第 60 卷 第 16 期/2023 年 8 月/激光与光电子学进展 研究论文

激光与光电子学进展 先进成像

基于关键点距离表征网络的物体位姿估计方法

夏梦，杜弘志，林嘉睿，孙岩标*，郁继贵

天津大学精密测试技术及仪器国家重点实验室，天津 300072

摘要 提出一种新型的关键点距离表征学习网络，该网络利用位姿变换过程的几何不变性信息，在网络中引入距离量的估计，进而推导出稳健关键点，以此来提升基于深度学习的六自由度物体位姿估计方法的精度。所提方法包含两个阶段。首先，设计了关键点距离表征网络，通过一种骨干网络模块和特征融合结构实现RGB-D图像特征提取，并结合多层感知机预测物体逐点相对于关键点的距离量、语义和置信度。其次，根据可视点投票法及四点距离定位法，利用网络输出的多维信息推理计算关键点坐标，并最终通过最小二乘拟合算法得到物体位姿。为了证明所提方法的有效性，在公开数据集LineMOD和YCB-Video上进行了测试，实验结果表明，所提方法相比于原PSPNet框架中的ResNet参数量减少一半且精度有所提升，在两个数据集上精度分别提升了1.1个百分点和5.8个百分点。

关键词 机器视觉；六自由度位姿估计；深度学习；关键点距离表征网络；特征提取

中图分类号 TP391 **文献标志码** A **DOI**：10.3788/LOP223015

Object Pose Estimation Method Based on Keypoint Distance Network

Xia Meng, Du Hongzhi, Lin Jiarui, Sun Yanbiao*, Zhu Jigui

State Key Laboratory of Precision Measurement Technology and Instrument, Tianjin University, Tianjin 300072, *China*

Abstract Herein, we present a novel keypoint distance learning network, which utilizes geometric invariance information in pose transformation. Distance estimation is added to the network and robust keypoints are determined, which improves the pose estimation accuracy within six degrees of freedom based on deep learning. The proposed method consists of two stages. First, a keypoint distance network is designed, which achieves RGB-D image feature extraction using a backbone network module and a feature fusion structure and predicts the distances of each point relative to the keypoints, semantics, and confidence using a multilayer perceptron. Second, based on the visual point voting method and the four point distance positioning method, keypoint coordinates are calculated using the multi-dimensional information output from the network. Finally, object poses are obtained through the least square fitting algorithm. To prove the effectiveness of the proposed method, we tested it on public datasets LineMOD and YCB-Video. Experimental results show that the network parameters of this method can be reduced by 50% with improved accuracy compared to ResNet in the original PSPNet framework, with accuracy improvements of 1.1 percentage points and 5.8 percentage points on two datasets, respectively.

Key words machine vision; six degrees of freedom pose estimation; deep learning; keypoint distance network; feature extraction

1 引 言

物体的六自由度(6D)位姿估计是计算机视觉领域的重要问题，是机械臂作业、自动驾驶以及增强现实任务中的关键技术。其主要任务是根据相机等视觉传感器采集到的信息，估计场景中物体与传感器之间的变换关系，包括三自由度旋转变换和三自由度平移变换[1-2]。

传统的6D位姿估计方法需要人工提取并选择合适的特征，如纹理、形状等全局特征和SIFT、SURF、ORB特征点等局部特征[3]，采用机器学习等方法实现与模板之间的特征匹配。但由于人工选取的特征比较单一且鲁棒性差，在目标被遮挡、光照变换强烈、背景杂乱等复杂情况下结果较差。随着深度学习技术的飞速发展，深度卷积神经网络(DCNN)被应用到此任务

收稿日期：2022-11-10；**修回日期**：2022-11-22；**录用日期**：2022-11-24；**网络首发日期**：2023-01-04

基金项目：国家自然科学基金(52075382)

通信作者：*yanbiao.sun@tju.edu.cn

中，代替人工自动提取更深层次的抽象特征，鲁棒性高，在复杂环境中表现出更优的性能。Xiang 等[4]、Chen 等[5]、Wang 等[6]通过神经网络直接回归目标物体6D位姿，此种框架较为简洁高效，但由于旋转空间是非线性的，网络回归的结果往往不够精确，还需结合iterative closest point(ICP)等耗时的迭代优化过程才能达到更优的水平。而 Rad 等[7]、Zhao 等[8]、Peng 等[9]则采用二阶段的方法，先通过神经网络进行关键点检测再通过 perspective-n-point(PnP)算法求解位姿。此方法的结果更加稳定。与 Rad 等[7]不同的是，Peng 等[9]选用目标表面的8个关键点代替包围框的8个角点，使得网络可以更好地获得上下文信息。同时 Peng 等[9]对目标可视部分的每个像素特征都回归了一个指向关键点的单位向量，在遮挡情况下具有很好的鲁棒性。但它们都是基于二维投影空间的检测，二维中较小的误差在三维空间中可能很大，且三维中不同关键点的二维投影可能会重叠，较难分辨。此外，二维投影会使刚性物体的部分几何约束信息丢失。随着 RGB-D 传感器的发展，深度图像的引入为物体位姿估计增加了更多的几何信息，使其向三维空间的拓展成为可能，例如 Frustum PointNets[10]、MPCS-Net[11]等。在 Wang 等[12]提出的 DenseFusion 中，深度图被转化为点云数据通过 PointNet 提取点云特征，再与 RGB 图像特征逐像素融合，以生成目标最终的6D位姿预测。结合深度图像的物体位姿估计相较于仅使用 RGB 图的方法通常能得到更加精确且鲁棒的位姿，对弱纹理物体的估计也能保持结果的稳定性。综上所述，本文基于 RGB-D 图像采用深度神经网络实现三维空间中关键点的检测，改善二维关键点方法的缺陷，进而实现具有鲁棒性的物体位姿估计。

目前一些基于关键点检测网络的方法直接输出关键点坐标或指向关键点的向量，在相机坐标系中，关键点坐标或指向关键点向量随着物体姿态的变换而变化，使得网络较难学习到物体其他可视点与关键点之间的关系，网络泛化能力较弱，在场景变换等情况下鲁棒性较低。针对这一问题，本文提出基于距离表征的关键点检测网络，在网络中引入距离量，使网络直接输出物体每一可视点与关键点三维空间之间的距离。由于刚体质点之间的距离在姿态变换时保持不变，因此网络更容易学习到逐点与关键点之间的距离特征，从而提升网络泛化能力。

在第一阶段关键点检测网络中采用 PSPNet 框架[13]和 RandLA-Net 框架[14]分别对 RGB 图和由深度图像转化的点云进行特征提取融合。并且在 RGB 图的特征提取中设计了一种新的骨干网络模块，结合了 ResNet[15]和 DenseNet[16]的优势，大大减少了网络的参数量。第二阶段将网络输出的关键点距离通过推理计算得到物体的6D位姿，由于网络输出会存在一些异常值点影响后续计算，因此引入置信度网络，输出每一特征点距离预测的置信度，剔除置信度低的点，从而提升最终的位姿估计精度。

2　基于关键点距离表征网络

所提方法的总体框架如图1所示，主要包括神经网络和推理计算两部分。其中，神经网络部分由特征提取融合网络、关键点距离预测模块、结合聚类的语义分割模块和置信度网络组成。首先根据已知的相机内参将深度图转换成点云，再分别对 RGB 图和点云进行特征提取，并逐层逐点融合得到包含纹理信息和几何信息的特征，分别输入后续的关键点距离预测网络、语义分割网络和置信度网络，得到逐点与关键点的距离、逐点语义和逐点置信度。其中：距离量用于关键点计算；逐点语义用于物体分割；置信度用于异常值点剔除。

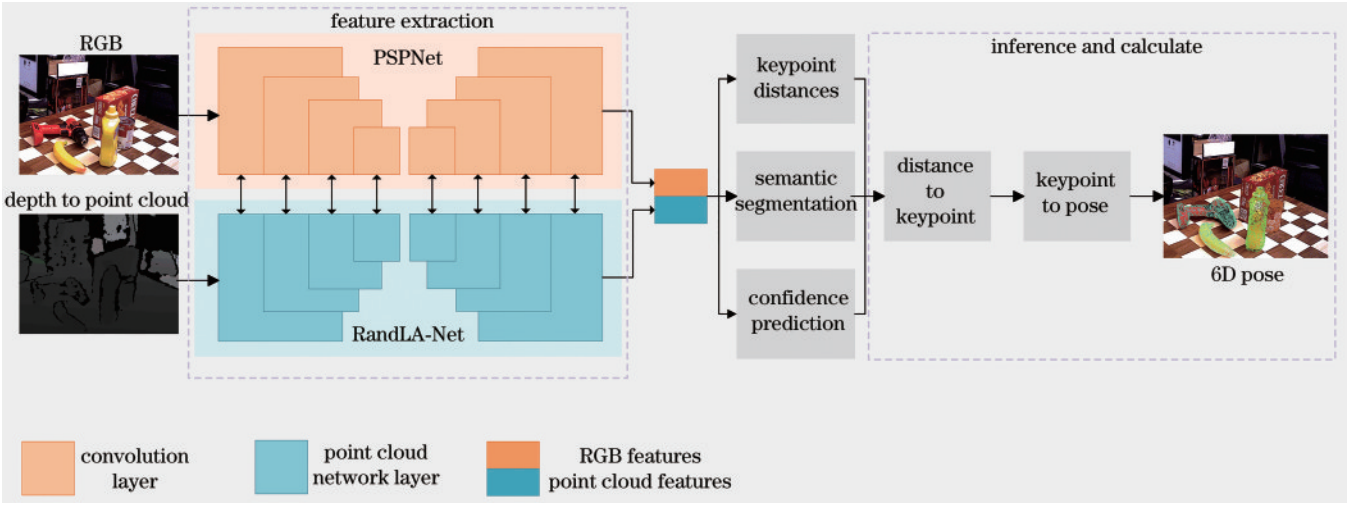

图1　位姿估计总体框架

Fig. 1　Overview of pose estimation

2.1　特征提取融合网络

为了获取图像中的纹理颜色信息和几何信息用于关键点的检测，需要从输入的RGB-D图像中分别提取特征，同时将物体同一位置的两种特征对应融合。首先，根据已知的相机内参将深度图转化为点云。分别采用PSPNet[13]和RandLA-Net[14]的核心特征提取框架中的卷积层和点云网络层对RGB图和点云进行特征提取。简单地将图像特征和点云特征拼接起来可以实现两种信息的结合，但在纹理特征或点云特征有部分缺失时，网络准确性随之下降。因此所提方法在每个编码层和解码层都将两者融合，如图1所示，在每个特征层次都实现了图像信息和几何信息的融合，以实现特征的互补。当有一方信息缺失时，另一方可以弥补，实现稳健可靠的特征提取。

PSPNet的核心模块是金字塔池化模块(pyramid pooling module)，该模块可以将不同区域的上下文信息聚合起来，增加全局信息的获取能力同时增大感受野。设计了一种新的骨干网络模块替代网络中原本的ResNet101，如图2所示。它结合DenseNet[15]与ResNet[16]的优势，包含稠密连接结构和残差短连接结构，增强了网络的学习能力并且减小了网络的规模，减少运行时间。RandLA-Net是一种基于随机降采样和局部特征聚合的轻量级的网络，采用随机降采样，大大减小了计算量，节省运行时间，同时采用局部特征聚合解决随机降采样导致的信息丢失问题。此网络框架计算效率高且内存占用少，同时可增大每个特征点有效的感受野，适用于本研究的点云特征提取。

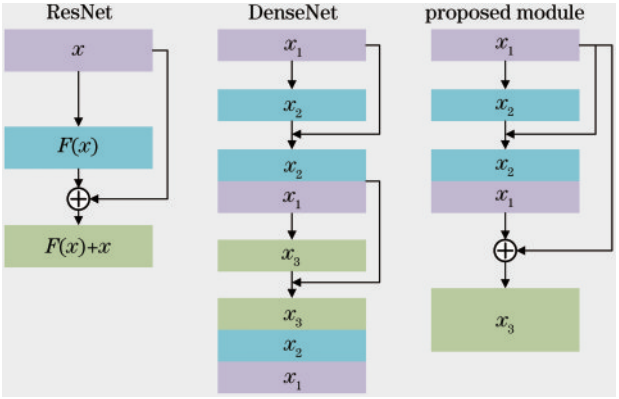

图2　骨干网络模块

Fig. 2　Backbone network module

RGB特征和点云特征逐层的融合结构如图3所示。由于RGB图和深度图是对齐的，可以根据深度图将RGB图中的每个像素点均转换为三维空间的点坐标，从而可以在RGB特征中找到与点云特征相对应的特征。将对应特征拼接后通过共享多层感知机(shared MLP)得到融合后的特征。

2.2　关键点距离预测模块

2.2.1　关键点的选择

为实现基于关键点的位姿估计，首先需要根据物体已知的CAD模型选择合适的关键点作为先验知识，关键点常常采用物体检测中的三维包围盒的8个角点[8]，但这8个点是虚拟的点，并且距物体的距离较远，使得基于点的网络无法获得上下文信息，从而产生较大的定位误差。针对这一问题，采用最远点采样(FPS)算法在物体表面选择关键点。在这一过程中，选择物体模型的中心点作为初始点，然后每次选择一个距离所有选中点最远的新点加入集合，直到得到所有关键点。得到的关键点分散在物体表面边缘，具有较丰富的几何信息。为了使关键点同时包含物体的纹理信息，结合SIFT特征提取方法，从不同视角对物体CAD模型进行图片采集，采用SIFT特征提取得到候选的2D关键点，再将其转换到三维空间，最后通过FPS算法得到N个关键点。

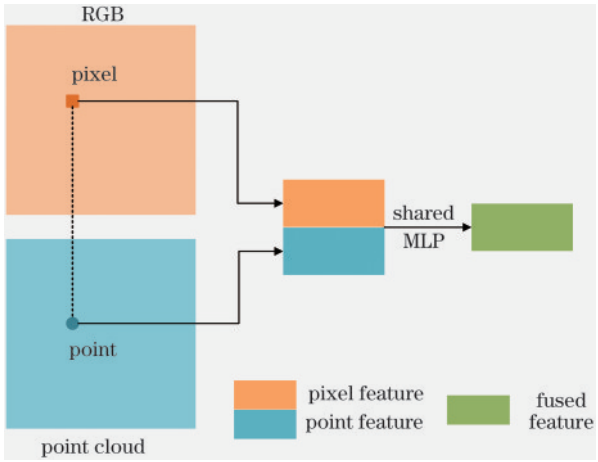

图3　特征融合结构

Fig. 3　Feature fusion structure

2.2.2　关键点距离预测

通过特征提取融合网络得到RGB-D的密集融合特征后，采用MLP实现关键点距离的预测。MLP层与层之间是全连接的，包括输入层、输出层和隐藏层，具备快速解决复杂问题的能力。输入提取特征，经过隐藏层，输出为N通道，分别表示其他可视点距N个关键点的距离。对于关键点距离量的监督采用L1损失，损失函数如式(1)所示：

$$L_{\text{keypoint}}=\frac{1}{M}\sum_{i}^{M}\sum_{j}^{N}\left\|\boldsymbol{d}_i^j-\boldsymbol{d}^{*\,j}_i\right\|,\tag{1}$$

式中：N为关键点的个数；M为物体可视点的数量；$\boldsymbol{d}$为逐点与关键点的距离；$\boldsymbol{d}^*$为距离的真值。

2.3　结合聚类的语义分割

当场景中存在多个物体时，为实现物体分割，以前的方法[12,14]通过目标检测或者分割对图像进行预处理从而获得单个物体的包围框。而所提方法在语义分割的基础上结合物体中心点的预测，实现结合聚类的语义分割，以达到更好的分割效果。采用MLP分别预测M个点的语义和M个点相对于中心点的偏移$\Delta\boldsymbol{v}_i\left(\Delta x_i,\Delta y_i,\Delta z_i\right)_{i=1}^{M}$，已知每点的三维坐标$\boldsymbol{p}_i\left(x_i,y_i,z_i\right)_{i=1}^{M}$，

根据式(2)可以得到逐点预测出的物体中心点$\boldsymbol{p}_{\mathrm{ctr}\,i}$。

$$\boldsymbol{p}_{\mathrm{ctr}\,i}=\boldsymbol{p}_i+\Delta\boldsymbol{v}_i。\tag{2}$$

根据点的语义类别将M个中心点分成S类，在理想情况下，属于同一类别的点预测的中心点应该是一致的。但实际的网络输出会有误差，使预测的中心点不完全一致，而对其直接求其平均值会使预测错误的点对结果影响较大，因此所提方法将属于同一类别的中心点通过MeanShift聚类算法得到每个物体的中心点，消除错误点的影响。分别计算M个预测的中心点与S个物体中心点的距离$\{d_{ij}\}|_{i=1}^{S}|_{j=1}^{M}$，选择距离$M$点最近的中心点的语义作为该点的语义更新初始语义分割，并且删除最小距离大于$0.8R$的点，R为物体的半径。与普通的语义分割相比，结合聚类的语义分割可以得到更优的结果，同时对于外观相似但大小不同的物体也有较好的分割效果。

在此任务中，中心点预测的损失函数为

$$L_{\mathrm{center}}=\frac{1}{M}\sum_{i}^{M}\left\|\Delta\boldsymbol{v}_i-\Delta\boldsymbol{v}_i^{*}\right\|,\tag{3}$$

式中：$\Delta\boldsymbol{v}_i$为预测的中心点偏移；$\Delta\boldsymbol{v}_i^{*}$为中心点偏移的真值。语义分割的损失采用Focal损失：

$$L_{\mathrm{semantic}}=-\alpha\left(1-q_i\right)^{\gamma}\log q_i,\tag{4}$$

式中：α和γ均为可调节参数，α用于调节正负样本损失之间的比例，γ用于调节难分与易分样本的损失贡献；q_i为预测类别正确的概率。

2.4　置信度模块

在以下几种情况时，网络的输出可能会存在一些异常值，影响估计结果：1)当两物体相互堆叠时，在其交界处可能会存在分割错误的点；2)当物体外观与背景相似或与另一物体相似时，存在预测错误的点；3)在距离关键点极近的位置，与关键点的距离值接近0，此时网络的输出值可能会出现异常。针对这一问题，设计置信度网络模块，在输出每点距离预测的同时输出每点的置信度，通过置信度评价距离预测的准确程度。此模块根据提取的逐点特征和关键点距离预测误差预测置信度。

置信度网络模块结构如图4所示，作为一个分类任务，包括MLP和Softmax层。网络的输入为上文中网络输出的拼接，包括逐点融合特征、关键点距离预测值以及语义分割结果，输出为两通道的向量，分别代表类别0和1的概率。置信度的标签由关键点距离预测的误差决定，当误差大于设定的阈值时标签为0，小于阈值时标签为1。损失函数为

$$L_{\mathrm{conf}}=-\frac{1}{M}\cdot\sum_{i=1}^{M}\left[y_i\cdot\lg c_i+\left(1-y_i\right)\lg\left(1-c_i\right)\right],\tag{5}$$

式中：y_i为置信度标签；c_i为预测的置信度。从损失函数可以看出，输出置信度值越接近1，损失越小。网络训练置信度值为1，因此将输出1通道的值作为最终的逐点置信度是合理的。

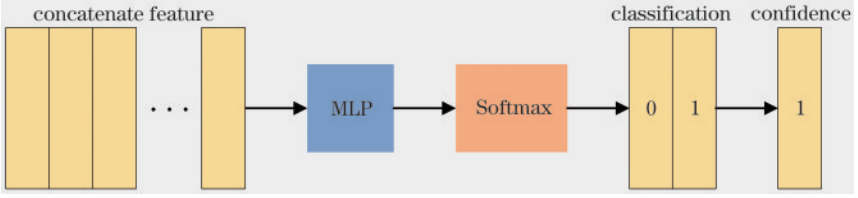

图4　置信度网络

Fig. 4　Confidence network

3　位姿推理计算

在基于关键点的位姿估计方法中，需要将网络输出的结果经过第二阶段的处理以得到最终的位姿。第二阶段的推理计算主要包括从距离量到关键点的计算和从关键点到位姿的计算，如图5所示。根据第一阶段的网络语义和置信度实现物体的分割以及异常值的剔除得到每个物体可视点到关键点的距离量，再根据可视点多少选择投票法或四点定位法实现距离量到关键点的转换。从计算得到的相机坐标系下的N个关键点坐标中挑选最优的3个点，与先验的物体坐标系下的对应关键点坐标组成关键点对，并通过最小二乘拟合计算得到最终的物体位姿。经过这两部分的计算可以将网络输出的语义预测、距离预测和置信度预测转化为物体的精确位姿。

3.1　距离量到关键点的计算

想要完成距离量到关键点的转换，首先需要根据每点的语义将不同的物体区分开，分别得到每个物体可视点的距离值，同时根据置信度值剔除异常的距离量。由于点云特征提取网络中对点云的下采样是随机的，每个物体采样到的点数量不同，点的数量会对估计结果产生影响，因此根据物体采样点数量的多少，采用不同方法完成距离量到关键点的转换。

当物体采样点较少时，从距离量到空间点的转换采用四点定位法。根据已知的4点坐标$\boldsymbol{p}_1$、$\boldsymbol{p}_2$、$\boldsymbol{p}_3$、$\boldsymbol{p}_4$和4个点到点$\boldsymbol{p}_0$的距离量d_1、d_2、d_3、d_4，可以求得另一点$\boldsymbol{p}_0$的空间坐标。首先根据距离公式可得

$$\begin{cases}\left(x_1-x_0\right)^2+\left(y_1-y_0\right)^2+\left(z_1-z_0\right)^2=d_1^2\\ \left(x_2-x_0\right)^2+\left(y_2-y_0\right)^2+\left(z_2-z_0\right)^2=d_2^2\\ \left(x_3-x_0\right)^2+\left(y_3-y_0\right)^2+\left(z_3-z_0\right)^2=d_3^2\\ \left(x_4-x_0\right)^2+\left(y_4-y_0\right)^2+\left(z_4-z_0\right)^2=d_4^2\end{cases}。\tag{6}$$

由于式(6)中存在高次项，不利于方程的求解，所以需要对方程进行三次差分消除高次项：

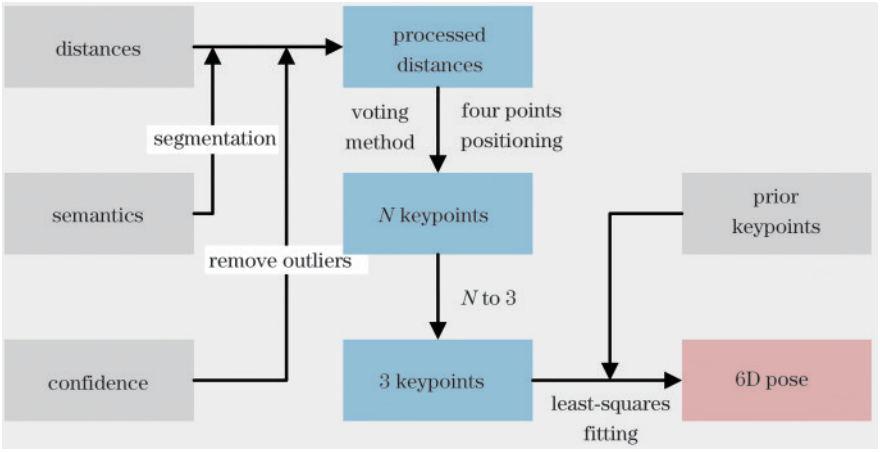

图 5　推理计算流程图

Fig. 5　Flow chart of reasoning calculation

$$\begin{cases} 2x_0(x_2-x_1)^2+2y_0(y_2-y_1)^2+2z_0(z_2-z_1)^2=d_1^2-d_2^2+x_2^2-x_1^2+y_2^2-y_1^2+z_2^2-z_1^2 \\ 2x_0(x_2-x_1)^2+2y_0(y_2-y_1)^2+2z_0(z_2-z_1)^2=d_1^2-d_3^2+x_3^2-x_1^2+y_3^2-y_1^2+z_3^2-z_1^2 \\ 2x_0(x_2-x_1)^2+2y_0(y_2-y_1)^2+2z_0(z_2-z_1)^2=d_1^2-d_4^2+x_4^2-x_1^2+y_4^2-y_1^2+z_4^2-z_1^2 \end{cases}。\tag{7}$$

用矩阵可表示为

$$\boldsymbol{A}\boldsymbol{c}=\boldsymbol{b},\tag{8}$$

式中：$\boldsymbol{A}=\begin{bmatrix} 2(x_2-x_1) & 2(y_2-y_1) & 2(z_2-z_1) \\ 2(x_3-x_1) & 2(y_3-y_1) & 2(z_3-z_1) \\ 2(x_4-x_1) & 2(y_4-y_1) & 2(z_4-z_1) \end{bmatrix}$；$\boldsymbol{c}=\begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}$；$\boldsymbol{b}=\begin{bmatrix} d_1^2-d_2^2+x_2^2-x_1^2+y_2^2-y_1^2+z_2^2-z_1^2 \\ d_1^2-d_3^2+x_3^2-x_1^2+y_3^2-y_1^2+z_3^2-z_1^2 \\ d_1^2-d_4^2+x_4^2-x_1^2+y_4^2-y_1^2+z_4^2-z_1^2 \end{bmatrix}$。

$\boldsymbol{c}$ 即为待求点，如果 $\boldsymbol{A}$ 的逆矩阵存在即可通过式(9)求得待求点坐标。

$$\boldsymbol{c}=\boldsymbol{A}^{-1}\boldsymbol{b}。\tag{9}$$

根据上述4点定位法可知，4点可以确定1个候选关键点，在物体所有可视点中随机选择4点计算得到关键点候选，重复多次可得到多个关键点候选点。由于预测距离存在误差，若随机选取的4点距离较近或接近于同一平面时，微小误差会造成关键点误差较大，因此选用MeanShift聚类方法得到大多数关键点候选的聚集位置，消除异常的关键点候选对于最终结果的影响。

但当物体采样点较多时，采用此方法计算量较大、时间代价较高。由于关键点均在物体表面，因此借助RGB-D图像得到的点云坐标，采用投票的方法在物体表面找到最合适的关键点。

首先，将物体表面的所有点均看作关键点 k 的一组候选点 $\left\{\boldsymbol{P}_{k,i}=(x_{k,i},y_{k,i},z_{k,i})|k=1,2,\cdots,N;i=1,2,\cdots,M\right\}$，根据其他点预测的距离 d_j 和已知坐标 $\left\{\boldsymbol{p}_j=(x_j,y_j,z_j)|j=1,2,\cdots,M-1\right\}$ 由式(10)计算得到该点与其他所有点 z 坐标的差值。

$$\Delta z'_{k,i,j}=\sqrt{d_j{}^2-(x_j-x_{k,i})^2-(y_j-y_{k,i})^2}。\tag{10}$$

而实际 z 坐标的差值可由 $\Delta z_{k,i,j}=\left|z_j-z_{k,i}\right|$ 计算得到，根据式(11)得到其他所有点对每个候选关键点的投票值。

$$v_{k,i}=\sum_{j=1}^{M-1}\left(\left|\Delta z_{k,i,j}-\Delta z'_{k,i,j}\right|>t\right),\tag{11}$$

式中：t 为误差阈值，若小于此阈值则认为 $\boldsymbol{p}_j$ 点对 $\boldsymbol{P}_i$ 候选点的投票为真。选择最大投票值 $v_{k,\max}$ 对应的点 $\boldsymbol{P}_{k,\max}$ 作为最后的关键点 k。

$$v_{k,\max}=\max\left\{v_{k,i}|i=1,2,\cdots,M\right\}。\tag{12}$$

3.2　关键点到位姿的计算

根据第3.1小节得到的相机坐标系下的关键点坐标 $\left\{\boldsymbol{p}_{\mathrm{key}\,i}\right\}_{i=1}^{N}$ 和已知的物体坐标系下的关键点坐标 $\left\{\boldsymbol{p}^*_{\mathrm{key}\,i}\right\}_{i=1}^{N}$ 采用最小二乘拟合的方法通过最小化 L_{fit} 损失[式(13)]实现物体位姿旋转矩阵 $\boldsymbol{R}$ 和平移矩阵 $\boldsymbol{T}$ 的计算。但在实际情况中可能会出现遮挡的情况，某些关键点在RGB-D图像中不可见，其关键点误差比可见关键点误差稍大，因此从所有关键点中挑选精度更高的关键点可以提升位姿估计准确度。由于采用最小二乘拟合方法求解物体位姿时需要至少3个点才能完成，因此可以从 N 个关键点中选择最合适的3个点参与解算。由于关键点的数量较少，可以采用穷举法，选择任意3点组合解算，共计算 C_N^3 种组合求得 C_N^3 个候选位姿结果。将已知的物体点云模型按每个候选位姿进行变换，寻找场景中物体点云与转换后的点云模型距离最近的点并一一对应，计算点与点之间的平均距离误差，选择

候选位姿中误差最小的位姿作为最终的位姿结果。

$$L_{\text{fit}}=\sum_{i=1}^{N}\left\| \boldsymbol{p}_{\text{key}\ i}^{*}-\left(\boldsymbol{R}\cdot\boldsymbol{p}_{\text{key}\ i}+\boldsymbol{T}\right)\right\|^{2}。\tag{13}$$

4　实验与分析

本实验基于PyTorch 1.11的环境，采用RTX3090显卡实现模型训练和测试。分别在LineMOD和YCB-Video两个公开数据集对所提方法进行验证，并与其他先进方法进行比较。

4.1　数据集

LineMOD数据集是一个包含13个低纹理对象的视频数据集。每个物体包含大约1200组数据，每组数据包括RGB-D图像、实例掩膜和标注的六自由度位姿。其中，图像大小为480 pixel×640 pixel。对此数据集位姿估计的主要挑战是场景杂乱、物体纹理弱和环境光照变化。

YCB-Video数据集包含21个形状和纹理各不相同的物体。它们分布在92个RGB-D视频中，每个视频场景中包含不同的物体组合，图像大小为480 pixel×640 pixel。数据集中包括标注的六自由度位姿和实例语义分割产生的掩码以及每个物体的CAD模型。数据集具有光照变化、图像噪声和遮挡等挑战。

4.2　评价指标

对位姿估计的评价通常采用ADD和ADDS两种指标。对于非对称物体，ADD指标计算物体模型分别通过预测值变换和真值变换后对应点对的平均距离，如式(14)所示。而对于对称物体，ADDS计算物体模型分别通过预测值变换和真值变换后距离最近的点对的平均距离，如式(15)所示。

$$D_{\text{ADD}}=\frac{1}{m}\sum_{\boldsymbol{v}\in\boldsymbol{o}}\left\|\left(\boldsymbol{R}\boldsymbol{v}_{o}+\boldsymbol{T}\right)-\left(\boldsymbol{R}^{*}\boldsymbol{v}_{o}+\boldsymbol{T}^{*}\right)\right\|,\tag{14}$$

$$D_{\text{ADDS}}=\frac{1}{m}\sum_{\boldsymbol{v}_1\in\boldsymbol{o}}\min_{\boldsymbol{v}_2\in\boldsymbol{o}}\left\|\left(\boldsymbol{R}\boldsymbol{v}_{1}+\boldsymbol{T}\right)-\left(\boldsymbol{R}^{*}\boldsymbol{v}_{2}+\boldsymbol{T}^{*}\right)\right\|,\tag{15}$$

式中：$\boldsymbol{v}$表示物体模型$\boldsymbol{o}$中的体素点；m表示点的总数；$\boldsymbol{R}$、$\boldsymbol{T}$表示预测的位姿；$\boldsymbol{R}^{*}$、$\boldsymbol{T}^{*}$表示位姿的真值。

对于YCB-Video数据集，本文按照文献[4]、[12]的评估方式，通过计算精度-阈值曲线下的面积，即随着距离阈值变化ADD(S)/ADD-S的变化曲线下的面积，后文称为AUC-ADD(S)。这里ADD(S)表示针对对称和非对称物体分别采用不同的指标，ADD-S表示对对称物体和非对称物体的综合评价。对于LineMOD数据集，按照文献[4]、[9]的评估方式，平均距离误差小于10%物体直径的预测被视为成功的预测，计算测试集中所有情况的预测精度，后文称为10%d-ADD(S)。

4.3　测试结果与分析

4.3.1　实验细节

所提网络模型中的特征提取融合网络为编码-解码结构。针对RGB图像的特征提取采用基于新型骨干模块的PSPNet。在点云的特征提取中，需要先对深度图随机采样12800个点实现深度图到点云的转换，再采用RandLA-Net实现点云的特征提取。在编码和解码的每层都通过shared MLP将两种特征逐点融合，共融合8次，得到最终的密集融合特征。再通过MLP分别实现语义分割、关键点距离预测和置信度预测。训练过程中采用小批量梯度下降法，单次批量设置为6，每个物体训练约30 epoch。语义分割和置信度预测模块的优化均采用Focal损失函数，关键点距离预测模块和分割中的中心点预测的优化采用L1损失函数，通过多任务损失函数来监督整个过程，损失函数如式(16)所示，其中，$w_1=w_3=1$、$w_2=w_4=2$。

$$L_{\text{all}}=w_1L_{\text{keypoints}}+w_2L_{\text{semantic}}+w_3L_{\text{center}}+w_4L_{\text{conf}}。\tag{16}$$

4.3.2　数据集测试结果

分别在LineMOD数据集和YCB数据集上进行测试。图6(a)和图6(b)分别为两数据集测试的可视化结果，图中物体覆盖的色块为将物体点云模型经过位姿结果变换并投影到图片的结果，覆盖程度越高表示估计位姿越准确。从图中可以看出，投影与物体重合度较高。

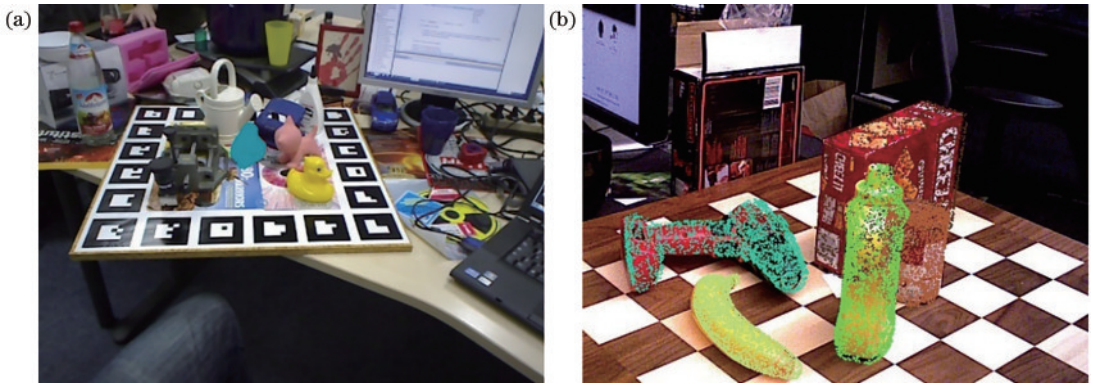

图6　可视化结果。(a) LineMOD数据集；(b) YCB-Video数据集

Fig. 6　Visualization results. (a) LineMOD dataset; (b) YCB-Video dataset

表 1 为所提方法与其他 4 种[4,9,12,5]具有代表性方法对 LineMOD 数据集中每类物体的测试结果。采用与其他 4 种方法相同的评价指标 10%d-ADD(S),将半径的 10% 作为阈值评价估计结果成功与否。eggbox 和 glue 两种对称物体采用 ADDS 指标,其他物体采用 ADD 指标。表 1 结果显示,所提方法相比于其他方法有较明显的优势,平均结果从对比方法中最好的 98.7% 提升到 99.8%。

表 1 LineMOD 数据集上的测试结果[10%d-ADD(S)]

Table 1 Test results on LineMOD dataset [10%d-ADD(S)]

unit: %

Object	PoseCNN	PVNet	DenseFusion	G2L-Net	Proposed method
ape	77.0	43.6	92.3	96.8	**98.7**
benchvise	97.5	99.9	93.2	96.1	**100.0**
camera	93.5	86.9	94.4	98.2	**99.9**
can	96.5	95.5	93.1	98.0	**100.0**
cat	82.1	79.3	96.5	99.2	**99.9**
driller	95.0	96.4	87.0	99.8	**100.0**
duck	77.7	52.6	92.3	97.7	**98.6**
eggbox	97.1	99.2	99.8	100.0	**100.0**
glue	99.4	95.7	**100.0**	**100.0**	99.9
holepuncher	52.8	82.0	92.1	99.0	**100.0**
iron	98.3	98.9	97.0	99.3	**100.0**
lamp	97.5	99.3	95.3	99.5	**100.0**
phone	87.7	92.4	92.8	98.9	**100.0**
mean	88.6	86.3	94.3	98.7	**99.8**

表 2 为所提方法与其他 3 种方法在 YCB-Video 数据集上每类物体的测试结果,由于 PVNet 未在此数据集上测试,因此未列出。采用与其他 3 种方法[4,12,6]相同的评价指标 AUC-ADD-S 和 AUC-ADD(S)计算精度-阈值曲线下的面积。其中:ADD-S 是将对称物体和非对称物体集成到同一评估中的指标,均采用 ADDS 指标计算;ADD(S)是对 bowl、wood block、large clamp、extra large clamp 和 foam brick 几种对称物体采用 ADDS 指标,其他物体采用 ADD 指标的结果。表 2 结果显示,ADD-S 和 ADD(S)两指标分别从 91.6%、84.3% 提升到了 94.1%、90.1%。

图 7 为在 YCB-Video 数据集几个复杂场景下的测试结果对比图。将所提方法与同样采用 RGB-D 作为数据源的 DenseFusion[12]进行对比。可以看出,所提方法在遮挡、背景干扰的复杂场景情况下仍表现良好,具有较强的鲁棒性。

表 2 YCB-Video 数据集上的测试结果[AUC-ADD(S)]

Table 2 Test results on YCB-Video dataset [AUC-ADD(S)]

unit: %

Object	PoseCNN		DenseFusion		GDR-Net		Proposed method	
	ADD-S	ADD(S)	ADD-S	ADD(S)	ADD-S	ADD(S)	ADD-S	ADD(S)
02 master chef can	83.9	50.2	95.3	70.7	**96.3**	65.2	95.1	**77.7**
03 cracker box	76.9	53.1	92.5	86.9	**97.0**	88.8	93.9	**89.4**
04 sugar box	84.2	68.4	95.1	90.8	**98.9**	**95.0**	96.5	94.4
05 tomato soup can	81.0	66.2	93.8	84.7	**96.5**	**91.9**	94.9	89.2
06 mustard bottle	90.4	81.0	95.8	90.9	**100.0**	92.8	96.8	**94.7**
07 tuna fish can	88.0	70.7	95.7	79.6	99.4	94.2	95.0	90.3
08 pudding box	79.1	62.7	94.3	89.3	64.6	44.7	**93.9**	**87.4**
09 gelatin box	87.2	75.2	**97.2**	95.8	97.1	92.5	97.0	**94.8**
10 potted meat can	78.5	59.5	89.3	79.6	86.0	80.2	**89.9**	**81.8**
11 banana	86.0	72.3	90.0	76.7	**96.3**	85.8	95.2	**91.1**
19 pitcher base	77.0	53.3	93.6	87.1	**99.9**	**98.5**	96.6	94.9
21 bleach cleanser	71.6	50.3	94.4	87.5	94.2	84.3	**95.0**	**90.6**
24 bowl	69.6	69.6	86.0	86.0	85.7	85.7	**88.1**	**88.1**
25 mug	78.2	58.5	95.3	83.8	**99.6**	**94.0**	96.8	92.5
35 power drill	72.7	55.3	92.1	83.7	**97.5**	90.1	95.6	**92.7**
36 wood block	64.3	64.3	89.5	89.5	82.5	82.5	**90.5**	**90.5**
37 scissors	56.9	35.8	90.1	77.4	63.8	49.5	**93.6**	**89.9**
40 large marker	71.7	58.3	**95.1**	**89.1**	88.0	76.1	95.0	85.3
51 large clamp	50.2	50.2	71.5	71.5	89.3	89.3	**93.3**	**93.3**
52 extra large clamp	44.1	44.1	70.2	70.2	**93.5**	**93.5**	88.2	88.2
61 foam brick	88.0	88.0	92.2	92.2	**96.9**	**96.9**	95.8	95.8
mean	75.9	59.9	91.2	82.9	91.6	84.3	**94.1**	**90.1**

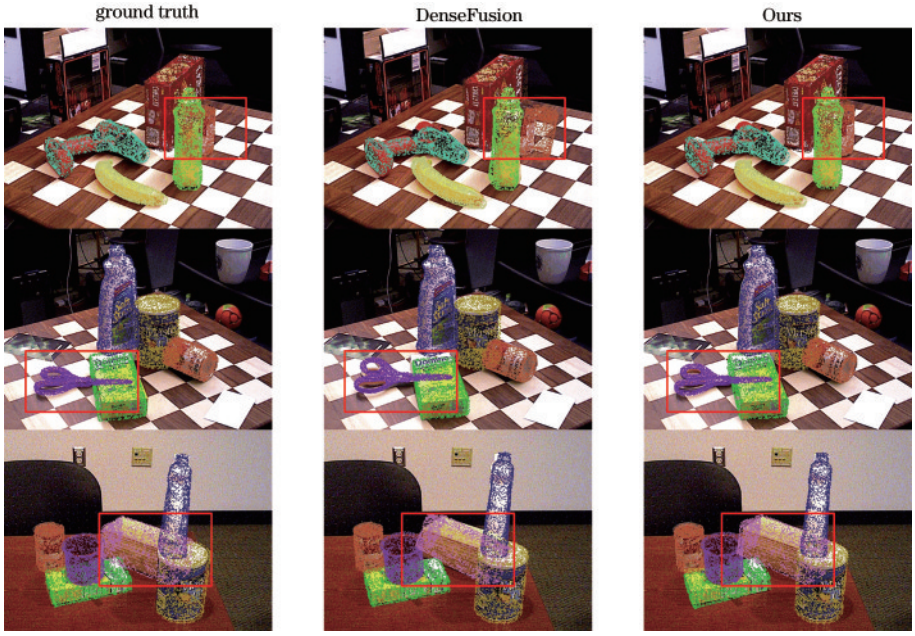

图 7　位姿估计结果对比

Fig. 7　Comparison of pose estimation results

4.3.3　消融实验

为验证所提方法的有效性，通过消融实验对各部分进行测试。表 3 为采用不同特征提取骨干模块的测试结果。表 4 为采用不同骨干模块时特征提取网络的参数量。结果显示，所设计的骨干网络相比于原 PSPNet 框架中的 ResNet 在精度提升的同时减少了 50% 的参数量，具有较好的性能。

表 3　不同骨干网络模块的精度比较

Table 3　Accuracy comparison of different backbone network modules　　unit: %

Dataset	ResNet	Proposed module
LineMOD	99.6	99.8
YCB-Video, ADD-S	92.1	94.1
YCB-Video, ADD(S)	86.5	90.1

表 4　不同骨干网络模块的特征网络参数量

Table 4　Feature network parameters of different backbone network modules

Module	ResNet	Proposed module
Parameters /MB	33.7	15.4

表 5 为其他条件不变时有无数据处理过程的结果对比，数据处理过程包括置信度低点的剔除以及误差较大关键点的剔除，表中数据均为 AUC-ADD(S)指标。结果表明，数据处理过程对于位姿估计准确度的提升有所帮助。

综上所述，在去掉某些环节的作用下，所提方法的位姿估计准确度依然高于其他方法，说明利用关键点距离量表征网络在物体位姿的估计任务中是有效且鲁棒的。

表 5　有无数据处理过程的结果对比

Table 5　Comparison of results with or without data processing　　unit: %

Dataset	Without data processing	With data processing
LineMOD	96.1	96.6
YCB-Video, ADD-S	93.5	94.1
YCB-Video, ADD(S)	88.3	90.1

5　结　　论

设计了一种基于关键点距离表征网络的物体位姿估计方法，通过网络输出距离量实现物体的位姿估计，提高网络泛化能力，从而提升位姿估计的精度。网络部分设计了一种基于新型骨干网络模块的特征提取网络，实现逐层逐点的特征融合，获得包含多层次纹理信息和几何信息的特征的同时大大减少参数量。基于融合特征实现基于聚类的语义分割、关键点距离预测以及置信度预测。置信度的预测可以剔除网络预测异常的点，提高位姿估计精度。在推理计算部分将得到的网络输出信息结合，针对物体采样点数量的多少采用不同的方法将网络输出的距离量转换为关键点坐标，再从关键点中选择 3 点通过最小二乘拟合得到最终的精确位姿。在 LineMOD 和 YCB-Video 数据集上的实验结果表明，与其他先进方法对比，所提方法在两数据集上的准确度分别提升了 1.1 个百分点和 5.8 个百分点，同时相比于原 PSPNet 框架中的 ResNet 减少了特征提取网络 50% 的参数量，节省了时间。所提方法可以实现准确稳定的物体位姿估计。

参 考 文 献

[1] 杨步一, 杜小平, 方宇强, 等. 单幅图像刚体目标姿态估计方法综述[J]. 中国图象图形学报, 2021, 26(2): 334-354.
Yang B Y, Du X P, Fang Y Q, et al. Review of rigid object pose estimation from a single image[J]. Journal of Image and Graphics, 2021, 26(2): 334-354.

[2] Sahin C, Carcia-Hernando G, Sock J, et al. A review on object pose recovery: from 3D bounding box detectors to full 6D pose estimators[J]. Image and Vision Computing, 2020, 96: 103898.

[3] Du G G, Wang K, Lian S G, et al. Vision-based robotic grasping from object localization, object pose estimation to grasp estimation for parallel grippers: a review[J]. Artificial Intelligence Review, 2021, 54(3): 1677-1734.

[4] Xiang Y, Schmidt T, Narayanan V, et al. PoseCNN: a convolutional neural network for 6D object pose estimation in cluttered scenes[EB/OL]. (2017-11-01)[2022-05-06]. https://arxiv.org/abs/1711.00199.

[5] Chen W, Jia X, Chang H J, et al. G2L-net: global to local network for real-time 6D pose estimation with embedding vector features[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA. New York: IEEE Press, 2020: 4232-4241.

[6] Wang G, Manhardt F, Tombari F, et al. GDR-net: geometry-guided direct regression network for monocular 6D object pose estimation[C]//2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 20-25, 2021, Nashville, TN, USA. New York: IEEE Press, 2021: 16606-16616.

[7] Rad M, Lepetit V. BB8: a scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth[C]//2017 IEEE International Conference on Computer Vision, October 22-29, 2017, Venice, Italy. New York: IEEE Press, 2017: 3848-3856.

[8] Zhao Z L, Peng G, Wang H Y, et al. Estimating 6D pose from localizing designated surface keypoints[EB/OL]. (2018-12-04)[2022-05-08]. https://arxiv. org/abs/1812.01387.

[9] Peng S D, Zhou X W, Liu Y, et al. PVNet: pixel-wise voting network for 6DoF object pose estimation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 44(6): 3212-3223.

[10] Qi C R, Liu W, Wu C X, et al. Frustum PointNets for 3D object detection from RGB-D data[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE Press, 2018: 918-927.

[11] 陈海永, 李龙腾, 陈鹏, 等. 复杂场景点云数据的6D位姿估计深度学习网络[J]. 电子与信息学报, 2022, 44(5): 1591-1601.
Chen H Y, Li L T, Chen P, et al. 6D pose estimation network in complex point cloud scenes[J]. Journal of Electronics & Information Technology, 2022, 44(5): 1591-1601.

[12] Wang C, Xu D F, Zhu Y K, et al. DenseFusion: 6D object pose estimation by iterative dense fusion[C]//2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 15-20, 2019, Long Beach, CA, USA. New York: IEEE Press, 2019: 3338-3347.

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

[14] Hu Q Y, Yang B, Xie L H, et al. RandLA-net: efficient semantic segmentation of large-scale point clouds[C]//2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 13-19, 2020, Seattle, WA, USA. New York: IEEE Press, 2020: 11105-11114.

[15] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition[C]//2016 IEEE Conference on Computer Vision and Pattern Recognition, June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE Press, 2016: 770-778.

[16] Huang G, Liu Z, van der Maaten L, et al. Densely connected convolutional networks[C]//2017 IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, HI, USA. New York: IEEE Press, 2017: 2261-2269.