to-local matching

**（1）局部特征表示**

近期基于 STM 的方法[189–191] 为过去帧的每张图都构建了全局的特征表示。然而，对于记忆帧，目标物体之外的特征可能导致特征匹配过程中对于相似物体错误匹配，如图 5-3中的红色实线所示。为了解决这个问题，RMNet 引入了**局部记忆特征**，仅在包含目标物体的区域内进行特征匹配。具体而言，对于 $i$ 时刻，在特征尺度上目标物体的掩膜可表示为 $\mathbf{M}_i \in \mathbb{N}^{H\times W}$，则对于第 $j$ 个物体的局部注意力图 $\mathbf{A}_i^j \in \mathbb{R}^{H\times W}$ 可通过如下方式计算获得：

$$\mathbf{A}_i^j(x,y) = \begin{cases} 1, & x_{\min} \le x \le x_{\max} \text{ and } y_{\min} \le y \le y_{\max} \\ 0, & \text{otherwise} \end{cases} \tag{5-1}$$

其中 $(x_{\min}, y_{\min})$ 和 $(x_{\max}, y_{\max})$ 分别表示目标物体边界框左上角和右下角的坐标。这两个坐标是通过如下公式确定的：

$$\begin{aligned} x_{\min} &= \max((\arg\min_x \mathbf{M}_i(x,y) = j) - \phi, 0) \\ x_{\max} &= \min((\arg\max_x \mathbf{M}_i(x,y) = j) + \phi, W) \\ y_{\min} &= \max((\arg\min_y \mathbf{M}_i(x,y) = j) - \phi, 0) \\ y_{\max} &= \min((\arg\max_y \mathbf{M}_i(x,y) = j) + \phi, H) \end{aligned} \tag{5-2}$$

其中 $\phi$ 表示边界框的膨胀像素值，它决定了对于所估计物体掩膜的错误容忍度。特别地，当第 $j$ 个物体在 $\mathbf{M}_i$ 中消失时，$\mathbf{A}_i^j = 0$。给定第 $j$ 个物体在记忆帧中的局部

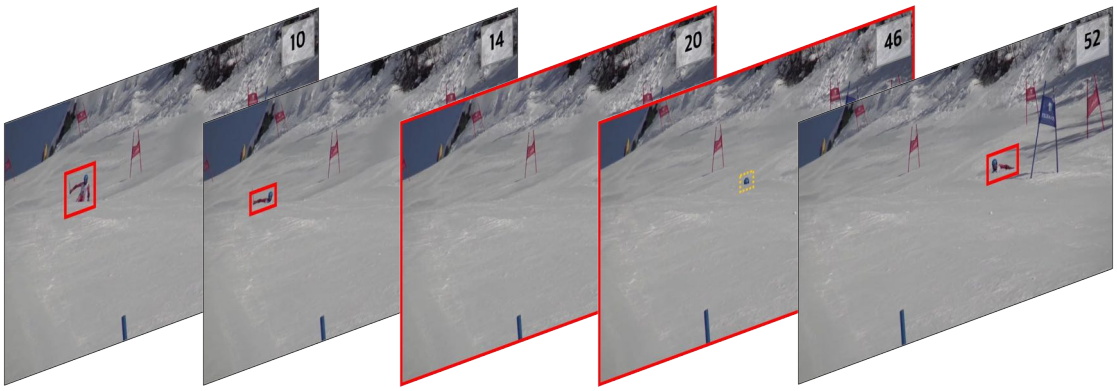



图 5-4 在目标物体被遮挡前后特征匹配区域的变化

Fig.5-4 The changes of matching regions for the target object before and after occlusion

注意力图 $\mathbf{A}_M^j = [\mathbf{A}_0^j, \ldots, \mathbf{A}_{t-1}^j]$，局部记忆特征中的局部键 $\mathbf{k}_M^j$ 和局部值 $\mathbf{v}_M^j$ 是由记忆编码器生成的全局记忆特征表示与局部注意力图 $\mathbf{A}_M^j$ 点乘得到的。

和局部记忆特征类似，RMNet 使用了**局部查询特征**缓解查询帧中对于相似物体的错误匹配，如图 5-3中的红色虚线所示。为了尽可能准确地获取当前帧中目标物体所在的区域，RMNet 为该物体跟踪并预测了一个粗略的掩膜 $\hat{\mathbf{M}}_t^j$。具体而言，上一帧物体的 $\mathbf{M}_{t-1}^j$ 通过所提出的 TinyFlowNet 估计的光流映射到当前帧，得到当前帧目标物体的掩膜 $\hat{\mathbf{M}}_t^j$。和局部记忆特征一样，$\hat{\mathbf{M}}_t^j$ 被用于生成查询帧中第 $j$ 个物体的局部注意力图 $\mathbf{A}_Q^j$。为了更好地处理对目标物体的遮挡，当目标物体的像素个数小于 $\eta$ 时，$\mathbf{A}_Q^j = 1$，这将会在查询帧中触发对目标物体的全局搜索。如图 5-4所示，当目标物体消失时，特征匹配区域（用红色边界框表示）将会被扩展至全图；当目标物体再次出现时，特征匹配区域重新聚焦至包含目标物体的区域。这个机制可以有利于基于光流的跟踪，使得网络可以感知目标物体的消失，使得网络可以对物体的遮挡更加鲁棒。和局部记忆特征类似，局部查询特征中的局部键 $\mathbf{k}_Q^j$ 和局部值 $\mathbf{v}_Q^j$ 是由查询编码器生成的全局查询特征表示与局部注意力图 $\mathbf{A}_Q^j$ 点乘得到的。

**（2）局部记忆阅读器**

在 STM[190] 中，时空记忆阅读器被用于度量查询帧和记忆帧的相似度。给定第 $j$ 个物体记忆特征中的键 $\mathbf{k}_M^j = \{k_M^j(\mathbf{p})\} \in \mathbb{R}^{T\cdot H\cdot W\times C/8}$ 和查询特征中的键 $\mathbf{k}_Q^j = \{k_Q^j(\mathbf{q})\} \in \mathbb{R}^{H\cdot W\times C/8}$，则 $\mathbf{p}$ 和 $\mathbf{q}$ 的相似度可被计算为：

$$s^j(\mathbf{p},\mathbf{q}) = \exp\left(k_M^j(\mathbf{p})k_M^j(\mathbf{q})^{\mathrm{T}}\right) \tag{5-3}$$

其中 $C$ 表示特征键中的通道数，$T$ 表示记忆帧的数量。令 $\mathbf{p} = [p_t, p_x, p_y]$ 和 $\mathbf{q} =$

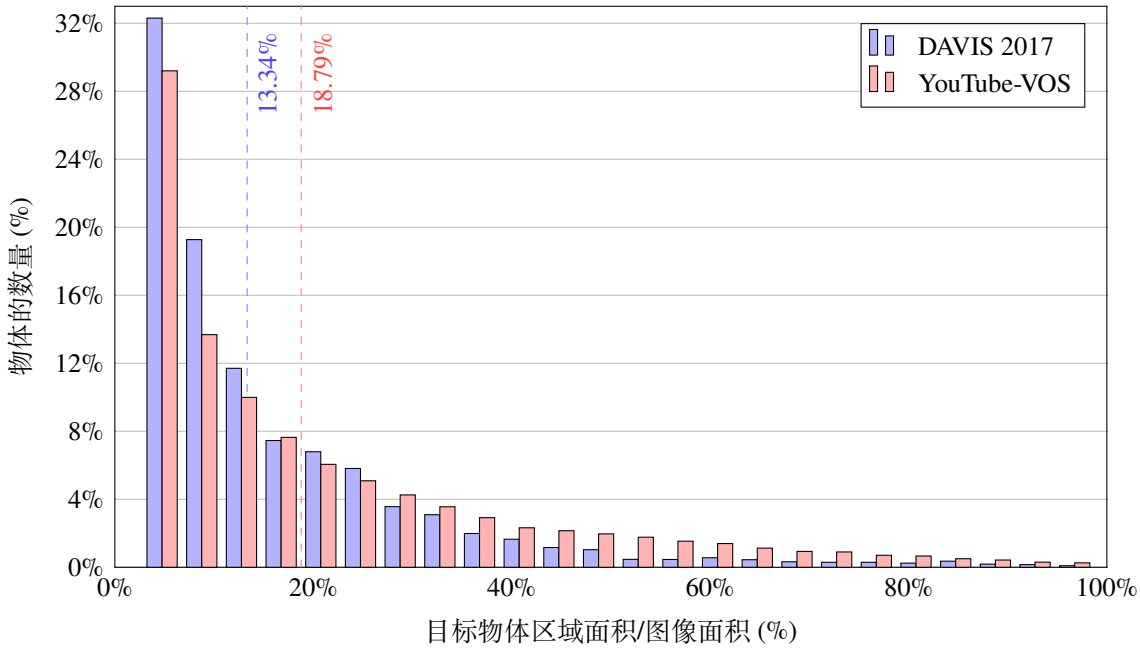



图 5-5 在 DAVIS 2017 和 YouTube-VOS 的训练集上目标物体边界框占全图比例的长尾分布
Fig. 5-5 The long tail distribution of the area ratio of the bounding boxes of target objects on the training set of the DAVIS 2017 and YouTube-VOS datasets

$[q_x, q_y]$ 分别表示在 $\mathbf{k}_M^j$ 和 $\mathbf{k}_Q^j$ 中的索引值，则在 $\mathbf{q}$ 位置上的查询值可被计算为：

$$v^j(\mathbf{q}) = \sum_{\mathbf{p}} \frac{s(\mathbf{p},\mathbf{q})}{\sum_{\mathbf{p}} s(\mathbf{p},\mathbf{q})} v_M^j(\mathbf{p}) \tag{5-4}$$

其中 $\mathbf{v}_M^j = \{v_M^j(\mathbf{p})\} \in \mathbb{R}^{T\times H\times W\times C/2}$ 表示记忆特征中的值。最终，时空记忆阅读器在 $\mathbf{q}$ 位置上的输出为：

$$y^j(\mathbf{q}) = \left[v_Q^j(\mathbf{q}), v^j(\mathbf{q})\right] \tag{5-5}$$

其中，$\mathbf{v}_Q^j = \{v_Q^j(\mathbf{q})\} \in \mathbb{R}^{H\times W\times C/2}$ 代表查询特征的值；$[\cdot]$ 表示拼接（Concatenation）。

基于局部特征表示，RMNet 中使用了局部记忆阅读器仅在包含目标物体的区域内进行局部-局部的特征匹配，如图 5-2所示。和之前工作中[189,191,202] 使用的全局-全局的记忆阅读器相比，局部记忆阅读器可以减少在记忆帧和查询帧中对于相似物体的错误特征匹配。令 $\mathcal{R}_M^j = \{\mathbf{p}\}$ 和 $\mathcal{R}_Q^j = \{\mathbf{q}\}$ 分别表示第 $j$ 个目标物体在记忆帧和查询帧中进行特征匹配的区域。在全局-全局的特征匹配中，像素之间的相似度是通过一个大矩阵乘法获得的，即 $\mathcal{R}_M^j$ 和 $\mathcal{R}_Q^j$ 为特征表示键值中的所有位置。而在局部记忆阅读器中，$\mathcal{R}_M^j$ 和 $\mathcal{R}_Q^j$ 分别定义为：

$$\begin{aligned}\mathcal{R}_M^j &= \left\{\mathbf{p}|\mathbf{A}_M^j(\mathbf{p}) \neq 0\right\} \\ \mathcal{R}_Q^j &= \left\{\mathbf{q}|\mathbf{A}_Q^j(\mathbf{q}) \neq 0\right\}\end{aligned} \tag{5-6}$$

对于满足 $\mathbf{p} \notin \mathcal{R}_M^j$ 或 $\mathbf{q} \notin \mathcal{R}_Q^j$ 的位置，它们的相似度定义为：

$$s^j(\mathbf{p},\mathbf{q}) = 0, \mathbf{p} \notin \mathcal{R}_M^j \text{ or } \mathbf{q} \notin \mathcal{R}_Q^j \tag{5-7}$$

令 $h_Q^j$ 和 $w_Q^j$ 分别表示查询帧中第 $j$ 个物体所在区域的高度和宽度，$h_M^j$ 和 $w_M^j$ 分别表示记忆帧中包含目标物体的区域的最大高度和最大宽度。因此，时空记忆阅读器的时间复杂度为 $O(TCH^2W^2)$。相比之下，局部记忆阅读器的时间复杂度被减小至 $O(TCh_Q^j w_Q^j h_M^j w_M^j)$。如图 5-5所示，$h_Q^j, h_M^j \ll H$ 且 $w_Q^j, w_M^j \ll W$。时空记忆阅读器本质上是一个非局部神经网络（Non-Local Neural Network）[206]，而它经常因为全局-全局特征匹配所导致的高计算复杂度被人诟病[207]；而所提出的局部记忆阅读器通过局部-局部的特征匹配使得其的时间复杂度显著降低。

**（3）网络结构**

TinyFlowNet 在 RMNet 中被用于估计相邻两帧的光流，从而感知物体的运动。和之前光流估计的工作[208–211] 不同，出于对速度的追求，TinyFlowNet 中并未使用任何耗时的操作，比如关联层（Correlation Layer）[208]、Cost Volume 层[210,211] 和空洞卷积层[209]。TinyFlowNet 使用了较小的输入和输出的通道数以减少参数量。最终，TinyFlowNet 的参数量仅为 FlowNetS[212] 的 1/3。为了进一步加快计算速度，输入至 TinyFlowNet 的图像被缩小为原来的 1/2。

记忆编码器以彩色图像和目标物体的掩膜作为输入，其中目标物体的掩膜以 0 至 1 之间的概率值表示；查询编码器仅以彩色图像作为输入。记忆编码器和查询编码器均使用了 ResNet50[132] 作为骨干网络。由于记忆编码器的输入包含 4 个通道，因此其中的 ResNet50 的第一层的输入通道数被改为 4。这两个编码器所输出的特征表示键和特征表示值是通过将 ResNet50 所生成的原图大小为 1/16 的特征图输入至 2 个并行的卷积层得到的。

给定局部记忆阅读器的输出，解码器输出当前帧目标物体的掩膜。解码器由一个残差块和两个微调块[190] 组成，将低分辨率的特征图逐渐上采样至原图大小。

### 5.3.2 实验结果与分析

**（1）数据集**

DAVIS 2016 数据集[213] 是视频物体分割任务主流的数据集之一，它的验证集

包含了 20 个视频，每个视频中包含 1 个物体。DAVIS 2017 数据集[214] 则是对 DAVIS 2016 数据集的多物体扩展版本，其训练集和验证集分别包含 60 个和 30 个视频。

YouTube-VOS 数据集[215] 是目前最大规模的视频物体分割的数据集。它包含 4,453 个视频，每个视频均包含多个物体的标注。其中来自 65 个类别的 3,471 个视频用于训练，剩余来自 26 个陌生类别的 507 个视频被用于测试。

**（2）度量指标**

和 STM 一样[190]，本文使用区域相似度（Region Similarity）和轮廓精确度（Contour Accuracy）作为度量指标。

区域相似性 $\mathcal{J}$ 被用于度量标注错误像素的数量。它是所估计的掩膜 $M$ 和对应 Ground Truth $G$ 之间的 Intersection over Union 函数，可形式化描述为：

$$\mathcal{J} = \left| \frac{M \cap G}{M \cup G} \right| \tag{5-8}$$

轮廓精确度 $\mathcal{F}$ 用于度量分割边界的准确率。将掩膜看成一系列闭合轮廓的集合，并计算基于轮廓的 F 度量，即准确率和召回率的函数。因此，轮廓精确度是对基于轮廓的准确率和召回率的 F 度量。令 $c(M)$ 和 $c(G)$ 分别表示掩膜 $M$ 和 $G$ 的闭合轮廓集合。轮廓精确度 $\mathcal{F}$ 可形式化描述为：

$$\mathcal{F} = \frac{2P_c R_c}{P_c + R_c} \tag{5-9}$$

其中 $P_c$ 和 $R_c$ 分别表示 $c(M)$ 和 $c(G)$ 的精度和召回率，它们是通过二分图匹配（Bipartite Graph Matching）[216] 计算得到的。

**（3）实现细节**

本文使用 PyTorch[128] 和 CUDA 实现了所提出的方法①，并使用了两块 NVIDIA Tesla V100 GPU 进行训练。参数 $\phi$ 和 $\eta$ 分别被设置为 4 和 10。在训练时，Batch Size 被设置为 4，并使用了 $\beta_1 = 0.9$ 和 $\beta_2 = 0.999$ 的 Adam[129] 优化器；同时，所有 Batch Normalization 的参数都被固定。和 STM 一样[190]，网络的训练被分为两个阶段。首先，使用不同的参数对静态图片进行仿射变换所得到的合成数据集进行预训练。然后，使用 DAVIS 和 YouTube-VOS 数据集对网络进行微调。初始的学习率被设置为 $10^{-5}$，训练一共持续 200 个 Epoch。

**（4）与现有方法的比较**

对于单个物体分割，所提出的 RMNet 和其他方法在 DAVIS 2016 数据集上进

① 代码已开源：https://github.com/hzxie/RMNet

表 5-1 在 DAVIS 2016 验证集上的量化结果
Table 5-1 The quantitative evaluation on the DAVIS 2016 validation set

| 方法 | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | Avg. | 运行时间 (秒) |
|---|---|---|---|---|
| OnAVOS [217] | 0.861 | 0.849 | 0.855 | 0.823 |
| OSVOS [203] | 0.798 | 0.806 | 0.802 | 0.642 |
| MaskRNN [185] | 0.807 | 0.809 | 0.808 | - |
| RGMP [204] | 0.815 | 0.820 | 0.818 | 0.104 |
| FAVOS [199] | 0.824 | 0.795 | 0.810 | 0.816 |
| CINN [197] | 0.834 | 0.850 | 0.842 | - |
| LSE [218] | 0.829 | 0.803 | 0.816 | - |
| VideoMatch [205] | 0.810 | 0.808 | 0.819 | - |
| PReMVOS [186] | 0.849 | 0.886 | 0.868 | 3.286 |
| A-GAME † [219] | 0.822 | 0.820 | 0.821 | 0.258 |
| FEELVOS † [192] | 0.817 | 0.881 | 0.822 | 0.286 |
| STM † [190] | 0.887 | 0.899 | 0.893 | 0.097 |
| KMN † [191] | **0.895** | **0.915** | **0.905** | - |
| CFBI † [193] | 0.883 | 0.905 | 0.894 | 0.156 |
| RMNet | 0.806 | 0.823 | 0.815 | **0.084** |
| RMNet † | 0.889 | 0.887 | 0.888 | **0.084** |

表 5-2 在 DAVIS 2017 验证集上的量化结果
Table 5-2 The quantitative evaluation on the DAVIS 2017 validation set

| 方法 | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | Avg. |
|---|---|---|---|
| OnAVOS [217] | 0.645 | 0.713 | 0.679 |
| OSMN [220] | 0.525 | 0.571 | 0.548 |
| OSVOS [203] | 0.566 | 0.618 | 0.592 |
| RGMP [204] | 0.648 | 0.686 | 0.632 |
| FAVOS [199] | 0.546 | 0.618 | 0.582 |
| CINN [197] | 0.672 | 0.742 | 0.707 |
| VideoMatch [205] | 0.565 | 0.682 | 0.624 |
| PReMVOS [186] | 0.739 | 0.817 | 0.778 |
| A-GAME † [219] | 0.672 | 0.727 | 0.700 |
| FEELVOS † [192] | 0.691 | 0.740 | 0.716 |
| STM † [190] | 0.792 | 0.843 | 0.818 |
| KMN † [191] | 0.800 | 0.856 | 0.828 |
| EGMN † [189] | 0.800 | 0.859 | 0.829 |
| CFBI † [193] | 0.791 | 0.846 | 0.819 |
| RMNet | 0.728 | 0.772 | 0.750 |
| RMNet † | **0.810** | **0.860** | **0.835** |

行了对比。DAVIS 数据集仅包含极少数量的视频，因此在训练时会造成过拟合，从而影响泛化性能。和近期的工作一样[190,191,193,199]，RMNet 在训练时额外使用了来

表 5-3 在 DAVIS 2017 test-dev 集上的量化结果
Table 5-3 The quantitative evaluation on the DAVIS 2017 test-dev set

| 方法 | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | Avg. |
|---|---|---|---|
| OnAVOS[217] | 0.534 | 0.596 | 0.565 |
| OSMN[220] | 0.377 | 0.449 | 0.413 |
| RGMP[204] | 0.513 | 0.544 | 0.529 |
| PReMVOS[186] | 0.675 | 0.757 | 0.716 |
| FEELVOS[192] | 0.552 | 0.605 | 0.578 |
| STM[190] | 0.680 | 0.740 | 0.710 |
| CFBI[193] | 0.711 | **0.785** | 0.748 |
| RMNet | **0.719** | 0.781 | **0.750** |

表 5-4 在 YouTube-VOS 验证集（2018 版本）上的量化结果
Table 5-4 The quantitative evaluation on the YouTube-VOS validation set (2018 version)

| 方法 | 已知类别 | | 未知类别 | | Avg. |
|---|---|---|---|---|---|
| | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | |
| OnAVOS[217] | 0.601 | 0.627 | 0.466 | 0.514 | 0.552 |
| OSMN[220] | 0.600 | 0.601 | 0.406 | 0.440 | 0.512 |
| OSVOS[203] | 0.598 | 0.605 | 0.542 | 0.607 | 0.588 |
| RGMP[204] | 0.595 | - | 0.452 | - | 0.538 |
| BoLTVOS[221] | 0.716 | - | 0.643 | - | 0.711 |
| PReMVOS[186] | 0.714 | 0.759 | 0.565 | 0.637 | 0.669 |
| A-GAME[219] | 0.678 | - | 0.608 | - | 0.661 |
| STM[190] | 0.797 | 0.842 | 0.728 | 0.809 | 0.794 |
| KMN[191] | 0.814 | 0.856 | 0.753 | 0.833 | 0.814 |
| EGMN[189] | 0.807 | 0.851 | 0.740 | 0.809 | 0.802 |
| CFBI[193] | 0.811 | **0.858** | 0.753 | **0.834** | 0.814 |
| RMNet | **0.821** | 0.857 | **0.757** | 0.824 | **0.815** |

自 YouTube-VOS 的数据，以解决在训练时过拟合的问题，这些方法在表中使用 † 记号表示。在表 5-1的实验结果表明，RMNet 和现有方法的性能相差无几。所有的方法都在配备有 NVIDIA Tesla V100 GPU 的机器上进行了运行时间的对比，同时在对比时排除了 IO 时间的干扰。表 5-1的结果表明，RMNet 具有更快的运算速度。

为了进一步验证对于多物体分割的性能，所提出的方法在 DAVIS 2017 验证集上和其他方法进行了对比，其性能如表 5-2所示。该实验结果表明，RMNet 的性能超越了其他所有的方法。在使用了额外的 YouTube-VOS 数据进行训练后，RMNet 可以取得更好的性能，并且依然超越了其他所有的方法，这些方法在表中使用 † 记号表示。所提出 RMNet 的性能也在 DAVIS 2017 的 test-dev 集上进行了验证，相比

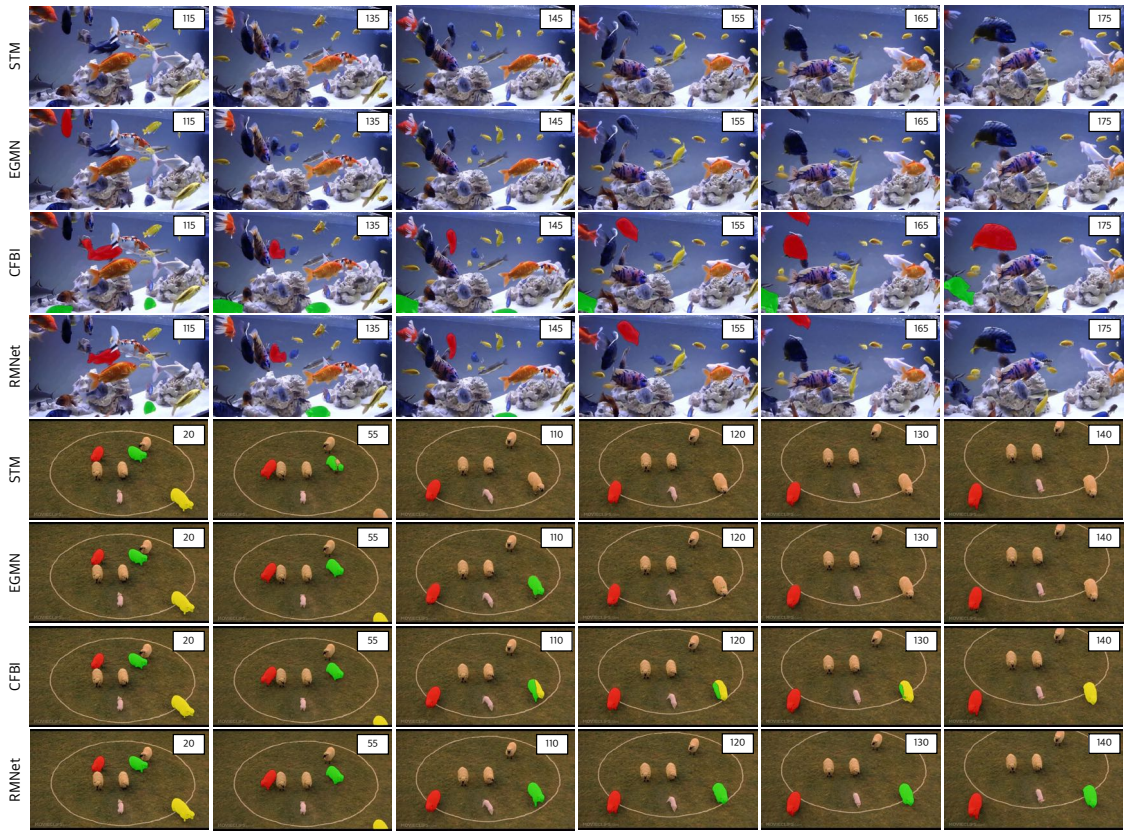



图 5-6 在 YouTube-VOS 验证集上的多物体分割结果的对比

Fig. 5-6 The comparison of multi-object video segmentation on the YouTube-VOS validation set

于 DAVIS 2017 的验证集，这里面包含更多具有挑战的样本。如表 5-3所示，RMNet 的性能超越了其他所有的方法和其他最近的工作[190,191] 一样，RMNet 的性能也在 YouTube-VOS 验证集 (2018 版本) 进行了验证。如表 5-4所示，RMNet 取得了均值为 0.815 的性能，并且超越了其他所有方法的性能。图 5-6展示了在 YouTube-VOS 上的量化结果。所有的方法均在 720p 的分辨率下测试，并且未使用任何测试时增强的方法。对于第一个视频，STM[190] 和 EGMN[189] 在 115 帧之前便已无法分割目标物体。该实验结果表明，RMNet 可以更有效地区分视频中出现的相似物体。

**（5）消融实验**

本文在 DAVIS 2017 验证集上开展了消融实验以验证 RMNet 中组件的有效性。

局部记忆阅读器在记忆帧和查询帧中进行局部特征匹配，从而减小减少了对相似物体的错误匹配并节省了计算时间。为了验证局部记忆阅读器的有效性和高效性，本文对比了记忆帧或查询帧中使用全局记忆阅读器的结果。表 5-5的结果表明，相比于全局记忆阅读器，局部记忆阅读器可以在精度和效率上取得更好的性能。此外，根据图 5-5所示，目标物体所在区域占全图的面积通常小于 20%，理论

表 5-5 局部记忆阅读器中全局特征匹配和局部特征匹配的性能对比

Table 5-5 The comparison between the global and local feature matching in Regional Memory Reader

| M.R. | Q.R. | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | Avg. | 特征匹配时间 (毫秒) |
|---|---|---|---|---|---|
| × | × | 0.792 | 0.843 | 0.818 | 10.68 |
| ✓ | × | 0.798 | 0.847 | 0.822 | 5.50 |
| × | ✓ | 0.803 | 0.853 | 0.828 | 5.50 |
| ✓ | ✓ | **0.810** | **0.860** | **0.835** | **2.09** |


图 5-7 目标物体帧间相似度对于分割结果的影响

Fig. 5-7 The impact of the similarity of the target object in the adjacent frames

上可以将 FLOPS 减少为原来的 1/25。图 5-7给出了在不同记忆阅读器中相似度得分的对比结果。其中，更明亮的颜色表示更高的相似度得分，“M.R.’’ 和 “Q.R.’’ 分别表示在局部记忆特征和局部查询特征中的记忆区域和查询区域。对于红色边界框所标出的目标物体，所提出的局部记忆阅读器有效避免了在记忆帧和查询帧中的错误匹配，从而提升了分割的性能。

在 RMNet 中，查询帧中的特征匹配区域是通过上一帧物体的位置和所估计光流共同决定的，记为 ‘‘基于光流的区域’’。在 FEELVOS[192] 中，查询帧中的特征匹配区域是通过上一帧物体的位置决定的；它在上一帧物体出现的位置附近进行特征匹配。而 KMN[191] 使用了由二维高斯核决定的区域，以相似度最高的像素作为该区域的中心。为了证明本文所使用的 ‘‘基于光流的区域’’ 的有效性，将它与其他特征匹配区域的生成方式进行了对比。在表 5-6中，‘‘上一帧的区域’’ 和 ‘‘最佳匹配区域’’ 分别代表 FEELVOS 和 KMN 所使用的区域生成方式。如表 5-6所示，‘‘基于光流的区域’’ 相比于其他两种区域可以取得更好的性能。‘‘上一帧的区域’’ 是基于相邻两帧物体的运动较小的假设，这使得该方法无法很好地处理物体的遮挡和漂移。‘‘最佳匹配区域’’ 只考虑了最佳匹配的像素，然而该像素非常容易被光照条件所干扰，因此无法很好地区分外观相似的物体。

TinyFlowNet 被用于估计相邻帧的光流。为了验证所提出的 TinyFlowNet 的有

表 5-6 不同方法所使用的特征匹配区域的性能对比
Table 5-6 The comparison of feature matching regions used in different methods

| 方法 | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | Avg. |
|---|---|---|---|
| 上一帧的区域[193] | 0.762 | 0.822 | 0.792 |
| 最佳匹配区域[191] | 0.792 | 0.845 | 0.819 |
| 基于光流的区域 | **0.810** | **0.860** | **0.835** |

表 5-7 TinyFlowNet 和其他方法在光流估计性能上的对比
Table 5-7 The comparison of TinyFlowNet compared to other methods for optical flow estimation

| Method | $\mathcal{J}$ Mean | $\mathcal{F}$ Mean | Avg. | 光流估计时间 (毫秒) |
|---|---|---|---|---|
| FlowNet2-CSS[212] | **0.814** | **0.860** | **0.837** | 59.93 |
| RAFT[222] | 0.808 | 0.859 | 0.834 | 157.78 |
| TinyFlowNet | 0.810 | **0.860** | 0.835 | **10.05** |

效性，本文对比了在 FlyingChairs[208] 预训练的 FlowNet2-CSS[212] 和 RAFT[222] 的性能。如表 5-7，当 TinyFlowNet 被 FlowNet2-CSS 或者 RAFT 替换时，分割的性能几乎是不变的；这说明 TinyFlowNet 可以满足在查询帧预测特征匹配区域的需要。此外，TinyFlowNet 的计算速度分别是 FlowNet2-CSS 和 RAFT 的 6 倍和 16 倍。

## 5.4 基于场景语义建模的多源多视三维场景和物体重建

### 5.4.1 模型与方法

为了重建场景的三维结构，并且使得场景中的物体可以直接从重建的场景中分离，本文提出了基于场景语义建模的多源多视三维场景重建方法，命名为 URNet。所提出的 URNet 的总体框架图如图 5-1所示。给定一个的图像序列，5.3节所提出的 RMNet 可以估计出每个物体在每张图像中的掩膜，并使用该掩膜将目标物体从图像序列中分离，该图像序列可以包含单目彩色图像、双目彩色图像或是深度图像。接着，多尺度上下文感知融合的多源多视三维物体重建方法可以从多张图像中恢复出每个物体的三维结构。然后，三维目标检测网络（Object Detection Network，ODN）可以估计出每个物体的三维边界框。最后，布局估计网络（Layout Estimation Network，LEN）可以估计相机位姿和场景布局参数，从而生成最终的三维场景重建结果。所提出的 URNet 的算法描述如算法 5-2所示。

估计 6D 物体姿态之前需要先估计场景的布局。和之前的工作[102,223] 一样，场景和场景中的目标物体被参数化为盒子。将世界系统设置为位于相机中心，其垂

算法 5-2 URNet 算法
Algo.5-2 The URNet algorithm

**Input:** 图像序列 $\mathcal{I} = \{\mathbf{I}_i\}_{i=1}^{n}$ 和第一帧目标物体的掩膜 $\mathbf{M}_1$

**Output:** 目标物体体素模型集合 $\mathcal{V} = \{\mathbf{V}_i\}_{i=1}^{n_{obj}}$，目标物体三维边界框集合 $\mathcal{B} = \{\mathbf{B}_i\}_{i=1}^{n_{obj}}$，相机姿态 $\mathbf{R}$ 和场景布局参数 $\mathbf{C}$

给定 $\mathbf{M}_1$，生成目标物体的二维边界框集合 $\mathcal{B}^P = \{\mathbf{B}_i^P\}_{i=1}^{n_{obj}}$；
给定 $\mathbf{I}_1$ 和 $\mathcal{B}^P$，使用 ODN 估计目标物体三维边界框集合 $\mathcal{B} = \{\mathbf{B}_i\}_{i=1}^{n_{obj}}$；
给定 $\mathbf{I}_1$；使用 LEN 估计相机姿态 $\mathbf{R}$ 和场景布局参数 $\mathbf{C}$；
给定 $\mathcal{I}$ 和 $\mathbf{M}_1$，算法 5-1估计图像序列中目标物体的掩膜 $\mathcal{M} = \{M_i\}_{i=1}^{n}$；
**for** $i = 1$ *to* $n_{obj}$ **do**
    使用目标物体掩膜 $\mathcal{M}$ 和图像序列 $\mathcal{I}$ 获得仅包含第 $i$ 个目标物体的图像序列 $\tilde{\mathcal{I}}^i$；
    使用算法 4-1重建第 $i$ 个物体的三维体素模型 $\mathbf{V}_i$；
**end**

直（y-）轴垂直于地板，其向前（x-）轴朝向相机，这样相机的位姿 $\mathbf{R}(\beta, \gamma)$ 可以由俯仰角（Pitch）和横滚角（Roll）$(\beta, \gamma)$ 决定。在世界坐标系中，一个盒子可以由 3 个参数决定：三维空间的中心 $\mathbf{C} \in \mathbb{R}^3$，三维空间的大小 $\mathbf{s} \in \mathbb{R}^3$，以及三维空间的方向角 $\theta \in [-\pi, \pi]$。对于场景中的物体，其三维中心 $\mathbf{C}$ 由其在图像平面上的二维投影 $\mathbf{c} \in \mathbb{R}^2$ 表示，并且距相机中心的距离为 $d \in \mathbb{R}$。给定相机内参矩阵 $\mathbf{K} \in \mathbb{R}^3$，$\mathbf{C}$ 可以由如下公式计算得到：

$$\mathbf{C} = \mathbf{R}^{-1}(\beta, \gamma) \cdot d \cdot \frac{\mathbf{K}^{-1}[\mathbf{c}, 1]^T}{||\mathbf{K}^{-1}[\mathbf{c}, 1]^T||_2} \tag{5-10}$$

二维投影中心 $\mathbf{c}$ 的计算可以被解耦成 $\mathbf{C}^b + \boldsymbol{\delta}$。$\mathbf{C}^b$ 是二维边界框的中心，它可以从目标物体的掩膜获得；$\delta \in \mathbb{R}^2$ 表示有待学习的偏移量。从二维边界框估计场景中目标物体的三维边界框的过程可以被看成一个函数 $\mathbf{F}(\boldsymbol{\delta}, d, \beta, \gamma, \mathbf{s}, \theta) \in \mathbb{R}^{3\times8}$。具体而言，三维目标检测网络被用于每个物体三维边界框的属性 $(\boldsymbol{\delta}, d, \mathbf{s}, \theta)$。布局估计网络用于估计相机姿态 $\mathbf{R}(\beta, \gamma)$ 和场景布局参数 $(\mathbf{C}, \mathbf{s}^l, \theta^l)$。

本文使用了 Total3D 中所提出的目标检测网络[102]，从目标物体的二维边界框估计三维边界框。之前的工作在估计目标物体的三维边界框仅考虑该物体本身[223]或两两物体之间的关系[224]；而在 Total3D 中，估计物体三维边界框时考虑了物体的多边关系，将场景中的所有物体均考虑在内，其总体框架图如图 5-8所示。具体

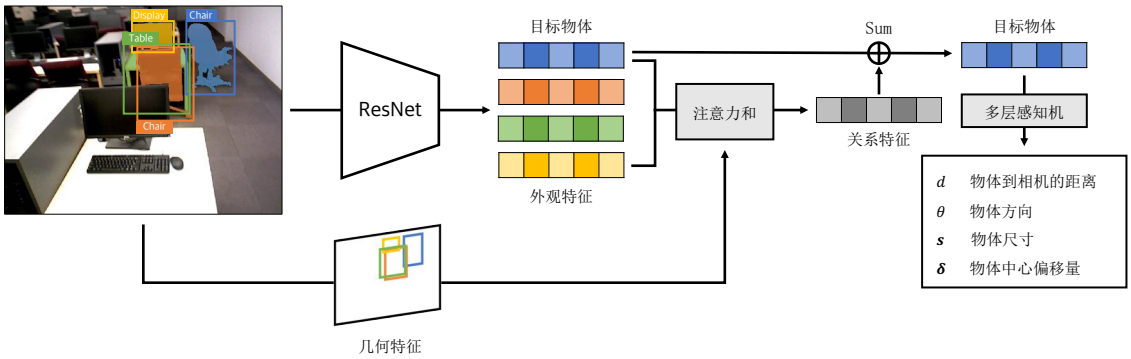



图 5-8 URNet 中所使用的三维目标检测网络的框架图

Fig. 5-8 Overview of the 3D object detection network in URNet

而言，ResNet-34 被用于抽取每个物体的外观特征（Appearnce Features），并且将二维边界框的位置和大小编码成几何特征（Geometry Feature）。对于每一个目标物体，分段特征求和[225] 被用于计算其关系特征（Relational Features）。该特征求和由目标物体与另一个目标物体在外观和几何形状上的相似性加权，如图 5-8中的‘‘注意力和（Attention Sum）’’ 所示。然后，关系特征被加至目标物体的外观特征上，并使用一个两层的感知机回归三维边界框的参数 $(\boldsymbol{\delta}, d, \mathbf{s}, \theta)$。

本文使用了 Total3D 中所提出的布局估计网络[102] 估计相机位姿 $\mathbf{R}(\beta, \gamma)$ 和场景的布局参数 $(\mathbf{C}, \mathbf{s}^l, \theta^l)$。布局估计网络和目标检测网络具有相同的网络结构，但从中移除了关系特征。$(\beta, \gamma, \mathbf{C}, \mathbf{s}^l, \theta^l)$ 由 ResNet 之后的两个多层感知机回归得到。和之前的工作一样[223]，三维空间的中心 $\mathbf{C}$ 是通过估计平均布局中心的偏移量得到的。

## 5.4.2 实验结果与分析

由于目前并没有针对本任务的三维场景重建的数据集，所提出的 URNet 和现有的三维场景重建方法（包括 Mesh R-CNN[100] 和 Total3D[102]）在 SUN3D 数据集[107] 上进行了对比。该数据集包含了 2,513,609 张室内场景的 RGB-D 图像，以及对应的相机位姿和相机参数。由于 SUN3D 数据集并未提供场景中每个物体完整的三维模型，因此无法给出定量的对比，定性的实验结果如图 5-9所示。由于 Mesh R-CNN 和 Total3D 均使用单视角图像作为输入，将图像序列中的第一张彩色图像作为算法的输入。可见，本节所提出的 URNet 在三维场景重建的质量上远超过其他两个方法，该结果同时证明了所提出方法在真实场景中的有效性。

除此之外，本文使用了 ZED 双目相机①拍摄了哈尔滨工业大学新技术实验楼

① https://www.stereolabs.com/zed/

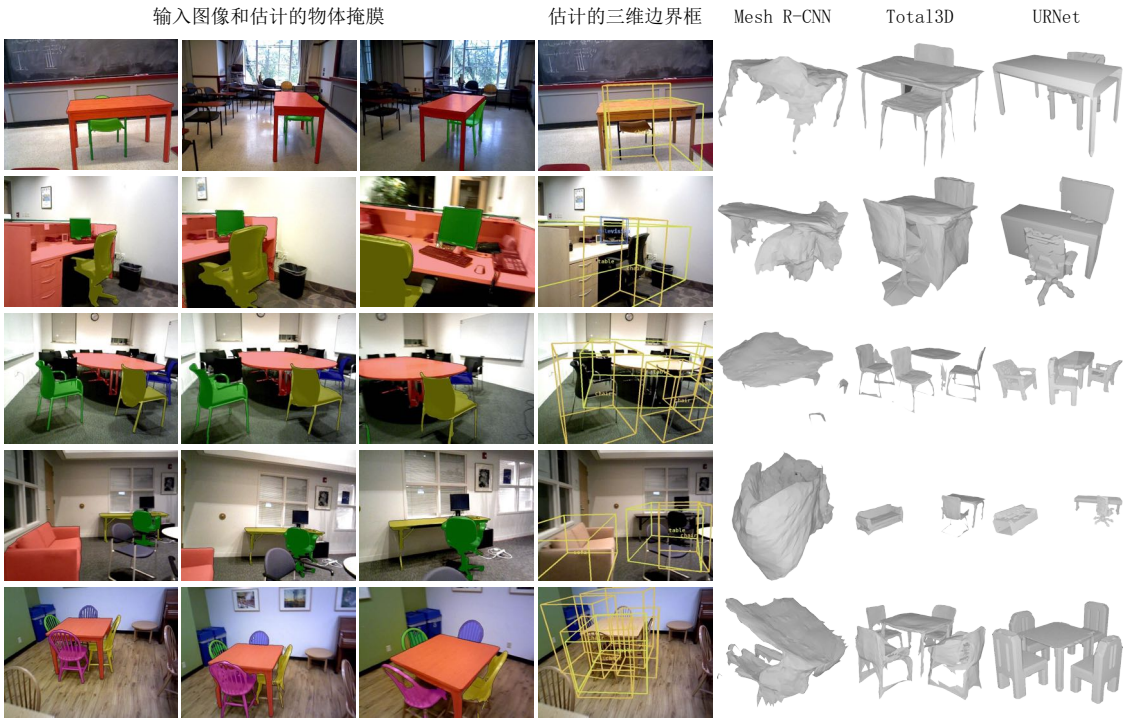



图 5-9 在 SUN3D 数据集上三维场景重建和视频物体分割结果

Fig. 5-9 The 3D scene reconstruction and video object segmentation results on SUN3D

618 会议室的场景，并对其进行了重建。图 5-10展示了三维边界框估计结果、物体掩膜估计结果、视差估计结果和场景重建结果。实验结果表明所提出的 URNet 可以较好地恢复该会议室场景中所有物体的完整三维结构而无需完整扫描待重建的物体，从而进一步证实了 URNet 的可用性。

## 5.5 本章小结

本章提出了基于局部特征记忆网络的视频物体分割方法，命名为 RMNet。该方法可以从图像序列中估计目标物体的掩膜，从而将目标物体从图像序列中分离。相比于之前视频物体分割方法，所提出的 RMNet 在 DAVIS 和 YouTube-VOS 数据集上取得了更高的精度和更快的推理速度。基于所提出的 RMNet 和多尺度上下文感知融合的多源多视三维物体方法，本文提出了基于场景语义建模的三维场景重建方法，命名为 URNet。该方法可以重建图像序列中多个物体的完整的三维形状，并估计每个物体的位姿，进而恢复场景的三维结构。和无语义的三维场景重建方法相比，URNet 可以在重建时完成对场景的语义理解，并恢复物体不可见部分的三维结构。和无语义的三维场景重建方法不同，本文可以直接从 URNet 重建的场景中分离某个物体的三维结构。在 SUN3D 数据集和实地拍摄的视频上的重建结果

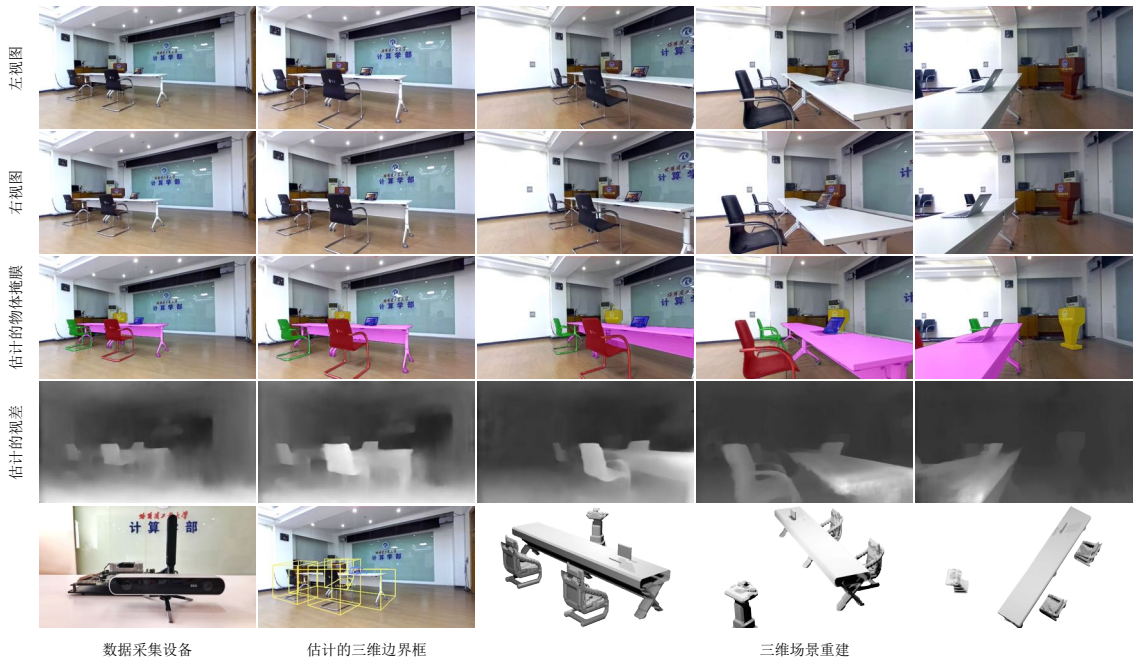



图 5-10 在哈尔滨工业大学新技术实验楼 618 会议室的三维重建结果

Fig. 5-10 The 3D scene reconstruction of Room 618, New Technology Labortory, HIT

表明，相比于其他现有的方法，所提出的 URNet 可以更好地恢复场景中物体的完整三维结构。

# 结　论

本文对三维场景和物体重建进行了深入研究，针对不同类型的相机提出了单源单视的三维物体重建方法，从彩色图像和深度图像中恢复某个物体的完整三维结构。为了融合不同数据源和不同视角的重建结果，本文提出了多尺度上下文感知融合的多源多视三维物体重建方法。以此为基础，本文进一步结合视频物体分割进一步提出了基于场景语义感知的多源多视三维场景和物体重建方法，从而从真实的彩色或深度图像序列中恢复场景的三维结构。总体而言，本文的主要的创新点包括以下几个方面：

（1）本文为单目彩色相机、双目彩色相机和深度相机分别提出了基于几何先验的单目彩色图像三维物体重建方法、基于深度感知的双目彩色图像三维物体重建方法和基于网格化残差网络的深度图像三维物体重建方法，这些方法通过几何先验从已知的三维结构中推断物体不可见的三维结构。所提出的方法可以解决 Shape from X 无法恢复物体不可见部分三维结构的问题。在 ShapeNet、StereoShapeNet、Pix3D、Driving 和 KITTI 数据集上的实验结果表明，所提出的方法相比于现有的方法有 3% 至 18% 不等的性能提升。

（2）本文提出了多尺度上下文感知融合的多源多视三维物体重建方法，更充分地利用了来自不同数据源和不同视角信息。所提出的方法使得彩色图像和深度图像的信息可以相互补充，可以更好地重建弱纹理、重复纹理和不发生反射的物体。在 ShapeNet 和 Pix3D 数据集的实验结果均表明，所提出的方法相比于现有方法在重建精度上有 4% 至 20% 的提升，并且推理速度最多提升了 7 倍。此外，本文构建了目前最大规模包含自然场景的多视角三维重建数据集——Things3D，该数据集显著提升了所提出的方法在真实场景数据集上的泛化能力。

（3）本文提出了基于场景语义感知的多源多视三维场景重建方法，使得场景中的物体可以直接从重建后场景中分离。为了实现场景语义感知，本文提出了基于局部特征记忆网络的视频物体分割方法，该方法有效减少了对于相似物体的错误匹配，从而得以更好地理解场景中的语义信息，并将目标物体从图像序列中分离。为了重建场景中的物体，本文设计了基于场景语义建模的三维场景重建方法，该方法分别为每个目标物体恢复其完整的三维结构，并估计它们的位姿和位置，进而还原场景的三维结构。和无语义的三维场景重建方法不同，本文可以直接从该

方法重建的场景中分离某个物体的三维结构。在 SUN3D 数据集和实地拍摄的视频上的实验结果验证了所提出方法的实用性和有效性。

本文对三维场景和物体重建进行了深入的探索，所提出的思想、模型和算法均通过实验验证。概括来说，本文提出了单源单视的三维物体重建方法，得以从单视角的彩色图像和深度图像恢复某个物体完整三维结构而无需完整扫描这些物体；提出了多尺度上下文感知融合的多源多视三维物体重建方法，得以更加鲁棒地融合不同数据源和不同视角的重建结果；提出了基于场景语义感知的多源多视三维场景重建方法，得以在恢复场景三维结构的同时重建场景中每个物体完整的三维结构。然而，三维场景和物体重建是一项极具挑战性的问题，目前的研究仍有一些问题亟待解决。未来的工作中，本文将从以下几个方面展开研究：

（1）**重建未知类别物体的三维结构**。尽管本文所提出的方法得以从多数据源多视角的图像序列中恢复真实场景的三维结构，但现有的方法仅能重建场景中已知类别的物体。另一方面，由于采集三维物体数据集的复杂性，导致数据集难以覆盖现实场景中所有的物体。因此，利用无监督或者自监督学习消除对大规模三维数据集的依赖并提高对于未知类别物体的泛化性能将成为未来的研究重点之一。

（2）**改善三维物体重建的细节**。本文所提出的多源多视三维物体重建方法主要以分辨率为 $32^3$ 的体素作为三维结构的主要表示形式，该分辨率造成了所恢复物体的三维结构中细节信息的丢失，而使用更高分辨率的体素又使得计算复杂度显著上升。因此，利用八叉树体素等类似的结构保留所恢复物体三维结构中的细节并保持现有的计算复杂度将会是未来研究的重点之一。

（3）**简化目标物体掩膜的生成**。本文所提出的视频物体分割方法需要给定第一帧中目标物体的掩膜，该掩膜通常需要人工标注获得，而现有的图像实例分割算法所生成掩膜的质量难以满足该算法的需求。因此，改善图像实例分割的结果或设计交互式视频物体分割（Interactive Video Object Segmentation）方法将会是未来研究的重点之一。

# 参考文献

[1] Buelthoff H H, Yuille A L. Shape-from-X: Psychophysics and computation[C] // Sensor Fusion III: 3D Perception and Recognition : Vol 1383. 1991 : 235-246.

[2] Laga H, Jospin L V, Boussaïd F, et al. A Survey on Deep Learning Techniques for Stereo-based Depth Estimation[J]. arXiv, 2020, 2006.02535.

[3] Furukawa Y, Hernández C. Multi-View Stereo: A Tutorial[J]. Foundations and Trends in Computer Graphics and Vision, 2015, 9(1-2) : 1-148.

[4] Fukushima S, Okumura T. Modeling a three-dimensional shape from a silhouette by detecting symmetry[J]. Systems and Computers in Japan, 1993, 24(3) : 59-69.

[5] Schönberger J L, Frahm J. Structure-from-Motion Revisited[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2016 : 4104-4113.

[6] Nayar S K, Nakagawa Y. Shape from focus: an effective approach for rough surfaces[C] // Proceedings of the Conference on Robotics and Automation (ICRA). 1990 : 218-225.

[7] Zhang R, Tsai P, Cryer J E, et al. Shape from Shading: A Survey[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1999, 21(8) : 690-706.

[8] Guan L, Franco J, Pollefeys M. 3D Occlusion Inference from Silhouette Cues[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2007.

[9] Aliomonos J, Swain M J. Shape from Texture[C] // Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI). 1985 : 926-931.

[10] Parodi P, Piccioli G. 3D Shape Reconstruction by Using Vanishing Points[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1996, 18(2) : 211-217.

[11] Benford S, Giannachi G. Performing mixed reality[M]. [S.l.] : The MIT Press, 2011.

[12] Rother D, Sapiro G. Seeing 3D objects in a single 2D image[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2009 : 1819-1826.

[13] Nevatia R, Binford T O. Description and Recognition of Curved Objects[J]. Artificial Intelligence, 1977, 8(1) : 77-98.

[14] Gupta A, Efros A A, Hebert M. Blocks World Revisited: Image Understanding Using Qualitative Geometry and Mechanics[C] // Lecture Notes in Computer Science, Vol 6314 : Proceedings of the European Conference on Computer Vision (ECCV), Part IV. 2010 : 482-496.

[15] Haag M, Nagel H. Combination of Edge Element and Optical Flow Estimates for 3D-Model-Based Vehicle Tracking in Traffic Image Sequences[J]. International Journal of Computer Vision, 1999, 35(3) : 295-319.

[16] Pentland A. Perceptual Organization and the Representation of Natural Form[J]. Artificial Intelligence, 1986, 28(3) : 293-331.

[17] Lim J J, Pirsiavash H, Torralba A. Parsing IKEA Objects: Fine Pose Estimation[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2013 : 2992-2999.

[18] Huang Q, Wang H, Koltun V. Single-view reconstruction via joint analysis of image and shape collections[J]. ACM Transactions on Graphics, 2015, 34(4) : 87:1-87:10.

[19] Blanz V, Vetter T. A Morphable Model for the Synthesis of 3D Faces[C] // Proceedings of the Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 1999 : 187-194.

[20] Zuffi S, Black M J. The stitched puppet: A graphical model of 3D human shape and pose[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2015 : 3537-3546.

[21] Cashman T J, Fitzgibbon A W. What Shape Are Dolphins? Building 3D Morphable Models from 2D Images[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(1) : 232-244.

[22] Vicente S, Carreira J, Agapito L, et al. Reconstructing PASCAL VOC[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2014 : 41-48.

[23] Laurentini A. The Visual Hull Concept for Silhouette-Based Image Understanding[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994, 16(2) : 150-162.

[24] Kar A, Tulsiani S, Carreira J, et al. Category-specific object reconstruction from a single image[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2015 : 1966-1974.

[25] Torrance K E, Sparrow E M. Theory for off-specular reflection from roughened surfaces[J]. Journal of the Optical Society of America, 1967, 57(9) : 1105-1114.

[26] Bakshi S, Yang Y. Shape from Shading for Non-Lambertian Surfaces[C] // Proceedings the International Conference on Image Processing (ICIP). 1994 : 130-134.

[27] Marinos C, Blake A. Shape from texture: the homogeneity hypothesis[C] // Proceedings of the International Conference on Computer Vision (ICCV). 1990 : 350-353.

[28] Loh A M, Hartley R I. Shape from Non-homogeneous, Non-stationary, Anisotropic, Perspective Texture[C] // Proceedings of the British Machine Vision Conference (BMVC). 2005.

[29] Tatarchenko M, Dosovitskiy A, Brox T. Octree Generating Networks: Efficient Convolutional Architectures for High-resolution 3D Outputs[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2017 : 2107-2115.

[30] Wu Z, Song S, Khosla A, et al. 3D ShapeNets: A deep representation for volumetric shapes[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2015 : 1912-1920.

[31] Girdhar R, Fouhey D F, Rodriguez M, et al. Learning a Predictable and Generative Vector Representation for Objects[C] // Lecture Notes in Computer Science, Vol 9910 : Proceedings of the European Conference on Computer Vision (ECCV), Part VI. 2016 : 484-499.

[32] Wu J, Zhang C, Xue T, et al. Learning a Probabilistic Latent Space of Object Shapes via 3D Generative-Adversarial Modeling[C] // Proceedings of the Conference on Neural Information Processing Systems (NIPS). 2016 : 82-90.

[33] Yang B, Rosa S, Markham A, et al. Dense 3D Object Reconstruction from a Single Depth View[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2019, 41(12) : 2820-2834.

[34] Gadelha M, Maji S, Wang R. 3D Shape Induction from 2D Views of Multiple Objects[C] // Proceedings of the International Conference on 3D Vision (3DV). 2017 : 402-411.

[35] Kato H, Harada T. Learning View Priors for Single-View 3D Reconstruction[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 9778-9787.

[36] Wu J, Wang Y, Xue T, et al. MarrNet: 3D Shape Reconstruction via 2.5D Sketches[C] // Proceedings of the Conference on Neural Information Processing Systems (NIPS). 2017 : 540-550.

[37] Wu J, Zhang C, Zhang X, et al. Learning Shape Priors for Single-View 3D Completion And Reconstruction[C] // Lecture Notes in Computer Science, Vol 11215 : Proceedings of the European Conference on Computer Vision (ECCV), Part XI. 2018 : 673-691.

[38] Zhang X, Zhang Z, Zhang C, et al. Learning to Reconstruct Shapes from Unseen Classes[C] // Proceedings of the Conference on Neural Information Processing Systems (NeurIPS). 2018 : 2263-2274.

[39] Riegler G, Ulusoy A O, Geiger A. OctNet: Learning Deep 3D Representations at High Resolutions[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 6620-6629.

[40] Hane C, Tulsiani S, Malik J. Hierarchical Surface Prediction for 3D Object Reconstruction[C] // Proceedings of the International Conference on 3D Vision (3DV). 2017 : 412-420.

[41] Fan H, Su H, Guibas L J. A Point Set Generation Network for 3D Object Reconstruction from a Single Image[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 2463-2471.

[42] Newell A, Yang K, Deng J. Stacked Hourglass Networks for Human Pose Estimation[C] // Lecture Notes in Computer Science, Vol 9912 : Proceedings of the European Conference on Computer Vision (ECCV), Part VIII. 2016 : 483-499.

[43] Jiang L, Shi S, Qi X, et al. GAL: Geometric Adversarial Loss for Single-View 3D-Object Reconstruction[C] // Lecture Notes in Computer Science, Vol 11212 : Proceedings of the European Conference on Computer Vision (ECCV), Part VIII. 2018 : 820-834.

[44] Yuan W, Khot T, Held D, et al. PCN: Point Completion Network[C] // Proceedings of the International Conference on 3D Vision (3DV). 2018 : 728-737.

[45] Tchapmi L P, Kosaraju V, Rezatofighi H, et al. TopNet: Structural Point Cloud Decoder[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 383-392.

[46] Huang Z, Yu Y, Xu J, et al. PF-Net: Point Fractal Network for 3D Point Cloud Completion[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 7659-7667.

[47] Wen X, Li T, Han Z, et al. Point Cloud Completion by Skip-Attention Network With Hierarchical Folding[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 1936-1945.

[48] Wang X, Ang M H, Lee G H. Cascaded Refinement Network for Point Cloud Completion[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 787-796.

[49] Wang N, Zhang Y, Li Z, et al. Pixel2Mesh: Generating 3D Mesh Models from Single RGB Images[C] // Lecture Notes in Computer Science, Vol 11215 : Proceedings of the European Conference on Computer Vision (ECCV), Part XI. 2018 : 55-71.

[50] Kato H, Ushiku Y, Harada T. Neural 3D Mesh Renderer[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 3907-3916.

[51] Mescheder L M, Oechsle M, Niemeyer M, et al. Occupancy Networks: Learning 3D Reconstruction in Function Space[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 4460-4470.

[52] Xu Q, Wang W, Ceylan D, et al. DISN: Deep Implicit Surface Network for High-quality Single-view 3D Reconstruction[C] // Proceedings of the Conference on Neural Information Processing Systems (NeurIPS). 2019 : 490-500.

[53] Xiong Y, Shafer S A. Depth from focusing and defocusing[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 1993 : 68-73.

[54] Darrell T, Wohn K. Depth from focus using a pyramid architecture[J]. Pattern Recognition Letters, 1990, 11(12) : 787-796.

[55] Nayar S K, Nakagawa Y. Shape from Focus[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1994, 16(8) : 824-831.

[56] Martin W N, Aggarwal J K. Volumetric Descriptions of Objects from Multiple Views[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1983, 5(2) : 150-158.

[57] Szeliski R. Rapid Octree Construction from Image Sequences[J]. CVGIP: Image Understanding, 1993, 58(1) : 23-32.

[58] Tarini M, Callieri M, Montani C, et al. Marching Intersections: An Efficient Approach to Shape-from-Silhouette[C] // Proceedings of the Fall Workshop on Vision, Modeling, and Visualization (VMV). 2002 : 283-290.

[59] Matusik W, Buehler C, McMillan L. Polyhedral Visual Hulls for Real-Time Rendering[C] // Eurographics : Proceedings of the Eurographics Workshop on Rendering Techniques. 2001 : 115-126.

[60] Hirschmüller H. Stereo Processing by Semiglobal Matching and Mutual Information[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 30(2) : 328-341.

[61] Mayer N, Ilg E, Häusser P, et al. A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2016 : 4040-4048.

[62] Mayer N, Ilg E, Fischer P, et al. What Makes Good Synthetic Training Data for Learning Disparity and Optical Flow Estimation?[J]. International Journal of Computer Vision, 2018, 126(9) : 942-960.

[63] Yao Y, Luo Z, Li S, et al. MVSNet: Depth Inference for Unstructured Multi-view Stereo[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part VIII : Vol 11212. 2018 : 785-801.

[64] Yao Y, Luo Z, Li S, et al. Recurrent MVSNet for High-Resolution Multi-View Stereo Depth Inference[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 5525-5534.

[65] Chen R, Han S, Xu J, et al. Point-Based Multi-View Stereo Network[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 1538-1547.

[66] Gu X, Fan Z, Zhu S, et al. Cascade Cost Volume for High-Resolution Multi-View Stereo and Stereo Matching[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 2492-2501.

[67] Snavely N, Seitz S M, Szeliski R. Photo tourism: exploring photo collections in 3D[J]. ACM Transactions on Graphics, 2006, 25(3) : 835-846.

[68] Agarwal S, Snavely N, Simon I, et al. Building Rome in a day[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2009 : 72-79.

[69] Sweeney C, Fragoso V, Höllerer T, et al. Large Scale SfM with the Distributed Camera Model[C] // Proceedings of the Conference on 3D Vision (3DV). 2016 : 230-238.

[70] Hartley R I, Sturm P F. Triangulation[J]. Computer Vision Image Understanding, 1997, 68(2) : 146-157.

[71] Zach C. Robust Bundle Adjustment Revisited[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part V : Vol 8693. 2014 : 772-787.

[72] Crandall D J, Owens A, Snavely N, et al. Discrete-continuous optimization for large-scale structure from motion[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2011 : 3001-3008.

[73] Sweeney C, Sattler T, Höllerer T, et al. Optimizing the Viewing Graph for Structure-from-Motion[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2015 : 801-809.

[74] Farenzena M, Fusiello A, Gherardi R. Structure-and-motion pipeline on a hierarchical cluster tree[C] // Proceedings of the International Conference on Computer Vision Workshops (ICCVW). 2009 : 1489-1496.

[75] Choy C B, Xu D, Gwak J, et al. 3D-R2N2: A Unified Approach for Single and Multi-view 3D Object Reconstruction[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part VIII : Vol 9912. 2016 : 628-644.

[76] Kar A, Häne C, Malik J. Learning a Multi-View Stereo Machine[C] // Proceedings of the Conference on Neural Information Processing Systems (NIPS). 2017 : 365-376.

[77] Yang B, Wang S, Markham A, et al. Robust Attentional Aggregation of Deep Feature Sets for Multi-view 3D Reconstruction[J]. International Journal of Computer Vision, 2020, 128(1) : 53-73.

[78] Wen C, Zhang Y, Li Z, et al. Pixel2Mesh++: Multi-View 3D Mesh Generation via Deformation[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 1042-1051.

[79] Lin C, Wang O, Russell B C, et al. Photometric Mesh Optimization for Video-Aligned 3D Object Reconstruction[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 969-978.

[80] Brostow G J, Shotton J, Fauqueur J, et al. Segmentation and Recognition Using Structure from Motion Point Clouds[C] // Lecture Notes in Computer Science, Vol 5302 : Proceedings of the European Conference on Computer Vision (ECCV), Part I. 2008 : 44-57.

[81] Boulch A, de La Gorce M, Marlet R. Piecewise-Planar 3D Reconstruction with Edge and Corner Regularization[J]. Computer Graphics Forum, 2014, 33(5) : 55-64.

[82] Hedau V, Hoiem D, Forsyth D A. Recovering the spatial layout of cluttered rooms[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2009 : 1849-1856.

[83] Lee D C, Gupta A, Hebert M, et al. Estimating Spatial Layout of Rooms using Volumetric Reasoning about Objects and Surfaces[C] // Proceedings of the Conference on Neural Information Processing Systems (NIPS). 2010 : 1288-1296.

[84] Pero L D, Bowdish J, Fried D, et al. Bayesian geometric modeling of indoor scenes[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2012 : 2719-2726.

[85] Schwing A G, Fidler S, Pollefeys M, et al. Box in the Box: Joint 3D Layout and Object Reasoning from Single Images[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2013 : 353-360.

[86] Zhang Y, Song S, Tan P, et al. PanoContext: A Whole-Room 3D Context Model for Panoramic Scene Understanding[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part VI : Vol 8694. 2014 : 668-686.

[87] Xu J, Stenger B, Kerola T, et al. Pano2CAD: Room Layout from a Single Panorama Image[C] // Proceedings of the Winter Conference on Applications of Computer Vision (WACV). 2017 : 354-362.

[88] Pero L D, Bowdish J, Kermgard B, et al. Understanding Bayesian Rooms Using Composite 3D Object Models[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2013 : 153-160.

[89] Bao S Y, Furlan A, Li F, et al. Understanding the 3D layout of a cluttered room from multiple images[C] // Proceedings of the Winter Conference on Applications of Computer Vision (WACV). 2014 : 690-697.

[90] Yang Y, Jin S, Liu R, et al. Automatic 3D Indoor Scene Modeling From Single Panorama[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 3926-3934.

[91] Hou J, Dai A, Nießner M. 3D-SIS: 3D Semantic Instance Segmentation of RGB-D Scans[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 4421-4430.

[92] Shao T, Xu W, Zhou K, et al. An interactive approach to semantic modeling of indoor scenes with an RGBD camera[J]. ACM Transactions on Graphics, 2012, 31(6) : 136:1-136:11.

[93] Nan L, Xie K, Sharf A. A *search-classify* approach for cluttered indoor scene understanding[J]. ACM Transactions on Graphics, 2012, 31(6) : 137:1-137:10.

[94] Satkin S, Rashid M, Lin J, et al. 3DNN: 3D Nearest Neighbor[J]. International Journal of Computer Vision, 2015, 111(1) : 69-97.

[95] Kim Y M, Mitra N J, Yan D, et al. Acquiring 3D indoor environments with variability and repetition[J]. ACM Transactions on Graphics, 2012, 31(6) : 138:1-138:11.

[96] Shen C, Fu H, Chen K, et al. Structure recovery by part assembly[J]. ACM Transactions on Graphics, 2012, 31(6) : 180:1-180:11.

[97] Kundu A, Li Y, Rehg J M. 3D-RCNN: Instance-Level 3D Object Reconstruction via Render-and-Compare[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 3559-3568.

[98] Engelmann F, Rematas K, Leibe B, et al. From Points to Multi-Object 3D Reconstruction[J]. arXiv, 2020, 2012.11575.

[99] Firman M, Aodha O M, Julier S J, et al. Structured Prediction of Unobserved Voxels from a Single Depth Image[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2016 : 5431-5440.

[100] Gkioxari G, Johnson J, Malik J. Mesh R-CNN[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 9784-9794.

[101] He K, Gkioxari G, Dollár P, et al. Mask R-CNN[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2017 : 2980-2988.

[102] Nie Y, Han X, Guo S, et al. Total3DUnderstanding: Joint Layout, Object Pose and Mesh Reconstruction for Indoor Scenes From a Single Image[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 52-61.

[103] Rünz M, Li K, Tang M, et al. FroDO: From Detections to 3D Objects[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 14708-14717.

[104] Popov S, Bauszat P, Ferrari V. CoReNet: Coherent 3D Scene Reconstruction from a Single RGB Image[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part II : Vol 12347. 2020 : 366-383.

[105] Silberman N, Hoiem D, Kohli P, et al. Indoor Segmentation and Support Inference from RGBD Images[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part V : Vol 7576. 2012 : 746-760.

[106] Geiger A, Lenz P, Urtasun R. Are we ready for autonomous driving? The KITTI vision benchmark suite[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2012 : 3354-3361.

[107] Xiao J, Owens A, Torralba A. SUN3D: A Database of Big Spaces Reconstructed Using SfM and Object Labels[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2013 : 1625-1632.

[108] Xiang Y, Mottaghi R, Savarese S. Beyond PASCAL: A benchmark for 3D object detection in the wild[C] // Proceedings of the Winter Conference on Applications of Computer Vision (WACV). 2014 : 75-82.

[109] Chang A X, Funkhouser T A, Guibas L J, et al. ShapeNet: An Information-Rich 3D Model Repository[J]. arXiv, 2015, 1512.03012.

[110] Song S, Lichtenberg S P, Xiao J. SUN RGB-D: A RGB-D scene understanding benchmark suite[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2015 : 567-576.

[111] Xiang Y, Kim W, Chen W, et al. ObjectNet3D: A Large Scale Database for 3D Object Recognition[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part VIII : Vol 9912. 2016 : 160-176.

[112] Song S, Yu F, Zeng A, et al. Semantic Scene Completion from a Single Depth Image[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 190-198.

[113] McCormac J, Handa A, Leutenegger S, et al. SceneNet RGB-D: Can 5M Synthetic Images Beat Generic ImageNet Pre-training on Indoor Segmentation?[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2017 : 2697-2706.

[114] Armeni I, Sax S, Zamir A R, et al. Joint 2D-3D-Semantic Data for Indoor Scene Understanding[J]. arXiv, 2017, 1702.01105.

[115] Dai A, Chang A X, Savva M, et al. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 2432-2443.

[116] Chang A X, Dai A, Funkhouser T A, et al. Matterport3D: Learning from RGB-D Data in Indoor Environments[C] // Proceedings of the International Conference on 3D Vision (3DV). 2017 : 667-676.

[117] Sun X, Wu J, Zhang X, et al. Pix3D: Dataset and Methods for Single-Image 3D Shape Modeling[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 2974-2983.

[118] Li W, Saeedi S, McCormac J, et al. InteriorNet: Mega-scale Multi-sensor Photo-realistic Indoor Scenes Dataset[C] // Proceedings of the British Machine Vision Conference (BMVC). 2018 : 77.

[119] Armeni I, He Z, Zamir A R, et al. 3D Scene Graph: A Structure for Unified Semantics, 3D Space, and Camera[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 5663-5672.

[120] Zheng J, Zhang J, Li J, et al. Structured3D: A Large Photo-Realistic Dataset for Structured 3D Modeling[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part IX : Vol 12354. 2020 : 519-535.

[121] Fu H, Cai B, Gao L, et al. 3D-FRONT: 3D Furnished Rooms with layOuts and semaNTics[J]. arXiv, 2020, 2011.09127.

[122] Petersen F, Bermano A H, Deussen O, et al. Pix2Vex: Image-to-Geometry Reconstruction using a Smooth Differentiable Renderer[J]. arXiv, 2019, 1903.11149.

[123] Yang S, Xu M, Xie H, et al. Single-View 3D Object Reconstruction from Shape Priors in Memory[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2021 : 3152-3161.

[124] Fischler M A, Bolles R C. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography[J]. Communications of the ACM, 1981, 24(6) : 381-395.

[125] Ronneberger O, Fischer P, Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation[C] // Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI), Part III : Vol 9351. 2015 : 234-241.

[126] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition[C] // Proceedings of the International Conference on Learning Representations (ICLR). 2015.

[127] Tulsiani S, Zhou T, Efros A A, et al. Multi-view Supervision for Single-View Reconstruction via Differentiable Ray Consistency[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 209-217.

[128] Paszke A, Gross S, Massa F, et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library[C] // Proceedings of the Conference on Neural Information Processing Systems (NeurIPS). 2019 : 8024-8035.

[129] Kingma D P, Ba J. Adam: A Method for Stochastic Optimization[C] // Proceedings of the International Conference on Learning Representations (ICLR). 2015.

[130] Su H, Qi C R, Li Y, et al. Render for CNN: Viewpoint Estimation in Images Using CNNs Trained with Rendered 3D Model Views[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2015 : 2686-2694.

[131] Ilg E, Saikia T, Keuper M, et al. Occlusions, Motion and Depth Boundaries with a Generic Network for Disparity, Optical Flow or Scene Flow Estimation[C] // Lecture Notes in Computer Science, Vol 11216 : Proceedings of the European Conference on Computer Vision (ECCV), Part XII. 2018 : 626-643.

[132] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2016 : 770-778.

[133] Iandola F N, Moskewicz M W, Ashraf K, et al. SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size[C] // Proceedings of the International Conference on Learning Representations (ICLR). 2017.

[134] Kendall A, Martirosyan H, Dasgupta S, et al. End-to-End Learning of Geometry and Context for Deep Stereo Regression[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2017 : 66-75.

[135] Richter S R, Roth S. Matryoshka Networks: Predicting 3D Geometry via Nested Shape Layers[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 1936-1944.

[136] Groueix T, Fisher M, Kim V G, et al. A Papier-Mâché Approach to Learning 3D Surface Generation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 216-224.

[137] Xiao J, Hays J, Ehinger K A, et al. SUN database: Large-scale scene recognition from abbey to zoo[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2010 : 3485-3492.

[138] Tatarchenko M, Richter S R, Ranftl R, et al. What Do Single-View 3D Reconstruction Networks Learn?[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 3405-3414.

[139] Zhang F, Prisacariu V A, Yang R, et al. GA-Net: Guided Aggregation Net for End-to-end Stereo Matching[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 185-194.

[140] Xu H, Zhang J. AANet: Adaptive Aggregation Network for Efficient Stereo Matching[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 1956-1965.

[141] Dai A, Qi C R, Nießner M. Shape Completion Using 3D-Encoder-Predictor CNNs and Shape Synthesis[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 6545-6554.

[142] Han X, Li Z, Huang H, et al. High-Resolution Shape Completion Using Deep Neural Networks for Global Structure and Local Geometry Inference[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2017 : 85-93.

[143] Sharma A, Grau O, Fritz M. VConv-DAE: Deep Volumetric Shape Learning Without Object Labels[C] // Lecture Notes in Computer Science, Vol 9915 : Proceedings of the European Conference on Computer Vision (ECCV), Part III. 2016 : 236-250.

[144] Stutz D, Geiger A. Learning 3D Shape Completion From Laser Scan Data With Weak Supervision[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 1955-1964.

[145] Nguyen D T, Hua B, Tran M, et al. A Field Model for Repairing 3D Shapes[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2016 : 5676-5684.

[146] Varley J, DeChant C, Richardson A, et al. Shape completion enabled robotic grasping[C] // Proceedings of the International Conference on Intelligent Robots and Systems (IROS). 2017 : 2442-2447.

[147] Liu Z, Tang H, Lin Y, et al. Point-Voxel CNN for Efficient 3D Deep Learning[C] // Proceedings of the Conference on Neural Information Processing Systems (NeurIPS). 2019 : 963-973.

[148] Mandikal P, Radhakrishnan V B. Dense 3D Point Cloud Reconstruction Using a Deep Pyramid Network[C] // Proceedings of the Winter Conference on Applications of Computer Vision (WACV). 2019 : 1052-1060.

[149] Wang Y, Sun Y, Liu Z, et al. Dynamic Graph CNN for Learning on Point Clouds[J]. ACM Transactions on Graphics, 2019, 38(5) : 146:1-146:12.

[150] Wang K, Chen K, Jia K. Deep Cascade Generation on Point Sets[C] // Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI). 2019 : 3726-3732.

[151] Thomas H, Qi C R, Deschaud J, et al. KPConv: Flexible and Deformable Convolution for Point Clouds[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 6410-6419.

[152] Su H, Jampani V, Sun D, et al. SPLATNet: Sparse Lattice Networks for Point Cloud Processing[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 2530-2539.

[153] Mao J, Wang X, Li H. Interpolated Convolutional Networks for 3D Point Cloud Understanding[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 1578-1587.

[154] Qi C R, Su H, Mo K, et al. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 77-85.

[155] Achlioptas P, Diamanti O, Mitliagkas I, et al. Learning Representations and Generative Models for 3D Point Clouds[C] // Proceedings of Machine Learning Research, Vol 80 : Proceedings of the International Conference on Machine Learning (ICML). 2018 : 40-49.

[156] Lin H, Xiao Z, Tan Y, et al. Justlookup: One Millisecond Deep Feature Extraction for Point Clouds By Lookup Tables[C] // Proceedings of the International Conference on Multimedia and Expo (ICME). 2019 : 326-331.

[157] Qi C R, Yi L, Su H, et al. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space[C] // Proceedings of the Conference on Neural Information Processing Systems (NIPS). 2017 : 5099-5108.

[158] Liu M, Sheng L, Yang S, et al. Morphing and Sampling Network for Dense Point Cloud Completion[C] // Proceedings of the Conference on Artificial Intelligence (AAAI). 2020 : 11596-11603.

[159] Zhang K, Hao M, Wang J, et al. Linked Dynamic Graph CNN: Learning on Point Cloud via Linking Hierarchical Features[J]. arXiv, 2019, 1904.10014.

[160] Hassani K, Haley M. Unsupervised Multi-Task Feature Learning on Point Clouds[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 8159-8170.

[161] Li D, Shao T, Wu H, et al. Shape Completion from a Single RGBD Image[J]. IEEE Transactions on Visualization and Computer Graphics, 2017, 23(7) : 1809-1822.

[162] Wang Z, Lu F. VoxSegNet: Volumetric CNNs for Semantic Part Segmentation of 3D Shapes[J]. IEEE Transactions on Visualization and Computer Graphics, 2020, 26(9) : 2919-2930.

[163] Hua B, Tran M, Yeung S. Pointwise Convolutional Neural Networks[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 984-993.

[164] Lei H, Akhtar N, Mian A. Octree Guided CNN With Spherical Kernels for 3D Point Clouds[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 9631-9640.

[165] Lan S, Yu R, Yu G, et al. Modeling Local Geometric Structure of 3D Point Clouds Using Geo-CNN[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 998-1008.

[166] Li Y, Bu R, Sun M, et al. PointCNN: Convolution On X-Transformed Points[C] // Proceedings of the Conference on Neural Information Processing Systems (NeurIPS). 2018 : 828-838.

[167] Xu Y, Fan T, Xu M, et al. SpiderCNN: Deep Learning on Point Sets with Parameterized Convolutional Filters[C] // Lecture Notes in Computer Science, Vol 11212 : Proceedings of the European Conference on Computer Vision (ECCV), Part VIII. 2018 : 90-105.

[168] Liu Y, Fan B, Xiang S, et al. Relation-Shape Convolutional Neural Network for Point Cloud Analysis[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 8895-8904.

[169] Liu Y, Fan B, Meng G, et al. DensePoint: Learning Densely Contextual Representation for Efficient Point Cloud Processing[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 5238-5247.

[170] Wu W, Qi Z, Li F. PointConv: Deep Convolutional Networks on 3D Point Clouds[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 9621-9630.

[171] Hermosilla P, Ritschel T, Vázquez P, et al. Monte Carlo convolution for learning on non-uniformly sampled point clouds[J]. ACM Transactions on Graphics, 2018, 37(6) : 235:1-235:12.

[172] Esser S K, McKinstry J L, Bablani D, et al. Learned Step Size Quantization[C] // Proceedings of the International Conference on Learning Representations (ICLR). 2020.

[173] Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: The KITTI dataset[J]. The International Journal of Robotics Research, 2013, 32(11) : 1231-1237.

[174] Yang Y, Feng C, Shen Y, et al. FoldingNet: Point Cloud Auto-Encoder via Deep Grid Deformation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 206-215.

[175] Wang Y, Tan D J, Navab N, et al. SoftPoolNet: Shape Descriptor for Point Cloud Completion and Classification[C] // Proceedings of the European Conference on Computer Vision (ECCV), Part III : Vol 12348. 2020 : 70-85.

[176] Li R, Li X, Fu C, et al. PU-GAN: A Point Cloud Upsampling Adversarial Network[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 7202-7211.

[177] Fuentes-Pacheco J, Ascencio J R, Rendón-Mancha J M. Visual simultaneous localization and mapping: a survey[J]. Artificial Intelligence Review, 2015, 43(1) : 55-81.

[178] Huang P, Matzen K, Kopf J, et al. DeepMVS: Learning Multi-View Stereopsis[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 2821-2830.

[179] Paschalidou D, Ulusoy A O, Schmitt C, et al. RayNet: Learning Volumetric 3D Reconstruction With Ray Potentials[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 3897-3906.

[180] Vinyals O, Bengio S, Kudlur M. Order Matters: Sequence to sequence for sets[C] // Proceedings of the International Conference on Learning Representations (ICLR). 2016.

[181] Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks[C] // JMLR Workshop and Conference Proceedings, Vol 28 : Proceedings of the International Conference on Machine Learning (ICML). 2013 : 1310-1318.

[182] Hwang K, Sung W. Single stream parallelization of generalized LSTM-like RNNs on a GPU[C] // Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2015 : 1047-1051.

[183] Cadena C, Carlone L, Carrillo H, et al. Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age[J]. IEEE Transactions on Robotics, 2016, 32(6) : 1309-1332.

[184] Lorensen W E, Cline H E. Marching cubes: A high resolution 3D surface construction algorithm[C] // Proceedings of the Conference on Computer Graphics and Interactive Techniques (SIGGRAPH). 1987 : 163-169.

[185] Hu Y, Huang J, Schwing A G. MaskRNN: Instance Level Video Object Segmentation[C] // Proceedings of the Conference on Neural Information Processing Systems (NIPS). 2017 : 325-334.

[186] Luiten J, Voigtlaender P, Leibe B. PReMVOS: Proposal-Generation, Refinement and Merging for Video Object Segmentation[C] // Lecture Notes in Computer Science, Vol 11364 : Proceedings of the European Conference on Computer Vision (ECCV), Part IV. 2018 : 565-580.

[187] Perazzi F, Khoreva A, Benenson R, et al. Learning Video Object Segmentation from Static Images[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 3491-3500.

[188] Chen Y, Pont-Tuset J, Montes A, et al. Blazingly Fast Video Object Segmentation With Pixel-Wise Metric Learning[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 1189-1198.

[189] Lu X, Wang W, Danelljan M, et al. Video Object Segmentation with Episodic Graph Memory Networks[C] // Lecture Notes in Computer Science, Vol 12348 : Proceedings of the European Conference on Computer Vision (ECCV), Part III. 2020 : 661-679.

[190] Oh S W, Lee J, Xu N, et al. Video Object Segmentation Using Space-Time Memory Networks[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 9225-9234.

[191] Seong H, Hyun J, Kim E. Kernelized Memory Network for Video Object Segmentation[C] // Lecture Notes in Computer Science, Vol 12367 : Proceedings of the European Conference on Computer Vision (ECCV), Part XXII. 2020 : 629-645.

[192] Voigtlaender P, Chai Y, Schroff F, et al. FEELVOS: Fast End-To-End Embedding Learning for Video Object Segmentation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 9481-9490.

[193] Yang Z, Wei Y, Yang Y. Collaborative Video Object Segmentation by Foreground-Background Integration[C] // Lecture Notes in Computer Science, Vol 12350 : Proceedings of the European Conference on Computer Vision (ECCV), Part V. 2020 : 332-348.

[194] Tsai Y, Yang M, Black M J. Video Segmentation via Object Flow[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2016 : 3899-3908.

[195] Cheng J, Tsai Y, Wang S, et al. SegFlow: Joint Learning for Video Object Segmentation and Optical Flow[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2017 : 686-695.

[196] Xu Y, Fu T, Yang H, et al. Dynamic Video Segmentation Network[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 6556-6565.

[197] Bao L, Wu B, Liu W. CNN in MRF: Video Object Segmentation via Inference in a CNN-Based Higher-Order Spatio-Temporal MRF[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 5977-5986.

[198] Li X, Loy C C. Video Object Segmentation with Joint Re-identification and Attention-Aware Mask Propagation[C] // Lecture Notes in Computer Science, Vol 11207 : Proceedings of the European Conference on Computer Vision (ECCV), Part III. 2018 : 93-110.

[199] Cheng J, Tsai Y, Hung W, et al. Fast and Accurate Online Video Object Segmentation via Tracking Parts[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 7415-7424.

[200] Khoreva A, Benenson R, Ilg E, et al. Lucid Data Dreaming for Video Object Segmentation[J]. International Journal of Computer Vision, 2019, 127(9) : 1175-1197.

[201] Chen X, Li Z, Yuan Y, et al. State-Aware Tracker for Real-Time Video Object Segmentation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2020 : 9381-9390.

[202] Oh S W, Lee J, Xu N, et al. Fast User-Guided Video Object Segmentation by Interaction-And-Propagation Networks[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 5247-5256.

[203] Caelles S, Maninis K, Pont-Tuset J, et al. One-Shot Video Object Segmentation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 5320-5329.

[204] Oh S W, Lee J, Sunkavalli K, et al. Fast Video Object Segmentation by Reference-Guided Mask Propagation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 7376-7385.

[205] Hu Y, Huang J, Schwing A G. VideoMatch: Matching Based Video Object Segmentation[C] // Lecture Notes in Computer Science, Vol 11212 : Proceedings of the European Conference on Computer Vision (ECCV), Part VIII. 2018 : 56-73.

[206] Wang X, Girshick R B, Gupta A, et al. Non-Local Neural Networks[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 7794-7803.

[207] Zhu Z, Xu M, Bai S, et al. Asymmetric Non-Local Neural Networks for Semantic Segmentation[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 593-602.

[208] Dosovitskiy A, Fischer P, Ilg E, et al. FlowNet: Learning Optical Flow with Convolutional Networks[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2015 : 2758-2766.

[209] Schuster R, Wasenmüller O, Unger C, et al. SDC - Stacked Dilated Convolution: A Unified Descriptor Network for Dense Matching Tasks[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 2556-2565.

[210] Sun D, Yang X, Liu M, et al. PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 8934-8943.

[211] Xu J, Ranftl R, Koltun V. Accurate Optical Flow via Direct Cost Volume Processing[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 5807-5815.

[212] Ilg E, Mayer N, Saikia T, et al. FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2017 : 1647-1655.

[213] Perazzi F, Pont-Tuset J, McWilliams B, et al. A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2016 : 724-732.

[214] Pont-Tuset J, Perazzi F, Caelles S, et al. The 2017 DAVIS Challenge on Video Object Segmentation[J]. arXiv, 2017, 1704.00675.

[215] Xu N, Yang L, Fan Y, et al. YouTube-VOS: Sequence-to-Sequence Video Object Segmentation[C] // Lecture Notes in Computer Science, Vol 11209 : Proceedings of the European Conference on Computer Vision (ECCV), Part V. 2018 : 603-619.

[216] Martin D R, Fowlkes C C, Malik J. Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues[J]. IEEE Transactions on Pattern Analysis and Machine Intellegence, 2004, 26(5) : 530-549.

[217] Voigtlaender P, Leibe B. Online Adaptation of Convolutional Neural Networks for Video Object Segmentation[C] // Proceedings of the British Machine Vision Conference (BMVC). 2017.

[218] Ci H, Wang C, Wang Y. Video Object Segmentation by Learning Location-Sensitive Embeddings[C] // Lecture Notes in Computer Science, Vol 11215 : Proceedings of the European Conference on Computer Vision (ECCV), Part XI. 2018 : 524-539.

[219] Johnander J, Danelljan M, Brissman E, et al. A Generative Appearance Model for End-To-End Video Object Segmentation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2019 : 8953-8962.

[220] Yang L, Wang Y, Xiong X, et al. Efficient Video Object Segmentation via Network Modulation[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 6499-6507.

[221] Voigtlaender P, Luiten J, Leibe B. BoLTVOS: Box-Level Tracking for Video Object Segmentation[J]. arXiv, 2019, 1904.04552.

[222] Teed Z, Deng J. RAFT: Recurrent All-Pairs Field Transforms for Optical Flow[C] // Lecture Notes in Computer Science, Vol 12347 : Proceedings of the European Conference on Computer Vision (ECCV), Part II. 2020 : 402-419.

[223] Huang S, Qi S, Xiao Y, et al. Cooperative Holistic Scene Understanding: Unifying 3D Object, Layout, and Camera Pose Estimation[C] // Proceedings of the Conference on Neural Information Processing Systems (NeurIPS). 2018 : 206-217.

[224] Kulkarni N, Misra I, Tulsiani S, et al. 3D-RelNet: Joint Object and Relational Network for 3D Prediction[C] // Proceedings of the International Conference on Computer Vision (ICCV). 2019 : 2212-2221.

[225] Hu H, Gu J, Zhang Z, et al. Relation Networks for Object Detection[C] // Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR). 2018 : 3588-3597.

# 攻读博士学位期间发表的论文及其他成果

## （一）已发表的学术论文

[1] **Haozhe Xie**, Hongxun Yao, Xiaoshuai Sun, Shangchen Zhou, Shengping Zhang. Pix2Vox: Context-aware 3D Reconstruction from Single and Multi-view Images[C]. Proceedings of the International Conference on Computer Vision **(ICCV)**, 2019: 2690-2698.（EI 收录号：20201208326813，第 2 章，CCF-A 类）

[2] **Haozhe Xie**, Hongxun Yao, Shangchen Zhou, Jiageng Mao, Shengping Zhang, Wenxiu Sun. GRNet: Gridding Residual Network for Dense Point Cloud Completion [C]. Proceedings of the European Conference on Computer Vision **(ECCV)**(12534), 2020: 365-381.（EI 收录号：20204909584740，第 3 章，CCF-B 类）

[3] **Haozhe Xie**, Hongxun Yao, Shengping Zhang, Shangchen Zhou, Wenxiu Sun. Pix2Vox++: Multi-scale Context-aware 3D Object Reconstruction from Single and Multi-view Images[J]. International Journal of Computer Vision **(IJCV)**, 128(12): 2919-2935, 2020.（SCI 收录号：000548483600001，第 4 章，CCF-A 类）

[4] **Haozhe Xie**, Hongxun Yao, Shangchen Zhou, Shengping Zhang, Wenxiu Sun. Efficient Regional Memory Network for Video Object Segmentation [C]. Proceedings of the Conference on Computer Vision and Pattern Recognition **(CVPR)**, 2021: 1286-1295.（第 5 章，CCF-A 类）

## （二）在投的学术论文

[1] **Haozhe Xie**, Xiaojun Tong, Hongxun Yao, Shangchen Zhou, Shengping Zhang, Wenxiu Sun. Toward 3D Object Reconstruction from Stereo Images [J]. Neurocomputing.（第 2 章）

## （三）与他人合作发表的代表性论文

[1] Shangchen Zhou, Jiawei Zhang, Wangmeng Zuo, **Haozhe Xie**, Jinshan Pan, Jimmy S. J. Ren. DAVANet: Stereo Deblurring with View Aggregation [C]. Proceedings of the International Conference on Computer Vision and Pattern Recognition **(CVPR)**, 2019: 10996-11005.（EI 收录号：202000508113886，CCF-A 类）

[2] Shangchen Zhou, Jiawei Zhang, Jinshan Pan, **Haozhe Xie**, Wangmeng Zuo, Jimmy S.

J. Ren. Spatio-Temporal Filter Adaptive Network for Video Deblurring [C]. Proceedings of the International Conference on Computer Vision **(ICCV)**, 2019: 2482-2491.（EI 收录号：20201208327464，CCF-A 类）

[3] Shuo Yang, Min Xu, **Haozhe Xie**, Stuart Perry, Jiahao Xia. Single-View 3D Object Reconstruction from Shape Priors in Memory. Proceedings of the Conference on Computer Vision and Pattern Recognition **(CVPR)**, 2021: 3152-3161.（CCF-A 类）

**（四）参与的科研项目及获奖情况**

[1] 姚鸿勋等。视觉语义的 Web 统计模型及理解深化，国家自然科学基金面上项目。课题编号：61472103。

[2] 姚鸿勋等。图像情感元素计算，国家自然科学基金面上项目。课题编号：61772158。

[3] 佟晓筠等。基于复合混沌序列轻量级密码关键技术研究，山东省自然科学基金面上项目。课题编号：ZR2019MF054。

[4] 姚鸿勋等。智能机器人的三维环境重建与目标感知，机器人技术与系统国家重点实验室（哈尔滨工业大学）自主研究课题。课题编号：SKLRS202002D。

# 哈尔滨工业大学学位论文原创性声明和使用权限

## 学位论文原创性声明

本人郑重声明：此处所提交的学位论文《多源多视的三维场景和物体重建》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：谢浩哲　　日期：2021 年 6 月 23 日

## 学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

（1）学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；（2）学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；（3）研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：谢浩哲　　日期：2021 年 6 月 23 日

导师签名：[illegible]　　日期：2021 年 6 月 23 日

# 致 谢

凡所过往，皆为序章。

回首过去四年，转瞬即逝却又历历在目。这一路走来，有太多想说的话，也有太多想感谢的人。恩师、同窗、亲友、爱人，是你们帮助和关怀使我一路走来，也正因为有你们的陪伴，我的博士生涯才充满着绚丽的色彩。

衷心感谢我的导师佟晓筠教授和姚鸿勋教授。感谢佟老师和姚老师一直以来对我的栽培和指导，以及在学习和生活上对我帮助、支持和鼓励。在我攻读博士学位的这些年里，两位老师为我付出了大量的时间和心血，给我提供了非常好的科研条件和宝贵的交流机会。她们对待工作勤勤恳恳、一丝不苟的敬业精神，和对待生活积极乐观的人生态度对我的学术生涯和人生选择上都带来了非常积极的影响，这将是我一辈子的宝贵财富。同为杭州人的姚老师，让我在遥远的哈尔滨感受到了家的温暖。

感谢商汤科技的孙文秀博士。我曾于 2019 年 3 月至 2020 年 11 月在商汤科技移动智能事业群的 AI 画质部实习。孙文秀博士平易近人，在百忙之中指导我的科研工作。在与她的交流中，她总是热情地解答我的疑问并提供指导，给予了我非常大的启发。她擅长制定严格的工作计划、对工作和科研充满了激情，这些都影响着我自己在未来的工作中的习惯和态度。在商汤科技实习的这 18 个月，我获得了几乎无限的算力资源，使得我的科研工作得以快速推进。

感谢实验室赵德斌教授、范晓鹏教授、刘岩副教授和刘绍辉副教授。感谢各位老师为我们创造了良好的实验室环境，营造了轻松、和谐、积极、严谨的科研氛围。感谢本文的责任专家左旺孟教授以及参加答辩的唐降龙教授、赵铁军教授、王宽全教授、姚鸿勋教授为本文提出的宝贵意见。

感谢亦师亦友的张盛平教授。在读博期间，盛平师兄对学术的专注和严谨深深感染着我。论文中的每一个用词他都反复斟酌，确保准确和到位。那些一起讨论的清晨，一起奋斗的黑夜，依然历历在目。学术上他严格要求，生活中他平易近人。他的谆谆教诲指引我不断前行。

感谢实验室的同窗们。我很庆幸加入 VILab 这个大家庭。导师的谆谆教诲，师兄师姐的无私帮助，师弟师妹的朝夕陪伴使我不断地进步和成长。正是你们让 VILab 成为一个温暖而又有活力的集体，在实验室这么多年的科研和生活中，有你们的陪伴多了许多欢乐和温暖，也给我的博士生活留下了许多美好的回忆。

感谢我的挚友周尚辰和刘环宇。在此，不得不感叹缘分妙不可言！如果我当年不买烤冷面，就不会认识尚辰，更不会从事计算机视觉的研究。与尚辰在哈尔滨和深圳共同生活的时光留下了太多有趣的回忆。作为我读博期间所有文章的合作作者，我依然清楚地记得我们一起探讨科研想法（和扯犊子）的无数个白天和夜晚。在我攻读硕士学位期间，非常偶然结识了环宇。环宇和他的导师李君宝教授在读博期间给了我非常多的帮助和支持。

感谢我的父母及家人，在我漫长的求学道路上离不开他们的关心和支持。家人对我学业上的支持、精神上的鼓励、生活上的关心，给予我克服困难的勇气和不断进取的力量。

特别感谢我的爱妻郑晓路。在攻读博士的这些年，我经历过低谷，也有过“高光”时刻。是你，给了我无私的爱，也承担了太多本该我扛起的责任。是你，和我走南闯北，一直陪在我的身边。是你，在我低谷的时候让我不要放弃，在我取得成就的时候让我继续加油。

道阻且长，行则将至。

# 个人简历

谢浩哲，男，1993 年 3 月生，浙江杭州人。主要研究领域为计算机视觉、三维重建和视频物体分割。在攻读博士期间，以第一作者身份发表国际期刊论文 1 篇，国际会议论文 3 篇，这些文章均被计算机视觉的顶级期刊和会议收录，包括 CVPR、ICCV、ECCV 和 IJCV。

**教育经历**

2017 年 9 月—至今

哈尔滨工业大学，计算学部，计算机科学与技术专业，在读博士

2015 年 9 月—2017 年 7 月

哈尔滨工业大学，计算机科学与技术学院，计算机技术专业，工程硕士

2011 年 9 月—2015 年 7 月

合肥工业大学，软件学院，工学学士

**工作经历**

2019 年 3 月—2020 年 11 月

深圳市商汤科技有限公司，见习研究员

**学术职务**

会议审稿人 CVPR、ICCV、ICLR、ICML、NeurIPS、WACV 等

期刊审稿人 IJCV、TIP、TMM 等

**获奖情况**

2021 年　　哈尔滨工业大学优秀博士毕业生

2020 年　　博士生国家奖学金

2020 年　　商汤科技优秀实习生

2017 年　　哈尔滨工业大学优秀硕士毕业生

2016 年　　硕士生国家奖学金

2015 年　　合肥工业大学优秀本科毕业生

2013 年　　联发科技奖学金