一种新的 6D 目标姿态估计的深度和颜色特征融合框架

Guangliang Zhou, Yi Yan, Deming Wang, Qijun Chen, Senior Member, IEEE

摘要—本文旨在利用 RGB-D 图像解决遮挡下物体的 6D 姿态估计问题。大多数现有的方法通常分别使用颜色和深度图像的信息进行预测，这限制了它们在存在遮挡的情况下的性能。相反，我们提出了一个有效融合颜色和深度信息的管线，并进行区域级姿态估计。我们的方法首先使用 CNN 提取颜色特征，然后将颜色特征结合到点云中获得融合特征。与现有方法不同的是，融合特征以点集的形式出现，而不是以特征映射的形式出现。我们进一步使用一个类似 PointNet++的网络来处理融合特征，获得几个区域级特征。每个区域级别的特征都可以用置信度预测一个姿态。具有最高置信度的姿势被选择为最终输出。实验表明，该方法在 LINEMOD 和遮挡 LINEMOD 数据集上都优于目前最先进的方法，表明该方法能够获得准确的姿态估计结果，对遮挡具有较好的鲁棒性。

关键词：目标姿态估计，颜色和深度特征融合，区域级特征

I. 介绍

近年来，图像中的 6D 目标姿态估计受到了广泛的关注，在机器人操作、自动驾驶、虚拟现实和增强现实等诸多应用中发挥着重要作用。最近的研究集中在解决无纹理目标的姿态估计和提高对背景杂波、前景遮挡、光照变化和传感器噪声的鲁棒性。相比于仅使用 RGB 的方法，低成本的 RGB-D 相机的出现使得更准确地推断无纹理对象的姿态成为可能。然而，现有的使用 RGB-D 图像的方法大多只使用深度图像来细化姿态。深度信息没有得到充分利用。因此，这些方法不能很好地处理遮挡情况下目标的姿态估计问题。我们认为，在遮挡下的姿态估计问题的关键是同时有效地利用颜色信息和深度信息。

传统的[1]-[4]方法在一定程度上解决了纹理对象的这一问题。该方法首先从 RGB 图像中检测出二维关键点，建立二维关键点和三维关键点的对应关系，然后通过求解 PnP[5]问题计算出姿态。然而，这些方法在无纹理对象上的性能较差，因为无纹理对象的表面无法提供足够的信息来提取二维关键点。基于模板的方法[6]-[13]的目的是执行无纹理对象的姿态估计。LINEMOD[8]就是这些方法的代表。通过将模板图像的颜色梯度和表面法向量与真实图像进行比较，可以找到与真实图像最相似的模板图像，从而得到目标的姿态。在没有遮挡的情况下，基于模板的方法可以快速、高精度地进行识别，但遮挡会显著降低识别性能。

近年来，随着深度学习在图像分类[14]、目标检测[15]等许多其他领域取得的巨大成功，在姿态估计领域出现了一些使用深度学习的方法。这些方法可以分为两类。一类是利用深度学习方法检测 2D 关键点[16]-[22]，解决了传统方法不适用于无纹理对象的问题。然而，这种方法仍然对遮挡敏感。另一类使用深度学习，使用 PoseCNN[23]和 Triplet 网络[24]等彩色图像直接回归物体的 6D 姿态。然而，回归的姿态通常是不准确的，所以昂贵的后处理步骤，如 ICP[25]是这些方法实现良好结果必不可少的。以上两种基于深度学习的方法都只使用彩色图像作为网络的输入，没有利用深度信息，也没有将深度信息与颜色信息结合起来。在三维手部姿态估计[26]、[27](手部关节的三维位置估计)的背景下，已经提出了一些利用点云深度信息的方法，如 Hand PointNet[28]和 Cascaded PointNet[29]。两者都是利用 PointNet-family 网络直接对三维点云进行姿态回归处理，取得了较好的效果。然而，这些方法试图消除在三维手部姿态估计过程中各种手部方向的影响，而手部方向正是我们关注的重点。因此，这些方法不适用于 6D 目标的姿态估计。最近，DenseFusion[30]试图融合颜色和深度信息来预测 6D 姿势。它使用不同的网络来处理颜色和深度信息，然后在每像素级别结合处理结果来执行像素级的姿态估计，实现了最先进的性能。然而，它从每个单点中提取了点云的几何特征。显然，用这种方式无法很好地表达三维几何特征。

在本文中，我们提出了一种端到端的深度学习方法，可以充分利用颜色和深度信息来执行区域级的姿态估计。该方法首先提取颜色特征，然后将其结合到点云中。然后，提取得到的点云的几个区域级特征来估计物体的 6D 姿态。这一想法源于现实生活中的两个事实:一是人类看到的物体是三维的，并具有颜色信息。另一种是，人们可以通过观察物体的局部来推断物体的姿势。受这两个事实的启发，我们将颜色信息组合成点云，然后发送到深层网络，使网络能够"看到"真实的三维场景。此外，我们将点云划分为几个区域，网络可以提取每个区域的特征。这样，我们可以得到结合颜色特征的区域级三维几何特征。然后进行区域级姿态估计，选取置信度最高的区域预测的姿态作为姿态估计结果。最后，利用姿态细分模块进一步提高了姿态估计的精度。该方案使模型能够充分利用局部外观和几何信息，这对于处理背景杂波和严重遮挡是必不可少的。我们在两个基准数据集 LINEMOD[8]和遮挡 LINEMOD[31]上评估了所提出的方法。我们的方法在两个数据集上都优于最先进的方法。

综上所述，我们的贡献如下所示:

- 提出了一种同时处理颜色信息和深度信息的方法。具体来说，我们将颜色信息组合成一个点云，然后利用神经网络对产生的点云进行处理，从而得到颜色和深度融合的特征。
- 我们将点云划分为几个区域，获取区域级特征，并利用这些特征进行区域级姿态估计来处理严重遮挡。
- 我们在 LINEMOD 和遮挡 LINEMOD 数据集上实现了最先进的 6D 姿态估计性能。

II. 相关工作

A. 基于模板的方法

模板匹配是基于模板的方法的一项关键技术，它以较小的像素步长对整个场景图像进行多个离散大小的滑动窗口扫描，并在模板数据集中寻找最佳匹配。每个模板都标记了一个 6D 姿势的对象。当找到最佳模板时，6D 对象的姿态就确定了。LINEMOD[8]是一种代表性的方法，将真实图像的颜色梯度和表面法线与模板图像的颜色梯度和表面法线进行匹配，从而获得 6D 姿态。Rios-Cabrera 等人[11]提出了一种扩展的方法，利用集群来区分模板。该方法对大量目标具有较好的识别能力，检测速度较快。Cai 等人的[12]和 Hodaˇn 等人的[13]通过级联式和散列编码投票方案优化匹配，提高了 6D 姿态估计的精度。然而，这些方法的缺点是遮挡会显著降低识别性能。

B. 基于对应的方法

基于对应的方法包括两个步骤:首先在二维图像中检测目标的关键点，然后根据二维关键点和三维关键点的对应关系，利用 PnP 算法得到 6D 目标的姿态。对于纹理丰富的物体，传统的特征描述符[1]-[4]可以在观察到的图像上找到二维关键点，得到满意的结果。然而，这些传统的描述符并不适用于纹理较弱或没有纹理的物体。为了解决这个问题，最近的著作[16]-[21]定义了一组语义关键点，并使用 cnn 作为关键点检测器。Tekin 等人使用 YOLO 架构来估算该物体的 9 个关键点。Hu 等人[19]提出了一种分割驱动的 6D 姿态估计框架，其中物体的每个可见部分以二维关键点位置的形式提供局部姿态预测。Peng 等人[20]提出了一个像素级投票网络来确定关键点位置。但是关键点检测容易受到遮挡的干扰，而像素级投票方法需要大量的训练数据才能达到较好的效果。为了处理遮挡，Park 等人的[21]利用了生成对抗训练的最新成果来精确恢复遮挡部分，仅使用 RGB 图像就实现了最先进的性能。

C. RGB-D 方法

经典的方法是从输入的 RGB-D 数据中提取手工特征，然后进行对应分组和假设验证[31]-[36]。然而，依赖于手工制作的特征和固定的匹配程序限制了它们在存在严重遮挡和光照变化的情况下的经验性能。较新的方法使用 cnn 直接预测物体的姿态[23], [24], [37]-[41]。

他们要么仅从颜色数据估计 6D 姿态，要么将深度图像融合为彩色图像[41]的额外通道，然后使用一个姿态细分步骤来利用深度信息。然而，位姿的细化耗时较长，难以满足实时应用的要求。这些方法仅利用彩色图像进行姿态估计。在预测步骤中没有有效地利用深度信息，没有将深度和颜色信息结合起来进行预测。DenseFusion[30]在这方面进行了尝试，使用不同的网络处理颜色和深度信息，然后在像素级将两者结合，进行像素级的姿态估计，实现了最先进的性能。

由于密集融合的成功，我们也结合颜色信息和深度信息来估计 6D 姿态。相比之下，我们首先使用 CNNs 对彩色图像进行初步的特征提取，然后将提取的颜色特征结合到点云中。然后，我们使用一个类似 PointNet++的网络来处理产生的点云，获得几个区域级特征，并利用这些特征进行区域级姿态估计。结果表明，我们的区域级特征融合方案优于 DenseFusion 的像素级融合方法。

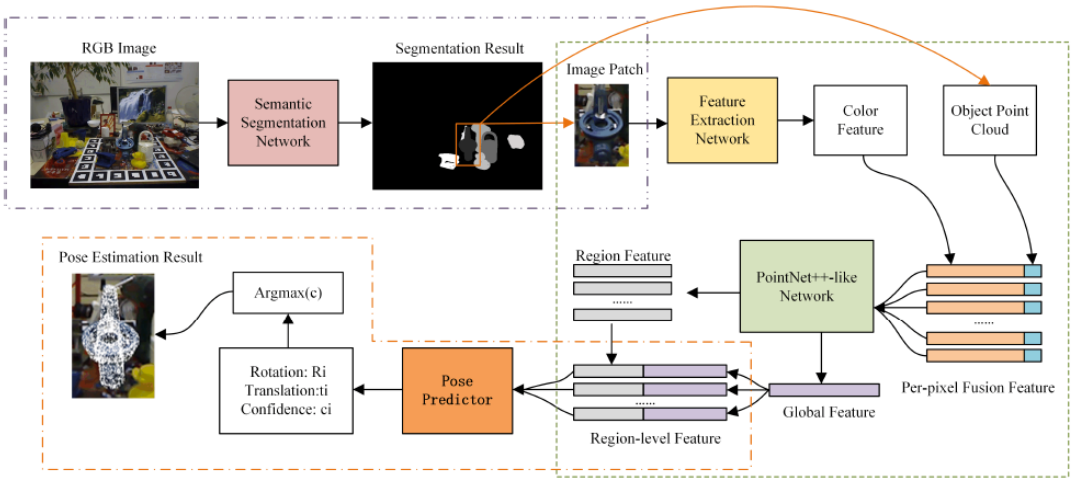

图 1. 我们模型的架构概述。我们的模型从彩色图像中生成目标分割掩模，获得图像补丁和目标点云。将图像块编码成彩色特征图，并在每个对应点与点云融合。从产生的点云中提取几个区域级别的特征，姿态预测器为每个区域特征生成一个姿态。选取置信度最高的姿态作为对象的 6D 姿态预测。(为了简单起见，这里不描述姿态优化模块)

III. 提出的方法

在本文中，我们提出了一种 6D 目标姿态估计方法。给定一幅图像，6D 姿态估计的任务是估计三维空间中物体的方向和平移。具体来说，6D 位姿由刚性变换表示[R|t]，其中 R 表示 3D 旋转，t 表示 3D 平移。我们认为，在强烈的遮挡和背景杂波等对抗条件下，实现准确的姿态估计的可行方法是将颜色和深度图像中包含的信息结合起来。如何合理有效地进行融合是一个关键的技术挑战。为了解决这一问题，我们将提取的颜色特征结合到相应的点云中，然后使用网络处理得到的点云，以获得几个点云区域特征。区域特征包含颜色和几何信息，它们可以帮助解决严重遮挡。最后，参考[30]中的细化方法，进一步提高姿态估计的精度。

A. 架构概述

图 1 给出了该方法的总体架构，主要由三个部分组成。第一部分对彩色图像进行语义分割。根据分割结果，得到由蒙版包围框裁剪的彩色图像块和由深度图像转换而来的物体点云。在第二部分，我们使用一个 CNN 结构来提取图像块的特征。然后，根据彩色图像与点云的像素对应关系，将提取的颜色特征组合到点云中。然后，我们使用一个类似 PointNet++的网络对生成的带有颜色特征的点云进行处理，得到全局特征和若干区域特征，然后将全局特征合并到每个区域特征中，得到若干融合特征。在第三部分，我们利用上述融合特征对姿态

进行估计，并以最高的置信度输出姿态。最后，通过姿态细分模块对估计结果进行优化。详情如下所述。

B. 语义分割

我们方法的第一部分是语义分割网络。近年来，在语义分割领域已经取得了许多好的研究成果，如[42]、[43]等。由于本文主要研究的是位姿估计算法，所以我们直接采用了 PoseCNN 中的分割网络。分割网络是一个以彩色图像为输入输出 N+1 通道分割图的编码器-解码器结构。第一个通道描述背景，其他 N 个通道描述 N 个已知类的对象。

C. 特征提取与融合

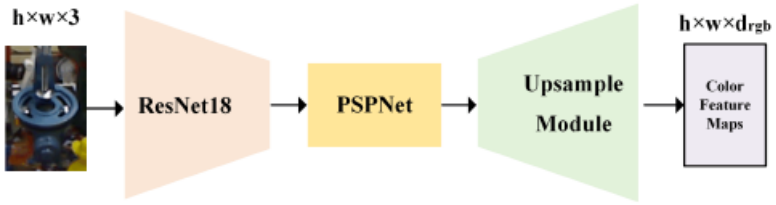

图 2. 颜色特征提取说明。由于 ResNet18 会缩小图像大小，我们采用了设计的上采样模块来保持输出大小不变。

1) 颜色特征提取：根据分割结果，我们对原始彩色图像进行裁剪，得到的图像块大小为$h \times w \times 3$，其中只包含一个特定的对象。然后,我们不直接将彩色图像转化为点云,因为彩色图像比深度通常包含更多的信息,和点云处理网络的特征提取能力太弱,从彩色图像中提取足够的功能,这将导致一个使用颜色信息不足。因此，我们设计了一个 encoder-decoder CNN 对彩色图像块进行特征提取。如图 2 所示，该网络主要由 ResNet18[44]和 PSPNet[45]组成。由于我们将提取的颜色特征进一步结合到点云中，所以颜色特征地图的大小必须与原始图像块相同。因此，我们采用 PSPNet 之后的上采样模块来扩大特征图。该网络最终输出大小为$h \times w \times d_{rgb}$的特征图。
2) 特征融合：在得到颜色特征图后，根据颜色像素与目标点云(由分割的深度图像转换而来)的对应关系选择 N 个颜色像素，并将d_{rgb}维向量组合成相应的点。生成的点云大小为$N \times (3 + d_{rgb})$，其中 N 为点云的点数，3 为点的三维坐标。

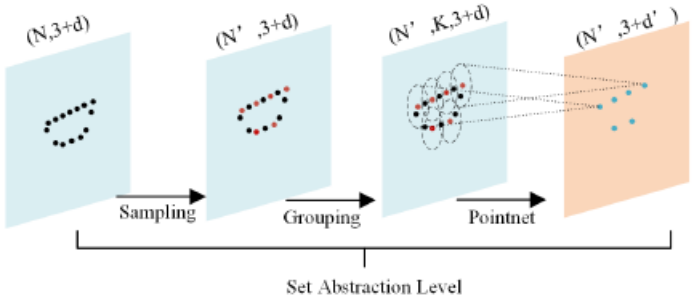

图 3. 集合抽象级别的说明。红点是选定的质心。

我们采用类似 PointNet++的结构来处理生成的点云。PointNet++网络由几个抽象级别组成。每个集合抽象层由采样层、分组层和 PointNet[47]层三部分组成，如图 3 所示。采样层以大小为$N \times (3 + d)$的点集作为输入，使用最远处点采样(FPS)算法从输入点中选择N'个点。这些N'点是彼此之间距离最远的点(以度量的 3D 距离而不是$(3 + d)$维的距离)，它们定义了局部区域的质心。然后分组层结构局部区域设置一个球查询方法,找到了用半径内的所有点查询点(区域质心),然后选择 k 最近的点的质心点集生成一个本地区域。如果点的数量小于 k,我们重用球点来补充。由于球查询的局部邻域保证了一个固定的区域尺度，因此我们不使用寻找固定数量的邻近点的 kNN，从而使局部区域特征在空间上更具有泛化性，这对提取局

部区域特征很重要。分组后得到大小为$N' \times k \times (3+d)$的N'个点集。然后，PointNet 层将N'个局部区域点集编码为大小为$N' \times d'$的特征向量。保留了N'个质心的 3D 坐标。因此，输出数据大小为$N' \times (3+d')$。我们使用几个集合抽象级别，这有助于将点云的局部小区域连续聚集成更大的区域。最后，倒数第二个抽象层输出d_{local}维的N_{patch}特征向量，即将原始点云划分为N_{patch}局部区域。最后一组抽象层对N_{patch}特征向量进行进一步处理，得到d_{global}维的全局特征。我们将全局特征结合到N_{patch}的d_{local}维特征向量中，得到维数为$(d_{local}+d_{global})$的N_{patch}融合特征向量。最终得到的特征向量既包含全局特征又包含局部特征，具有较好的表示能力。

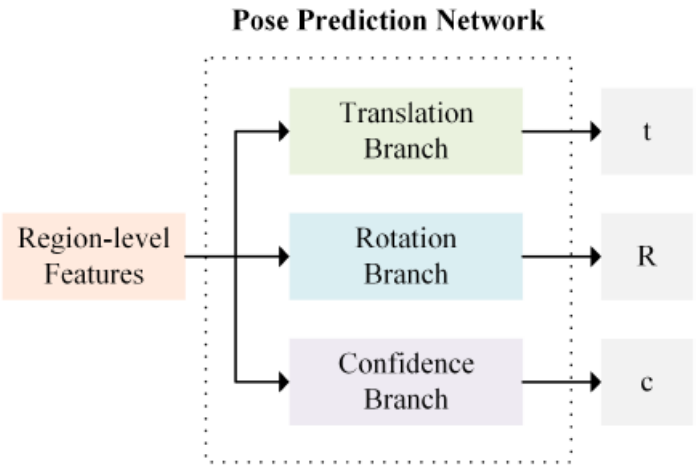

图 4. 位姿预测网络的结构，包括三个分支。

D. 位姿估计与修正

在获得融合特征后，将其发送到神经网络进行姿态估计。如图 4 所示，位姿预测网络有三个分支，分别预测旋转、平移和置信度。每个分支包含三个卷积层。所有卷积核大小均为 1，保证了每个区域融合特征都能使用置信度预测一个姿态。

我们选择置信度最高的区域所预测的姿态作为姿态估计结果。参照[30]的方法，我们设置网络损失函数:

$$L = \frac{1}{N_r}\sum_{i=1}^{N_r}(L_i^p c_i - w\log(c_i)) \qquad (1)$$

其中N_r为区域个数，w为超参数。L_i^p表示经地面真实位姿变换后的对象模型上的采样点与经预测位姿变换后的对应点之间的平均距离。

对于非对称对象，L_i^p的常见设置为:

$$L_i^p = \frac{1}{M}\sum_{j=1}^{M}\left\|(Rx_j+t)-(\hat{R}_i x_j+\hat{t}_i)\right\| \qquad (2)$$

对于对称对象，我们将L_i^p设置为:

$$L_i^p = \frac{1}{M}\sum_{j=1}^{M}\min_{0<k<M}\left\|(Rx_j+t)-(\hat{R}_i x_j+\hat{t}_i)\right\| \qquad (3)$$

其中，M为采样点个数，x_j为M中的第j个点，R为真值旋转，t为真值平移。$\hat{R}_i$和$\hat{t}_i$是第i个区域融合特征估计的旋转和平移量。直觉上，低置信度会导致低姿态估计损失，但会在

第二个周期招致高惩罚，反之亦然。

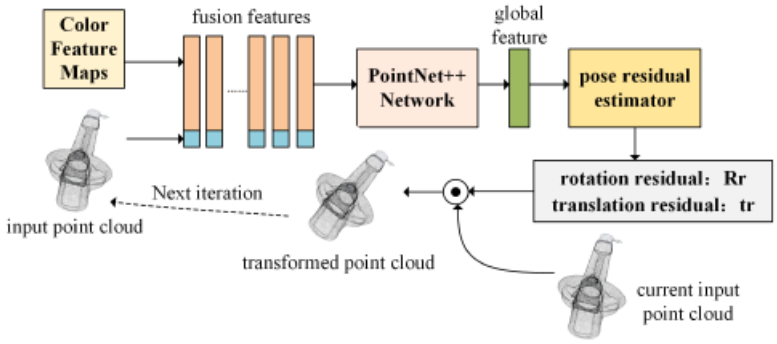

图 5. 姿态细分模块的说明。根据预测的位姿残差对点云进行变换。

得到姿态估计结果后，进行姿态细分。常用的位姿细化算法 ICP 耗时太长，无法满足实时应用的要求。因此，我们采用一种基于神经网络的迭代细化方法，以快速和鲁棒的方式改进最终的姿态估计结果。位姿细化过程如图 5 所示。给定初始预测姿态，首先对点云进行变换。以转换后的点云和原始颜色特征作为输入，细化模块使用一个类似于主网络的神经网络来估计位姿残差。然后根据位姿残差对输入点云进行变换，并将得到的点云作为下一次迭代的输入。经过多次迭代，将这些位姿残差与原始的预测位姿进行连接，得到最终的位姿估计结果。

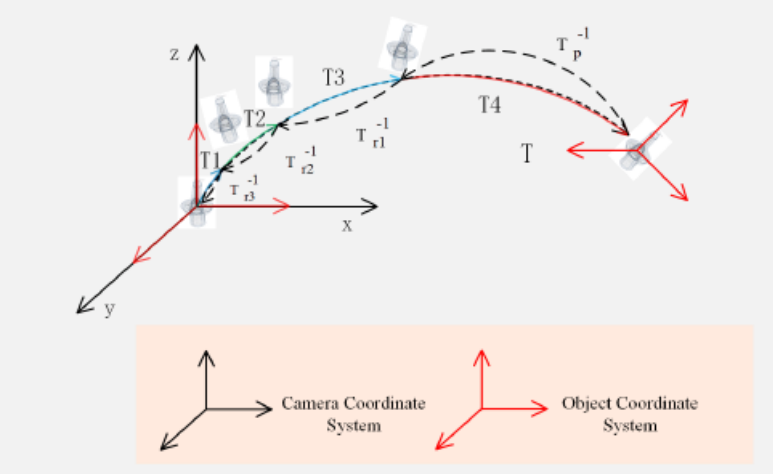

图 6. 姿态优化原理的说明。根据预测的位姿和位姿残差对点云进行变换。

位姿迭代优化原理如图 6 所示。如图，对象是T的姿势可以理解如下：最初的物体坐标系和摄像机坐标系是重合的，然后是对象在相机坐标系转换几次最后的姿势，例如，$\mathrm{T} = \mathrm{T}_4 \cdot T_3 \cdot T_2 \cdot T_1$。另外，根据坐标变换关系，可以得到$\mathrm{X_c} = T \times X_o$，其中$\mathrm{X_c}$和$\mathrm{X_o}$分别为点云在相机坐标系和物体坐标系中的坐标。

姿态预测器的原始输出为$\mathrm{T_p}$。假设$\mathrm{T_p}$对应$\mathrm{T_4}$，可以得到:

$$X_c^1 = T_p^{-1} \times X_c = T_3 \cdot T_2 \cdot T_1 \times X_o \qquad (4)$$

然后，姿态细分模块将$\mathrm{X_c^1}$作为输入，并输出姿态残差$\mathrm{T_{r1}}$。设$\mathrm{T_{r1}}$对应$\mathrm{T_3}$，得到:

$$X_c^2 = T_{r1}^{-1} \times X_c^1 = T_2 \cdot T_1 \times X_o \qquad (5)$$

重复这个过程，我们可以得到如下:

$$T_{r3}^{-1} \cdot T_{r2}^{-1} \cdot T_{r1}^{-1} \cdot T_p^{-1} \times X_c = X_o \qquad (6)$$

然后在 3 次迭代之后得到的结果是最终的位姿为$\mathrm{T_p} \cdot T_{r1} \cdot T_{r2} \cdot T_{r3}$。在N次迭代之后，结果位姿是$\mathrm{T_p} \cdot T_{r1} \cdot T_{r2} \cdots T_{rN}$。

颜色特征与点云具有像素级的对应关系，且对点云进行变换后，对应关系不发生变化。因此，在每次迭代中，我们不重复提取颜色特征，而是从主网络中重新提取颜色特征，并与新的转换后的点云进行融合。迭代细化模块使用全局特征在每次迭代时仅估计一个姿态残差，而不是预测多个有置信度的姿态残差。

IV. 实验

A. 数据集

1) LINEMOD 数据集：LINEMOD 数据集是 6D 对象姿态估计最广泛使用的基准数据集。它包含 13 种具有弱纹理的对象，显示了许多姿态估计的挑战:杂乱的场景，遮挡和光照变化。
2) 遮挡 LINEMOD 数据集：这个数据集是通过另外注释 LINEMOD 图像的子集来创建的。本数据集中的每幅图像都包含多个标注过的对象，这些对象都严重遮挡。

B. 评价指标

我们使用两个标准度量来评估我们的方法:平均距离(添加)度量[8]和2D重投影度量[48]。

平均距离(ADD)度量计算根据地面真实姿态和估计姿态转换的 3D 模型点之间的成对距离的平均值:

$$\mathrm{ADD} = \frac{1}{m} \sum_{x \in \mathcal{M}} \left\| (Rx + t) - (\hat{R}x + \hat{t}) \right\| \qquad (7)$$

其中M为三维模型点集，m为点个数。R和t分别为真值旋转和平移。$\hat{R}$和$\hat{t}$分别是估计的旋转和平移量。对于对称对象，点之间的匹配对于某些视图是模糊的。为了解决这个问题，我们采用了[8]中提出的方法。平均距离使用最近点距离计算:

$$\mathrm{ADD-S} = \frac{1}{m} \sum_{x_1 \in \mathcal{M}} \min_{x_2 \in \mathcal{M}} \left\| (Rx_1 + t) - (\hat{R}x_2 + \hat{t}) \right\| \quad (8)$$

如果平均距离小于预定义的阈值，6D 姿势被认为是正确的。通常，阈值设置为 3D 模型直径的 10%。

当使用二维重投影度量时，我们认为当物体的三维模型点的二维投影与地面真实姿态之间的平均距离小于 5 个像素[48]时，姿态估计是正确的。这测量了物体的真实图像投影与使用估计姿态获得的图像投影的接近程度。

C. 实现细节

颜色特征提取网络的输出是 32 个颜色特征映射，即每个像素都用一个 32 维的特征向量来表示。我们将 32 维向量与点云的三维坐标相结合，然后将它们发送到基于 PointNet++ 的网络中。该网络有三个集合的抽象层，最终输出 8 个 1536 维的区域级特征向量，包括 512 维的局部区域特征和 1024 维的全局特征。然后，姿态预测网络输出姿态估计结果和每个特征向量的置信度。姿态细化模块由三个完全连接的层组成，从全局特征中输出姿态残差，我们对所有实验都使用两次细化迭代。

我们在 Intel Core i7-9700K@3.60GHz CPU 和单个 NVIDIA GeForce RTX 2070 GPU 上运行我们的代码。在训练过程中，我们遵循了[48]中的设置，它选择 15%的图像进行训练，剩下的用于测试。所选的训练图像使相关的对象姿态的最小角距离为 15°。对于 LINEMOD 数据集，使用 2373 张图像进行训练，使用 13404 张图像进行测试。对于遮挡 LINEMOD 数据集，使用 181 张图像进行训练，使用 1031 张图像进行测试。在训练中不使用其他渲染图像。对于所有的epoch，我们为 LINEMOD 数据集设置 6000，为遮挡 LINEMOD 数据集设置 4000。由于位姿细化模块难以收敛，我们没有与主网络共同训练。因此，我们首先对主网络进行训

练，直到它收敛，然后对主网络进行固定，然后开始训练位姿细化模块。

TABLE I
THE ACCURACIES OF OUR METHOD AND THE BASELINE METHODS ON THE LINEMOD DATASET IN TERMS OF THE ADD(-S) METRIC.

	w/o refinement				w/refinement				
	Pix2Pose	PVNet	DenseFusion	Ours	BB8 w/ref	PoseCNN+DeepIM [49]	SSD6D+ICP	DenseFusion	Ours
ape	58.1	43.62	79.5	80.46	40.4	77.0	65	92.3	**93.51**
bench v.	91.0	**99.90**	84.2	84.97	91.8	97.5	80	93.2	95.15
cam	60.9	86.86	76.5	82.16	55.7	93.5	78	**94.4**	93.92
can	84.4	95.47	86.6	84.74	64.1	**96.5**	86	93.1	95.47
cat	65.0	79.34	88.8	91.92	62.6	82.1	70	96.5	**98.60**
driller	76.3	**96.43**	77.7	81.27	74.4	95.0	73	87.0	94.84
duck	43.8	52.58	76.3	84.41	44.3	77.7	66	92.3	**95.96**
eggbox	96.8	99.15	99.9	99.91	57.8	97.1	100	99.8	**100.0**
glue	79.4	95.66	99.4	99.61	41.2	99.4	**100**	**100.0**	99.71
hole p.	74.8	81.92	79.0	81.92	67.2	52.8	49	92.1	**94.48**
iron	83.4	**98.88**	92.1	92.95	84.7	98.3	78	97.0	97.54
lamp	82.0	**99.33**	92.3	93.38	76.5	97.5	73	95.3	98.94
phone	45.0	92.41	88.0	85.40	54.0	87.7	79	92.8	**95.10**
average	72.4	86.27	86.2	87.94	62.7	88.6	79	94.3	**96.40**

TABLE II
THE ACCURACIES OF OUR METHOD AND THE BASELINE METHODS ON THE LINEMOD DATASET IN TERMS OF THE 2D REPROJECTION METRIC.

	YOLO6D	BB8	BB8 w/ref	DenseFusion	Ours
ape	92.10	95.3	96.6	**96.85**	96.09
bench v.	**95.06**	80.0	90.1	88.26	92.92
cam	93.24	80.9	86.0	**93.82**	93.14
can	**97.44**	84.1	91.2	96.06	94.09
cat	97.41	97.0	**98.8**	96.11	98.00
driller	79.41	74.1	80.9	84.84	**91.58**
duck	94.65	81.2	92.2	**98.50**	97.18
eggbox	90.33	87.9	91.0	**99.34**	**99.34**
glue	96.53	89.0	92.3	95.46	**97.87**
hole p.	92.86	90.5	**95.3**	87.91	93.53
iron	82.94	78.9	84.8	93.97	**94.28**
lamp	76.87	74.4	75.8	92.32	**96.26**
phone	86.07	77.6	85.3	**92.99**	92.22
average	90.37	83.9	89.3	93.60	**95.14**

D. 在 LINEMOD 数据集上的表现

表 I 显示了我们的方法和其他以前的方法在 LINEMOD 数据集上根据 ADD(-S)度量的姿态估计结果。在这个表中，我们根据姿态细分模块的存在和缺失将这些方法分为两类。可以看出，无论是否有位姿细化模块，我们的方法的精度都超过了目前最先进的方法。在没有细化模块的情况下，我们将我们的方法与 Pix2Pose[21]、PVNet[20]和 DenseFusion[30]进行比较。该方法的精度达到 87.94%，优于其他三种方法，甚至比 BB8 和 SSD-6D 两种深度细化方法分别高出 25.24%和 8.94%。使用细化模块，我们的方法单独改进了 8.46%，准确率达到 96.40%。该方法的准确率优于其他方法，分别比 PoseCNN 和 DenseFusion 的准确率高 7.8%和 2.1%。在表 II 中，我们将我们的方法与 BB8[17]、YOLO6D[18]和 DenseFusion[30]的二维重投影度量进行比较。从表中可以看出，我们的方法的准确率为 95.14%，达到了最好的性能。特别是，我们的方法在 LINEMOD 数据集上的两个指标都比 DenseFusion 获得了更好的性能。一些姿态估计的定性结果如图 7 所示。

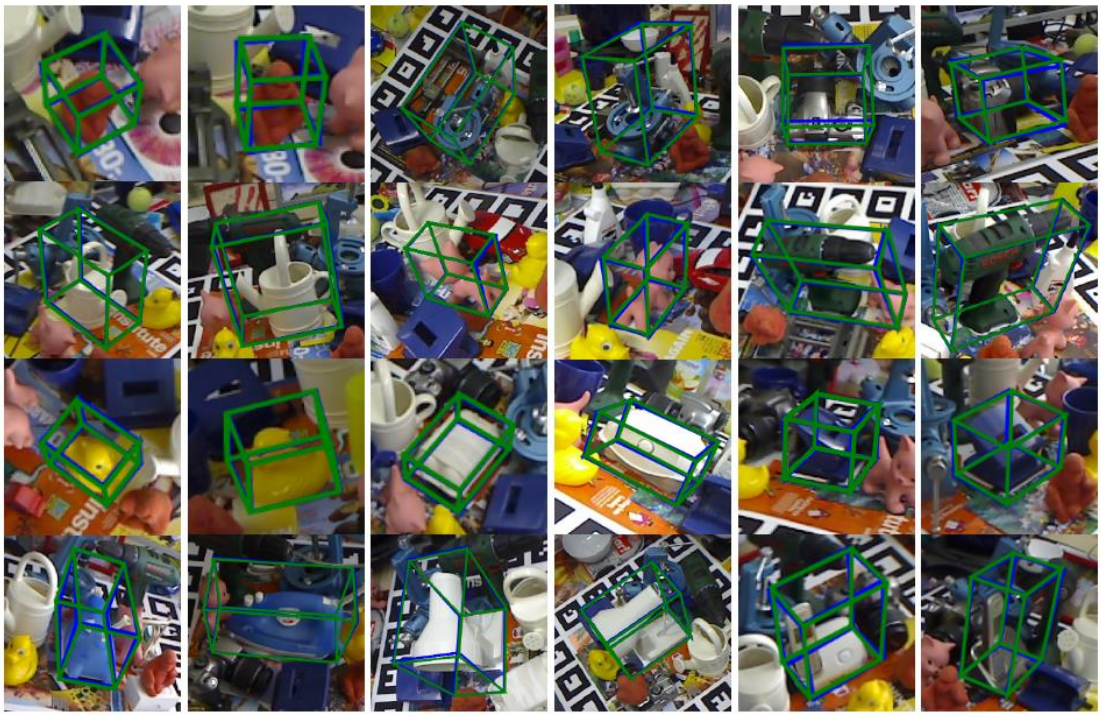

图 7. 在 LINEMOD 数据集上显示结果。绿色的边界框代表真实的姿势，蓝色的边界框代表我们的预测。

E. 在遮挡 LINEMOD 数据集上的表现

表 III 和表 IV 分别为遮挡 LINEMOD 数据集在 ADD(-S)度量和 2D 重投影度量下的结果。我们的方法在这两个指标的所有方法中取得了最好的性能。具体来说，我们的方法在二维重投影度量上分别优于分割驱动和密集融合，其边际分别为 40.87%和 1.44%。需要注意的是，分割驱动的方法只使用 RGB 图像，而 DenseFusion 和我们的方法使用 RGB-D 图像。在 ADD(-S)度量方面，我们的方法的准确率为 83.56%，比采用 ICP 和 DenseFusion 的 PoseCNN 分别高出 5.56%和 2.29%。这表明该方法能有效地处理遮挡和背景杂波。具体来说，与 DenseFusion 相比，我们的方法在除了猿之外的几乎所有对象上都表现出了更好的性能，这表明使用区域级特征的姿态估计比使用像素级特征的姿态估计更准确，对遮挡更有鲁棒性。值得注意的是，对猿和鸭的测试结果并不令人满意。通过对输入图像的分析，我们认为这是因为部分图像上的两类物体被严重遮挡，只能看到非常小的区域。因此，从彩色图像和点云中提取的特征受到背景的严重干扰，导致区域级特征含有过多的噪声，难以准确估计姿态。对于一般咬合，可以准确地估计姿势。图 8 显示了部分测试结果。

TABLE III
THE ACCURACIES OF OUR METHOD AND THE BASELINE METHODS ON THE OCCLUSION LINEMOD DATASET IN TERMS OF THE **ADD(-S)** METRIC.

	Segmentation-driven (RGB)	Pix2Pose (RGB)	PVNet (RGB)	Hinterstoisser *et al.* [50] (depth)	Michel *et al.* [51] (RGB-D)	PoseCNN+ICP (RGB-D)	DenseFusion (RGB-D)	Ours (RGB-D)
ape	12.1	22.0	15.81	**81.4**	80.7	76.2	73.20	68.43
can	39.9	44.7	63.30	**94.7**	88.5	87.4	88.64	92.65
cat	8.2	22.7	16.68	55.2	57.8	52.2	72.22	**77.97**
driller	45.2	44.7	65.65	86.0	94.7	90.3	92.50	**95.13**
duck	17.2	15.0	25.24	**79.7**	74.4	77.7	59.65	62.14
eggbox	22.1	25.2	50.17	65.5	47.6	72.2	94.24	**96.06**
glue	35.8	32.4	49.62	52.1	73.8	76.7	92.62	**93.54**
hole p.	36.0	49.5	39.67	95.5	**96.3**	91.4	78.77	83.64
average	27.0	32.0	40.77	76.3	76.7	78.0	81.27	**83.56**

TABLE IV
THE ACCURACIES OF OUR METHOD AND THE BASELINE METHODS ON THE OCCLUSION LINEMOD DATASET IN TERMS OF THE **2D REPROJECTION** METRIC.

	Segmentation -driven	Oberweger [52]	DenseFusion	Ours
ape	59.1	69.6	**85.78**	80.41
can	59.8	82.6	85.21	**90.40**
cat	46.9	65.1	**88.10**	87.89
driller	59.0	73.8	85.98	**91.53**
duck	42.6	61.4	70.85	**72.82**
eggbox	11.9	13.1	**93.03**	92.63
glue	16.5	54.9	88.41	**88.66**
hole p.	63.6	66.4	77.99	**81.89**
average	44.9	60.9	84.33	**85.77**

图 8. 在遮挡 LINEMOD 数据集上显示结果。绿色的边界框代表真实的姿势，而其他颜色的边界框代表我们的预测。

F. 迭代神经网络优化方法与 ICP 优化方法的比较

为了评估迭代神经网络的性能，我们在 LINEMOD 数据集上与 ICP 算法进行了对比实验。我们采用最大迭代次数为 20 的点对点 ICP。实验结果表明，迭代神经网络对单个目标的平均运行时间为 0.0026 秒，而 ICP 网络的平均运行时间为 0.0505 秒。我们的方法比 ICP 算法快 20 倍。在 ADD(-S)度量方面的精度比较如表 V 所示，我们可以看到迭代神经网络在几乎每个对象上都比 ICP 获得了更好的性能。迭代神经网络的平均精度比 ICP 算法高 5.84%。我们注意到在一些物体上(如猿，鸭子，胶水)，使用 ICP 甚至会降低精度。原因是物体很小，点之间距离太近，这可能会导致 ICP 算法选择错误的最近点，导致结果不佳。

V. 结论与未来的工作

本文提出了一种基于 RGB-D 图像的已知目标 6D 姿态估计方法。我们的方法将提取的颜色特征结合到点云中，然后使用 PointNet++网络处理得到的点云，并提取一个全局特征和几个局部区域特征。将全局特征与每个局部特征相结合，得到多个融合特征，从而估计出最终的 6D 姿态。通过这种融合方法，我们的方法有效地结合了颜色和深度信息，局部和全局信息，在两个基准数据集上实现了最先进的性能。实验表明，该方法能够有效地处理目标姿态估计中前景遮挡、背景杂波和光照变化等问题。颜色信息与深度信息的融合方法也为姿态估计的研究提供了一种新的思路。在未来，我们将考虑将我们的方法扩展到视频，这更适合于动态应用。与图像中的姿态估计相比，视频中需要考虑时间上下文，这有助于提高姿态估计的时间效率和精度。如何利用时间语境，并将其结合到我们的方法中，将是未来工作的

重点。