

中国科学技术大学

硕士学位论文



面向智能服务机器人的多模态物体 识别与定位技术

作者姓名： 张泽坤

学科专业： 计算机应用技术

导师姓名： 陈小平 教授

完成时间： 二〇一八年三月二十六日

University of Science and Technology of China
A dissertation for master's degree



**Multi-Modula Object Recognition
and Localization Technologies
Aiming Intelligent Service Robot**

Author: Zekun Zhang

Speciality: Applied Computer Technologies

Supervisor: Prof. Xiaoping Chen

Finished time: March 26, 2018

中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文，是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外，论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名：_____

签字日期：_____

中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一，学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权，即：学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅，可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

保密的学位论文在解密后也遵守此规定。

公开 保密（____年）

作者签名：_____

导师签名：_____

签字日期：_____

签字日期：_____

摘要

智能服务机器人已经走进了普通民众的生活，并将在未来扮演越来越重要的角色。在工作过程中，机器人将会面临多种多样的物体定位和识别挑战，开发实用的视觉算法和技术是服务机器人研究领域的核心研究方向之一。本文针对智能服务机器人需要完成的典型任务，提出了一套基于深度卷积神经网络的物体分类器，和一套基于多三维摄像头的物体定位与识别系统。本文主要做出了如下创新：

服务机器人需要从很少量的训练数据中训练识别分类器。传统的基于手工构建的特征的方法鲁棒性不强；而在小数据集上从头训练深度卷积神经网络会发生严重的过拟合现象。为将深度学习应用在服务机器人系统中，本文结合了迁移学习和数据增强的方法，并采用了学习率规划，成功使用少量数据训练了较大规模的神经网络分类器。该分类器具有一定的鲁棒性，可以在变化的环境中工作，并在机器人上实时运行。为获得待识别物体的图片，本文还利用了三维摄像头提供的点云信息对物体图像进行了有效的自动分割。实验中，分类器的精度超过了之前的分类器。

三维摄像头可以提供比二维相机更丰富的视觉信息，其中很多对于服务机器人的物体操作过程十分有用，但是在小尺度下获得物体精确而完整的三维结构并不容易。本文使用了多个预先校准好变换关系的三维摄像头从不同角度对物体进行观测，获得了较为精确和完整的物体点云。从点云出发，实现了精度较高的物体定位和特征提取。随后搭建了实验平台对定位和识别结果进行了验证实验。通过系统分析实验结果，说明了本文的方法可以在一定程度上补偿三维摄像头的系统误差。实验中机械臂对各种物体进行了成功的操作，表明此方法的精度可以满足服务机器人的要求。

关键词：物体识别；卷积神经网络；立体视觉；物体定位；服务机器人

ABSTRACT

Intelligent service robots, which have already made their debut in people's daily life, will play more important roles in the future. In their service, robots face various object recognition and object localization challenges, which demand cutting-edge research towards practical algorithms and technologies in robot vision. Aiming typical tasks of intelligent service robots, a convolutional neural network based object classifier and a multiple stereo cameras based vision system are proposed. This work features the following two main innovations.

Compared to object recognition competitions such as ImageNet, service robots need to train their object classifier from very limited training images. Conventional hand-crafted features based methods lack generalizability, while training deep neural networks on limited training data will result in severe over-fitting. To overcome such difficulties and apply deep learning into service robot systems, transfer learning, data augmentation, and learning rate scheduling methods are used to successfully train the neural network. The object classifier based on this network can run on service robots in real-time, and can generalize to unprecedented environments. Point clouds from stereo cameras are used to segment the images of objects from the background. Experiments illustrate that this classifier exceeds its predecessors in classification accuracy.

Stereo cameras can collect more information than 2D cameras, which is potentially benefiting to service robots. Yet obtaining integrated and precise 3D models of given objects at relatively small scale has been proven to be a non-trivial job. Several pre-aligned multiple stereo cameras are used to observe the object at different viewpoints, and a complete 3D model can be constructed. Then the object's location and 3D features can be calculated. A experiment platform is built to test the efficiency of this method. Analysis of the results shows that the errors of stereo cameras are compensated in certain degree. An arm is used to operate several types of objects according to calculated location and features. The success of such operation proves that this method can meet the requirements of service robots.

Key Words: Object Recognition; Convolutional Neural Network; Stereo Vision; Object Localization; Service Robot

目 录

第 1 章 绪论 ······	1
1.1 计算机视觉的研究背景 ······	1
1.2 基于二维图像的物体识别方法 ······	2
1.3 三维物体识别方法 ······	4
1.4 智能服务机器人的发展现状 ······	5
1.5 本文主要工作与创新点 ······	6
第 2 章 基于小数据集的深度卷积神经网络物体分类器 ······	9
2.1 深度卷积神经网络基本原理 ······	9
2.2 微调与数据增强 ······	12
2.3 方法概述 ······	13
2.3.1 神经网络结构 ······	13
2.3.2 训练数据集的采集与处理 ······	16
2.3.3 神经网络的训练过程 ······	18
2.4 实验环境 ······	19
2.4.1 实验所用相关软件库简介 ······	19
2.4.2 实验流程 ······	21
2.5 实验结果与分析 ······	22
2.5.1 神经网络的训练过程 ······	22
2.5.2 在机器人上的运行结果 ······	23
2.6 本章小结 ······	25
第 3 章 多摄像头物体定位与识别系统 ······	27
3.1 三维视觉传感器简介 ······	27
3.2 多摄像头立体视觉系统 ······	29
3.3 物体的定位与三维特征的提取 ······	30
3.4 实验平台的搭建 ······	34
3.5 实验结果与分析 ······	35
3.5.1 物体定位实验 ······	36
3.5.2 三维特征提取实验 ······	38
3.5.3 机械臂精度评估 ······	40
3.5.4 物体操作实验 ······	41
3.6 本章小结 ······	41

目 录

第 4 章 总结与展望.....	43
4.1 本文工作总结	43
4.2 未来工作展望	44
参考文献	45
致谢	49
在读期间发表的学术论文与取得的研究成果	51

图 目 录

2.1 函数的卷积示意图	9
2.2 不同的卷积核的作用	10
2.3 随机梯度下降示意图	12
2.4 对图像的旋转扩充	13
2.5 VGG-16 网络结构	13
2.6 三种常见激活函数的图像	14
2.7 修改后的 VGG-16 网络结构与中间层输出	15
2.8 VGG-16 网络的前两个卷积模块输出的特征图	17
2.9 可佳机器人及其视觉传感器	17
2.10 可佳机器人的 ROS 节点示意图	20
2.11 实验所用物体	22
2.12 神经网络的训练过程	23
3.1 微软 Kinect 传感器示意图	27
3.2 结构光的工作原理示例	28
3.3 本文使用的标签板	31
3.4 主成分提取示意图	32
3.5 实验平台示意图	34
3.6 动作捕捉系统使用的相机和标记物	35
3.7 视觉系统校准过程示意图	36
3.8 物体定位实验所用的物体	36
3.9 物体定位实验结果	37

图 目 录

3.10 物体定位实验测量结果的分布情况	39
3.11 不同物体的三维特征提取结果	40
3.12 机械臂对物体进行抓取	42

表 目 录

2.1 不同概率分布的信息熵	22
2.2 不同环境下机器人运行分类器的识别率	24
3.1 不同范围内的分布比例	38

算 法 目 录

2.1 训练图片的预处理过程	18
3.1 物体的定位与特征计算过程	33

第1章 绪 论

1.1 计算机视觉的研究背景

人工智能研究的着眼点和最高目标是设计并实现具有类似于或者超过人类智能的智能体。智能体应该具有一个关键特征：自主地具有对环境进行感知并做出适当的反应。智能体在运行过程中需要在决策单元中利用感知部分收集到的数据建立自己对环境的认知模型，并根据一定的规则通过执行部分做出适当的反应^[1]。最典型也为大众所熟知的智能体系统是各种智能机器人。典型的智能机器人系统由感知、运动、控制等部分有机结合而成。其中感知部分常具有激光探测器、摄像头、麦克风、超声波定位仪、陀螺仪、触觉传感器等传感器，其作用是获取环境的相关信息。这一部分相当于人类的眼睛、耳朵、鼻子、皮肤等感受器官。运动部分用于和环境的交互以及机器人自身的移动，如驱动机器人的传动轮、用于操作物体的机械臂等。该部分相当于人类的骨骼和肌肉。控制部分位于机器人系统的中心位置，用于处理感知部分传来的信息，并做出适当的决策，转换为控制指令发送给运动部分。该部分相当于人类的中枢神经系统。随着传感器和动力机械技术的不断发展，机器人可以以更高的精度和广度感知周围环境的信息，同时可以以更精细的方式与环境交互。这就对机器人的控制部分提出了更高的要求。例如，自动驾驶汽车需要根据传感器感知的路况沿着车道行驶，并且可以避让其他车辆和行人；家庭服务机器人需要理解人类的语音指令，并在复杂的环境中完成各种任务。

在智能体系统中，感知部分是整个决策过程的起点，能否准确和全面地对环境进行感知直接决定了后续过程的可靠性和准确性。对于人类而言，视觉可能是所有感觉中最为重要的一种。用于处理视网膜发送到大脑的视觉信息的视觉皮层极为发达，占据了大脑的很大一部分^[2]。在机器人领域，机器人视觉同样是研究的重点。机器人视觉或计算机视觉一般指利用可见光（波长在 400 nm ~ 800 nm 之间的电磁波）信息对环境进行感知，并对信息进行处理的过程。机器人视觉研究涵盖了从机器人感知部分的视觉传感器，到控制部分的处理硬件以及处理算法的整个流程。计算机视觉系统一般使用和结构与人眼类似的摄像机成像来感知环境。典型的摄像机通过光学折射和反射系统将三维环境投影在二维的成像平面上，该平面上的成像元件排列成一个阵列，每个元件各自接收光照信息形成一个像素，所有像素按顺序组合即可构成完整的二维图像。在二维图像不能提供足够信息的场合，采用双目视觉、结构光、飞行时间等技术可以得到更为丰富的三维点云信息^[3]。和声音等信息相比，视觉信息的维度与复杂度要高得多。

这导致了处理视觉信息需要更多的计算，同时视觉信息也更难进行压缩与特征提取。

机器人的视觉系统需要从视觉信息出发对环境的本质进行理解。一般而言，环境被看成是由多个相互独立的物体组合而成。为了理解环境，机器人往往需要在完整的二维图像或三维点云中检测和识别出感兴趣的物体。传统上，首先需要对视觉信息的整个视野进行分割，然后对分割出来的每个候选提取各种视觉特征，然后与已知物体的特征进行匹配，其中匹配度高的候选即被看作识别出的物体。大部分实际应用场合下环境复杂多变，一种方法需要在光照、背景、尺度发生变化的情况下有效工作才能在现实中使用。开发精确度高，鲁棒性强的物体检测和识别算法是整个机器人视觉研究中最为活跃的研究领域之一，相关工作层出不穷。本文接下来介绍基于二维图像的物体检测与识别的研究现状，以及基于三维信息的视觉研究情况。

1.2 基于二维图像的物体识别方法

图像识别的研究历史几乎与现代计算机科学的发展历史一样长。早在 1963 年，Roberts 等人就提出了从二维图像中识别并重建出形状规则的三维几何体的方法^[4]。该方法使用算法从图像中检测几何体在背景中形成的边缘，从而得到该几何体的投影信息。然后通过投影信息与已知的几何体三维模型就可以计算几何体的三维变换，从而识别出几何体并重构出物体的取向和位置等信息。该方法是物体识别领域最早的工作之一，对于后续的研究有极大的启发性和指导意义。但是该方法只能适用于三维模型已知的简单几何体，且要求物体和背景很容易区分开来，因而几乎无法应用在实际场景中。1973 年，Haralick 等人提出了一种基于图像中的纹理特征进行图像分类的方法^[5]。纹理可以看成图像的某个局部中不同颜色的二维分布情况，从纹理中可以用一定的算法计算特征向量。从一张二维图像计算各个部位的纹理特征向量，然后根据这些特征向量就可以对图像进行分类。基于局部特征的图像分类方法是图像识别和分类领域的主流研究方向之一，研究者提出了很多成功的特征计算算法，典型的有 1999 年提出的 SIFT 特征^[6] 和 2006 年提出的 SURF 特征^[7]。

利用这些方法进行图像识别时，首先需要从图像中计算得到一定数量的利于计算特征向量的关键点，然后计算每个关键点对应的特征向量，这些特征向量可以和已知图像数据库进行匹配，从而得到未知图像的类别以及取向等信息。这些基于局部特征的方法在很多不同的图像识别领域应用中取得了很大的成功。在卫星图像处理，文字识别等领域已经有非常成熟的商业软件。但是基于特征的方法因为其原理的限制存在一些局限性。这些方法往往依赖于专家构造的特征

向量计算算法。这些方法在有限、可预知的输入下表现良好，但在变化或未知的环境下往往效果不佳。例如具有红色花纹的白色物体在红光照射下各个部分都会显示为红色，使得原本的纹理消失。表面有突起的物体在侧光照射下会产生阴影，从而改变纹理特征。在弱光下，现有的传感器会在图像中产生很多噪点，影响识别结果。如果希望算法在这些变化环境中仍然表现良好，就需要设计更为稳健的特征，以及对传感器生成的图像进行一定的预处理。

另一种用于图像分类的方法是人工神经网络。人工神经网络是一种受动物神经系统启发而设计的计算模型，由一系列互相连接的人工神经元构成，每个连接都具有一定的权值。数值在神经元之间传递时，将和这些权值相乘，神经元的连接方式和连接的权值共同决定了模型的计算结果。神经元之间往往还会加入激活函数，使得模型具有非线性特征。一些输入神经元接受数值输入，数值经过连接在网络中进行传播计算后，从另一些输出神经元输出。在神经网络的结构和输入保持不变时，输出结果随着权值的改变而变化。通过调整权值，就可以让神经网络模拟某个函数的表现。理论上，如果神经网络足够复杂，通过调整连接的权值，就可以模拟任意函数。1986年，Rumelhart、Hinton 和 Williams 等人提出了一个实用的反向传播算法^[8] 用于在已知的输入输出对上训练神经网络，经过一定的算法对网络中连接的权值进行调整后，神经网络将在给定的输入上计算出希望的输出。此后神经网络被广泛用于各种监督学习任务当中，如果将待识别的图像作为输入，分类结果作为输出，就可以训练用于图像分类的神经网络。在大量数据上训练的人工神经网络的中间层自动的具备从输入提取一定的特征的能力。这些特征提取过程是通过大量的训练数据得到的，而不是由专家手工构建。如果训练数据的量足够大且具有代表性，训练得到的神经网络就可以用在未知数据上取得较好的效果，即具有较好的泛化能力。

在计算机视觉领域，卷积神经网络因为更符合图像的二维结构被广泛应用。卷积神经网络的卷积层中，卷积核综合计算了图像的某个局部的特征。1998年，LeCun 等人提出了一种使用卷积神经网络识别手写数字的方法^[9]。该网络，称为 LeNet-5，接受 32×32 分辨率的灰度图像作为输入，经 3 个卷积层和 2 个降采样层后，再经过三个全连接层，最后输出一个 10 维向量，向量的每个分量代表了输入属于每个字符的概率。LeNet-5 具有约 6×10^4 个可训练的参数。实验表明，训练后的卷积层可以有效的从输入图片中提取边缘、拐角等特征。LeNet-5 的识别准确率超过了当时的其他分类器，并成为后来几乎所有卷积神经网络的雏形。由于神经网络的性能取决于参数数量和训练集大小，训练一个能准确识别高分辨率图像的神经网络需要极大的计算量。很长一段时间内因为计算硬件条件的限制，神经网络的规模十分有限，限制了它的实际应用。

训练神经网络需要训练集的大小和网络规模相匹配，若训练集相对较小，网

络就会“记住”所有的训练集，而不是提取特征，从而产生过拟合。过拟合的网络无法有效识别训练集中不存在的数据。有效的神经网络需要具有较大的规模，相应的也需要大量的训练数据进行训练。进入21世纪后，计算硬件的发展释放了神经网络的巨大潜力。神经网络中大部分计算可以归结为矩阵乘法和加法，而通用图形处理器可以以很高的并行度进行这些计算，从而将神经网络的训练速度提高数十甚至数百倍。2012年提出的AlexNet^[10]在图像分类比赛ImageNet中以巨大的优势获得第一名，证明了在大数据集上深度卷积神经网络相对于传统方法的优势。AlexNet使用了5个卷积层和3个全连接层，使用ReLU作为激活函数，并采用了随机将中间层神经元输出清零的方法^[11]减少了过拟合。为了进一步减少过拟合，训练图片在输入网络前进行了随机裁剪和翻转，扩充了训练集的大小。由于网络参数很多，需要在两块GTX 580显卡上运行。此后深度卷积神经网络成为图像识别和分类领域的主流方法。后续的模型，如VGG^[12]和GoogleNet^[13]，都采用了更多的卷积层。2015年ImageNet比赛的第一名ResNet^[14]更是采用了上百层卷积。由于层数多的网络会面临传播过程中信息衰减的问题，ResNet中采用了很多“短路”连接，跳过中间层直接将前部的输出输入到后部。一些规范化方法^[15]也被广泛使用以取得更好的识别精确度。现在基于深度卷积神经网络的模型已经可以在特定数据集上达到甚至超过人类的识别精度，并已经被广泛应用在手写字符识别、场景理解、人脸识别等领域的商业软件中。

1.3 三维物体识别方法

由于实际中的大部分物体都不是能够自发光的光源，需要反射照明光才能被人眼或视觉传感器探测到，导致物体的像与光照环境密切相关。不同的亮度和颜色的光照下同一个物体会呈现出不同的外观。这使得在特定环境下收集到的图片作为训练集训练得到的分类器无法应用在其他场景中。同一个物体在不同视角下也会形成不同的像，这些像之间往往极为不同。此外，物体的成像有近大远小的特点，有效识别需要提取不随尺度变化的特征。这些都是二维成像技术原理产生的限制。现有的技术可以用单个传感器得到视野的彩色点云图像，相对于二维图像，三维点云受环境光照的影响较小。点云 $\{p_i\}$ 由一定数量的像素点 p 构成，每个像素点除了具有颜色信息外，还具有该点在三维空间中的位置坐标的信息，即 $p = (r, g, b, x, y, z)$ 。相对的，二维图像的像素点只具有颜色和像素位置信息，即 $p = (r, g, b, u, v)$ 。同一个物体在不同的距离和视角下保持了不变的形状。另外，大部分立体视觉传感器具有主动发光单元，因此在不同的光照环境下仍然可以得到相似的结果。这就使得三维视觉相对于传统的二维视觉受环境和观测条件影响较小。除了稳定性外，三维点云相对于二维图像具有更加丰富的信

息，这为后续处理提供了更多的可能性和便利性。例如，如果需要从二维图像中精确地分割出感兴趣的物体，需要根据颜色信息寻找物体的边缘，但是在一些光照条件下边缘将会十分模糊。在三维点云中，物体和周围环境的边缘则十分容易检测。二维图像中的局部特征一般基于颜色纹理，而三维点云中除了基于颜色纹理的特征外，还可以计算法向量等基于局部形状的特征，这些特征可以提高识别的精确度。将这些特征点的位置和已知模型进行匹配，不但可以识别物体的种类和位置坐标，还可以计算出物体的三维取向信息，这可以为后续的控制提供更多的信息。近十年来，微软 Kinect，英特尔 RealSense 等廉价三维视觉传感器的出现，使得三维物体识别成为机器人视觉领域中一个热门的研究方向。

基于三维点云的物体的识别主要有基于全局特征的方法和基于局部特征的匹配方法。典型的基于全局三维特征的方法有视角特征直方图^[16]，利用该直方图可以直接计算一个点云的三维特征。由于此特征是基于点云全体的，在计算前需要将点云中感兴趣的部分分割出来。基于局部的三维特征类与二维图像的局部特征类似，典型的如三维 SIFT 描述子^[17]。点云库^[18] 中提供了大量基于局部特征的形状匹配算法，这些算法大都计算点云中点的局部特征，然后与已知模型中的点进行匹配。

三维点云提供了比二维图像更多的信息，所以相关处理也需要更多的计算量。三维特征的计算与匹配都需要比二维图像更多的计算，在单线程计算架构下整个识别过程难以做到实时运行。另外，由于硬件的发展历史相对较短，三维视觉的精度相对二维视觉较差。对于主流的三维视觉传感器来说，点云中位于视野中心的点的位置坐标误差约为几毫米，而视野边缘的位置误差可达一厘米以上。这就使得视野边缘的物体的形状发生了畸变，局部特征的计算也会出现问题。如果物体被遮挡，其三维特征也会出现较大的偏差。

1.4 智能服务机器人的发展现状

近年来，服务机器人的发展令人瞩目，其中很多已经走进了大众的生活当中。其中，各种扫地机器人因为结构简单、成本低廉已经成为较为常见的家用电器。利用水平面内的二维激光信息，扫地机器人可以对家庭环境建立平面地图，然后通过路径规划算法进行清扫。结构更为复杂用途更为多样的服务机器人也在不断开发当中。其中，中国科学技术大学多智能体实验室于 2008 年开始开发的可佳智能家庭服务机器人是国内最早的家庭服务机器人之一。作为一款通用型机器人，可佳机器人集成了机器人研究领域各个方向的研究成果，力图具有完成一般任务的通用服务能力。在一年一度的 RoboCup 机器人世界杯的 @Home 分组比赛中，可佳机器人多次取得前三名的成绩，并在 2014 年获得冠军。

可佳机器人主要由定位导航、语音交互、视觉识别等模块构成。其中定位导航模块使用水平面内的二维激光探知周围环境，并可以自主建立地图。在操作人告知机器人地图各个部分的作用后，机器人就可以在不同地点间自主移动，同时避让路径中出现的家具、人等障碍。若环境发生了变化，机器人也可以在运动过程中对地图做相应的修改。语音交互模块可以接受操作人的语音指令并转化为文本，该文本将被自然语言处理模块分解成一系列任务，交由机器人其他模块执行。视觉识别模块由多个摄像头构成，具有对人体的检测和跟踪，对物体的识别和定位，以及识别条码、二维码、文本等各种功能。

家庭服务机器人的很大一部分任务都涉及到了物体的操作，例如为操作人拿来某个特定物体。物体的操作涉及到了识别、定位、抓取这一系列操作过程，其中每一步的成功都依赖于前一步的成功。机器人的视觉系统首先需要在视野中寻找并识别出感兴趣的物体，然后确定该物体对于机器人的相对位置。为了保证后续的抓取操作的成功率，除了物体的位置外，还需要知道物体的尺寸、形状、取向等信息。对这些信息的有效提取一直是可佳机器人开发过程中的一项核心任务。家庭服务机器人工作的环境虽然不像室外环境那样复杂多变，机器人仍然需要处理很多变化的场景。例如，同一个物体在冰箱中、客厅的桌子上、卧室的床头柜上中会受到不同的光照，导致其颜色和纹理发生一定的变化。由于环境的限制，机器人往往只能看到物体的一部分信息，而不能看到全貌。另外每个家庭的环境都是不同的，这就导致很多依赖于海量数据进行训练的方法无法直接应用。这些都要求可佳机器人的视觉系统需要综合一切已知信息并做出当前信息下最为可靠的判断。

1.5 本文主要工作与创新点

本文主要面向智能服务机器人对于物体识别和定位提出的典型需求，分别提出了基于二维图像信息的物体识别方法和基于三维点云信息的物体定位识别方法。第1章为绪论，介绍了计算机视觉领域物体识别方向的研究背景和发展历史，并说明了本文的研究平台，智能服务机器人的特点和特殊需求。

第2章提出了基于小数据集的深度卷积神经网络物体图像分类器，此分类器被设计为用于家庭服务机器人的工作场景中。在ImageNet等图像分类比赛中，待识别物体分为上千类，每一类都有上万个训练图像。而家庭服务机器人面对的任务是从非常有限的数据集中训练分类器。这是由于家庭环境中的物体种类十分多样，其中既有其他场景中常见的物体，如产量较大的生活用品；也有十分特别的物体，例如定制的小家具和饰品，这类物体几乎不会在其他场景中出现。所以用网络爬虫收集图像数据进行训练是不现实的。另外，家庭环境中需要对物体

做十分精细的识别。基于这些特点，可佳机器上曾经采用基于图像整体和局部特征的模板匹配方法来识别物体。但是实践表明这种方法费时费力，而且泛化性能不佳。因此，本文提出了基于深度卷积神经网络的方法提高可佳机器人的物体识别表现。如果采用直接从头训练网络的方法，会产生非常严重的过拟合。为了解决这个问题，本文采用了迁移学习的思路，现在大数据集上训练网络，然后改变网络结构，并在小数据集上进行微调，从而得到最终的分类网络模型。本文还从经验和理论分析出发，设计了网络训练的方法。另外，本文还在数据集上采用了多种不同的数据增强方法，以达到一定程度上扩充数据集的目的。由于机器人的控制计算机一般只具有有限的计算能力，考虑到实际应用场景，本文对网络结构进行了精简，以满足实时运行的要求。本文还采用了三维视觉和二维视觉相结合的方法，将待识别的物体从背景中分割出来。为检验分类器的效果，本文将分类器移植到可佳机器人的视觉系统中并在实际家庭环境中进行了测试。实验证明此分类器的精度和泛化能力超过了基于特征的分类器，运行速度也可以满足应用场景的要求。

第3章提出了一套基于多个摄像头的三维物体识别定位方法，此方法可以应用在通用服务机器人或物流分拣机器人的工作场景中。现有的机器人操作物体大多基于二维图像，通过一定的方法进行校准后，可以提供物体的位置和大小等十分有限的信息。实际中为了针对不同物体的特点进行有效的物体操作，需要物体的精确形状和取向等信息。使用机器学习的方法可以从二维图像中提取这些信息，但是这些方法需要大量训练数据。更直接的方法是从三维点云直接计算这些信息。三维点云是对于连续的物体表面的离散采样，基于点云的算法的有效性取决于该采样的质量。单个三维摄像头同一时刻只能收集到物体某个侧面的信息，使得各种视觉算法的结果出现很大的偏差。对多个三维摄像头进行校准后，可以将各自的三维点云合并到一起，构成物体的完整点云。从完整点云出发，可以计算物体准确的整体三维特征和局部三维特征。这些特征可以提高机器人操作物体的智能度和精确性。本文将机械臂和摄像头系统相结合搭建了实验平台，对提出的方法进行了验证，证明了该方法的有效性和稳定性。多摄像头视觉系统可以以较高的精度完成对多个不同类型物体的定位和特征提取，其结果符合人类经验对物体的认知。机械臂根据位置和特征可以对物体做出准确而智能的操作。

第4章对本文研究的优势和不足进行了总结，并展望了未来的工作计划。本文主要整合了机器人视觉领域的多项技术，并做了适当的修改和改进，以适用于智能服务机器人的应用场景。结合最新的计算机视觉技术和机器人本身的物理模型，可以对本文的算法做进一步的改进和提高。

第2章 基于小数据集的深度卷积神经网络物体分类器

2.1 深度卷积神经网络基本原理

深度卷积神经网络是现有的最有效的图像识别工具之一，其核心操作是二维图像上的卷积运算。二维图像上的卷积是一元函数卷积运算的推广。卷积是一种“滑动平均”操作，数学上，对于两个可积函数 $f(x)$ 和 $g(x)$ ，其卷积 $(f * g)(x)$ 定义为

$$(f * g)(x) = \int_{-\infty}^{+\infty} f(\tau)g(x - \tau)d\tau \quad (2.1)$$

卷积的值代表了两个函数相互滑动时图像与 x 轴之间所夹面积之间重叠部分的大小，如图 2.1 所示。

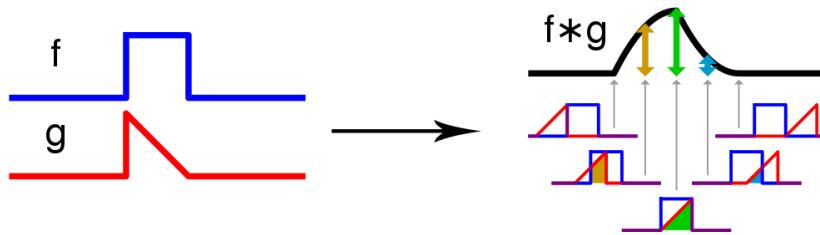


图 2.1 函数的卷积示意图

图像的卷积是上述概念在二维离散函数上的推广。如果将二维图像看作一个二维矩阵，以 4×4 分辨率的单通道图像为例，该矩阵可以看作定义在二维平面上的离散函数 $p(x, y)$ 。另有一个 3×3 的矩阵，称为卷积核，看作二维函数 $k(x, y)$ 。它们的卷积即为 $k(x, y)$ 在 $p(x, y)$ 上滑动到某一位置时对应元素乘积之和。一个例子如(2.2)所示。 3×3 卷积核在 4×4 图像上滑动，每次覆盖图像上 3×3 大小的一部分，如图中用红色标记的部分所示。此时卷积核中的每个元素与图像上被覆盖部分的对应元素相乘，乘积之和构成了卷积结果的一个元素。卷积核滑动到 4 个可能的位置，最终得到 2×2 尺寸的卷积结果。例子中采用的卷积核是对称的，得到的卷积结果保留了原图片的部分特征，即像素值从左到右，从上到下递增。

1	2	3	4
2	3	4	5
3	4	5	6
4	5	6	7

*

-1	1	-1
1	1	1
-1	1	-1

→

3	4
4	5

(2.2)

上述卷积操作是所谓“可行 (Valid)”方式的卷积操作，即卷积核滑动过程中完全位于原图像内部。此时卷积的结果尺寸小于原图像尺寸。如果将原图像扩展，除了像素点之外的部分看成值为 0 的像素，就是所谓“扩展 (Padding)”的卷积操

作, 如(2.3)所示。这种情况下, 原图片中像素的递增关系没有保留下, 但是沿对角线两侧对称的特征还保留着。如果卷积核每次滑动一个像素, 则扩展的卷积操作的结果尺寸与原图像尺寸相同。同时可以看到此时的卷积结果已经包含了前述可行方式进行卷积的结果, 图中用加粗字体标记, 而边缘部分的结果有一定的失真。此种方式进行卷积时结果的尺寸更容易计算, 所以在卷积神经网络模型中更常用。因为最终的结果中只有边缘的像素发生了一定的失真, 当图片尺寸较大时, 该影响是可以忽略的。

$$\begin{array}{|c|c|c|c|} \hline 1 & 2 & 3 & 4 \\ \hline 2 & 3 & 4 & 5 \\ \hline 3 & 4 & 5 & 6 \\ \hline 4 & 5 & 6 & 7 \\ \hline \end{array} * \begin{array}{|c|c|c|} \hline -1 & 1 & -1 \\ \hline 1 & 1 & 1 \\ \hline -1 & 1 & -1 \\ \hline \end{array} \rightarrow \begin{array}{|c|c|c|c|} \hline 2 & 3 & 5 & 8 \\ \hline 3 & \mathbf{3} & \mathbf{4} & 11 \\ \hline 5 & \mathbf{4} & \mathbf{5} & 13 \\ \hline 8 & 11 & 13 & 14 \\ \hline \end{array} \quad (2.3)$$

卷积操作提取了图像中一个局部的信息, 这些信息构成了新的二维矩阵 $p*k$, 该矩阵称为图像在卷积核操作卷积操作下的特征图。不同的卷积核起到不同的作用, 如图 2.2 所示。其中原图片为 400×400 的单通道灰度图片, 经 4×4 卷积核采用扩展方式进行卷积。卷积核 (a) 为平均卷积核, 对图像进行了一定的模糊; 卷积核 (b) 使图片产生了过曝光的效果; 卷积核 (c) 则提取了图片的边缘, 产生了类似于浮雕的效果。这类卷积操作是各种图像处理软件中常用的各种滤镜操作的基础, 调节卷积核的大小, 其中元素的数值, 以及卷积核在图像上的滑动方式, 就可以得到各种不同的图像处理效果。

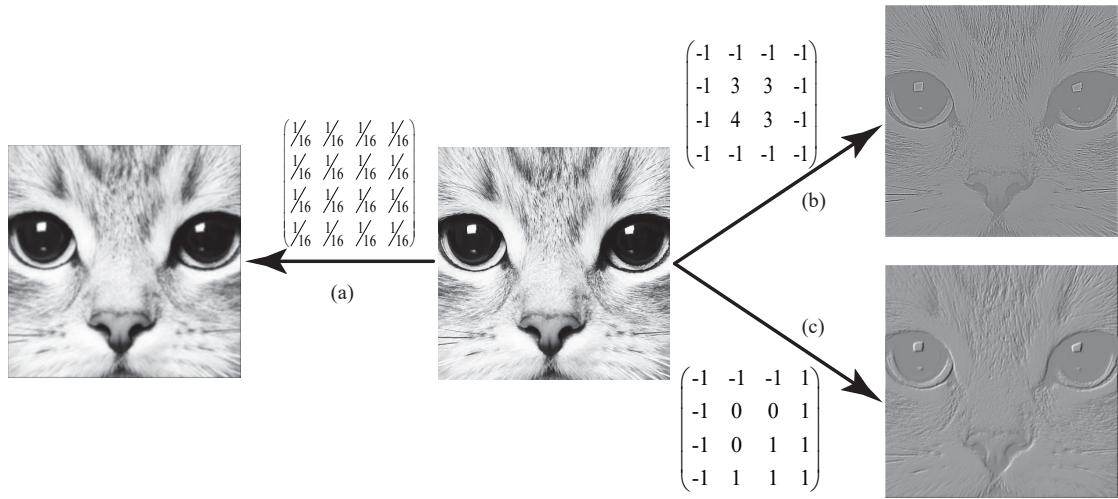


图 2.2 不同的卷积核的作用

上述几个例子中卷积核的各个元素之和都为 1, 这保证了卷积操作后原图像和特征图的像素值的均值不会发生变化。如果原图是经过一定的标准化处理过的 (如像素的均值为 0.5), 则特征图在统计上同样保持这个特征。这个约束保证

了图像的像素值不会过大而发生溢出或过小而消失。这在深度神经网络的训练过程中是必要的。

由于卷积操作中计算每个特征图的元素时，卷积核以一定方式“混合”了原图像中几个相邻像素的信息，卷积操作提取了原图片中的部分信息，而丢弃了其他信息，总信息的量减小了。这个过程可以看作对图片进行了降维。图像分类问题也可以看作对输入进行降维，保留主要成分（用于分类的主要特征），丢弃次要成分（其他无关的细节）的过程。用已知数据集训练卷积神经网络，就可以用反向传播算法使得卷积核“学会”提取用于分类的信息，而丢弃无关信息。例如区分圆形和方形色块时，色块边缘的形状是用于分类的信息，而色块本身的颜色则是无关的。再例如训练神经网络分类器时，如果训练集中每个类的图片既有在强光下拍摄的也有在弱光下拍摄的，神经网络将会注重于那些与亮度无关的特征。因此，卷积神经网络在很多场合下可以作为特征提取器使用，可以将较高维的图像转化为低维的特征图。这一点也是迁移学习的理论基础之一。

在运用反向传播算法时，假设神经网络为一个接收输入 P 的函数 F ，已知 P 的类别为 I ，网络的权值权值为参数 θ ，它输出 P 的分类预测值为

$$\hat{I} = F(P; \theta) \quad (2.4)$$

则可以定义预测值与真实值的误差，即损失函数

$$L = \|I - \hat{I}\|^2 \quad (2.5)$$

神经网络需要在训练数据集 $\{P_i\}$ 上最小化误差

$$\arg \min_{\theta} \sum_i (I_i - F(P_i; \theta))^2 \quad (2.6)$$

只需让 θ 沿损失函数的梯度方向 ∇L 下降即可。下降的距离称为训练步长。实际中由于全体训练集数据量非常大，现有的硬件条件下往往无法直接计算总体梯度，此时需要将训练集随机分成一系列小数据集，每次在小数据集上进行梯度下降。这种处理称为随机梯度下降法^[1]，如图 2.3 所示，每个小数据集称为迷你批次 (minibatch)。每次梯度下降的方向和总体梯度的方向会有一定的偏差，但是由于批次中的数据集是随机选取的，由他们计算出的梯度可以一定程度上代表总体梯度，多次下降的整体方向和总体梯度是一致的。本文训练网络就采用了随机梯度下降法。

需要指出，与很多传统的优化过程不同，神经网络的损失函数不一定具有良好的凸性质，优化过程并不能总是收敛到全局最优点，而是有可能收敛到某个局部最优点。神经网络的初始值 θ_{start} ，训练过程中学习率的选取，迷你批次的大小等诸多因素都会影响网络的收敛速度和最终的收敛点 θ_{opt} 。由于神经网络的计

算过程极为复杂，训练中其反向传播过程理论分析仍然在发展的初步阶段，大部分时候仅能对实验结果起到解释作用，对实际问题仅有有限的指导价值。实际中为了加快网络的收敛速率，以及防止网络优化到局部最优点，一些经验法则往往更为重要。本章后续介绍的训练过程即是结合文献中的理论分析，训练过程和实验结果调节而来。

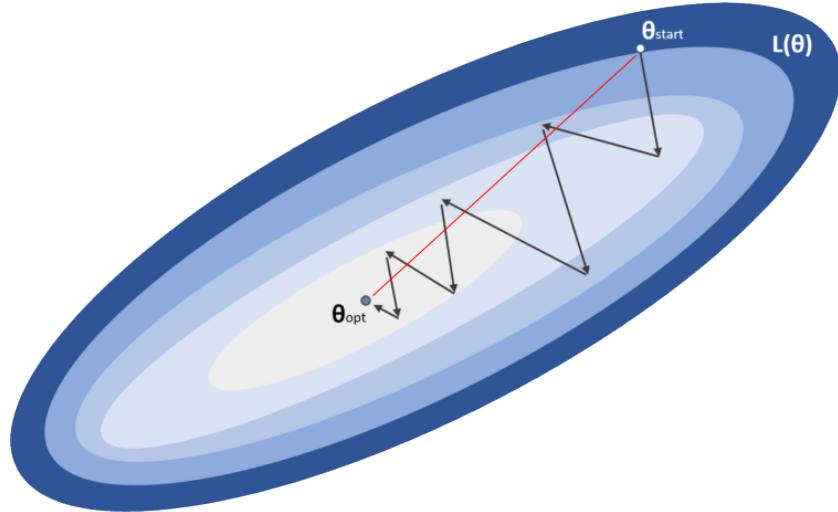


图 2.3 随机梯度下降示意图

2.2 微调与数据增强

迁移学习^[19]指将在某个领域上学习到的知识应用在另一个领域当中。此处的领域可以是不同的问题，也可以是同一个问题的不同数据集。实际生活中同样存在迁移学习的例子，例如已经学会骑自行车的人学习骑摩托车往往比不会骑自行车的人更快，因为他们把控制自行车的方法应用在控制摩托车上。迁移学习的思想被广泛引用在机器学习领域。在卷积神经网络的模型中，“知识”即是学习而来的网络权值 θ_{opt} 。2017 年 Kaiser 等人即提出了一套通用的神经网络模型^[20]，对输入做一定的预处理后，该模型可以同时解决图像分类问题和自然语言处理问题。这表明即使是被人类认为是完全不同领域的内在模式之间也存在一定的联系。

迁移学习思想的一个典型应用即是将大数据集上训练好的神经网络应用在小数据集上。如前文所述，当训练集的大小和网络规模相匹配时，训练好的网络可以有效地提取输入图片的特征。如果小数据集的数据和大数据集的数据类似（如都是真实物体的照片），该网络同样可以有效提取新数据集上图片的特征。此时数据的维度已经被有效减小了，可以在网络的输出后加上新的分类层进行分类。此时分类层的规模远小于整个卷积神经网络，即使在较小的数据集上也不会

发生过拟合。这种保留大部分参数不变，只在一小部分参数上应用反向传播算法的方法即被称为微调 (fine-tuning)。微调方法广泛应用于各种深度学习问题当中。



图 2.4 对图像的旋转扩充

数据增强是另一种减少过拟合的方法。其思路是对训练数据集做一些变换，但是保持关键特征不变，然后将原数据和变换后的数据都输入神经网络进行训练。例如数据集中有 100 张正立苹果的图片，对每一张图片做 90° 、 180° 、 270° 旋转，如图 2.4 所示，将数据量扩展到 400 张。如果不做这种扩展，神经网络有可能只能识别正立的苹果。这是因为训练集中只有正立的苹果，神经网络学习了一些随着图片旋转变化的特征。如果待识别的图片中的苹果是倒立的，这些特征将会发生变化，使得识别失败。此时就可以说神经网络发生了过拟合。但是对训练集做了旋转扩充后，神经网络不得不学习那些旋转不变的特征，或是学习所有 4 个角度下的特征，否则损失函数无法被优化。图像分类领域其他常用的数据增强方法还包括了随机裁剪，上下翻转，左右翻转，对比度调节，色调调节，加入随机噪声等等。需要指出的是，数据增强只能产生和真实数据十分类似的人造数据，这些数据对于训练起到的效果并不能和真实数据相比，一般只能作为一种辅助手段使用。训练较大规模的神经网络时还应该尽量多地收集有代表性的真实数据。

2.3 方法概述

2.3.1. 神经网络结构

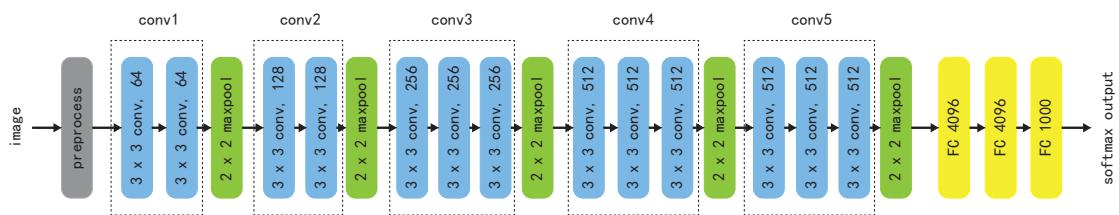


图 2.5 VGG-16 网络结构

本文采用了在 VGG-16 网络^[12]上修改得来的网络。VGG-16 的结构如图 2.5 所示。网络前部由 5 个依次相连的卷积模块构成，最前端的输入层接受 224×224 分辨率的三通道 RGB 图像作为输入。每个模块中都有 2 或 3 个 3×3 卷积核构成的卷积层，这些卷积层首尾相连，其输出使用 ReLU 函数进行激活。ReLU 函数的定义为

$$f(x) = \max(0, x) \quad (2.7)$$

各种神经网络模型中使用了很多不同的激活函数。三种常见激活函数的图像如图 2.6 所示。为了让输出处于同一范围内，图中 tanh 函数做了一定的变换，绘制的是 $\frac{1+\tanh x}{2}$ 的图像。可以看出，在输入较小时，三个函数的梯度类似；但是输入较大时，tanh 和 sigmoid 函数的梯度很小，这就使得反向传播计算时梯度不能有效传递，网络收敛变慢。AlexNet 实验表明^[10]，使用 ReLU 函数比其他常用激活函数有更好的收敛性能。此外，ReLU 函数比另外两种函数需要更小的计算开销。现在主流的图像识别模型大部分都使用 ReLU 作为激活函数。

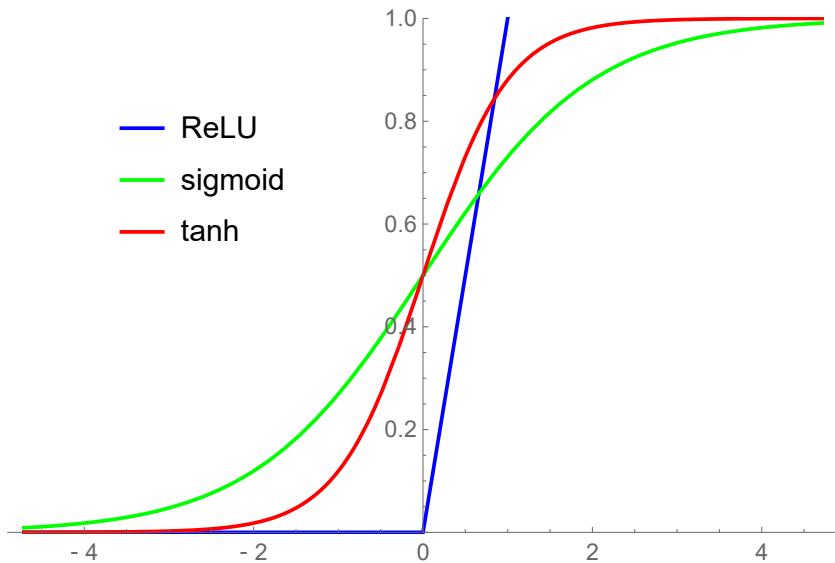


图 2.6 三种常见激活函数的图像

每个卷积模块后都有一个 2×2 最大值池化层 (maxpooling)，用于将输出的特征图分辨率减半。网络的末端是三个全连接层，前两个全连接层末端也采用 ReLU 函数激活。网络最终输出一个 1000 维向量，并使用 softmax 函数转化为概率分布。softmax 函数的定义为

$$p(i) = \frac{e^{x(i)}}{\sum_i e^{x(i)}} \quad (2.8)$$

其中 $x(i)$ 为输出的向量中的各个元素， $p(i)$ 为转化后的概率分布。VGG-16 网络在 ILSVRC-2012 数据集^[21] 上实现了 7.4% 的测试误差。

一般情况下，智能服务机器人工作的场景相对有限，其需要识别的物体类别要少得多（一般少于 100 类），相对而言 VGG-16 网络的全连接层规模过大，既占用了过多的存储和计算资源，也会导致过拟合。此时可以采用规模较小的网络模型，但是由于较小规模的模型的能力不足，实验中发现并不能有效提取图片特征，导致最终识别结果正确率很低。所以本文利用了大规模网络的特征提取能力，对原 VGG-16 网络的结构进行了精简，修改后的网络如图 2.7 所示。其中，所有卷积层原样保留。最后一个卷积层后的池化层增大到 6×6 ，使得输出的特征图的分辨率从 7×7 降低到 3×3 。后续的全连接层的尺寸也大大减小。VGG-16 全连接层的参数数目约为 120×10^6 个，修改后，如果最终输出向量为 50 维，参数减少到约 5×10^6 个。实验中，首先在 ILSVRC-2012 数据集上训练完整的 VGG-16 网络，该数据集有 1000 个类别，每个类别约有 10000 张图片。训练完成后，取各个卷积层的卷积核权值，并连接上修改后的全连接层构成新的网络。在修改后的新网络中，将前四个卷积模块的权值固定不动，只用新的较小的训练集训练最后三个卷积层以及三个全连接层的权值，即图中 conv5、FC 1024、FC 256、FC N 的权值。

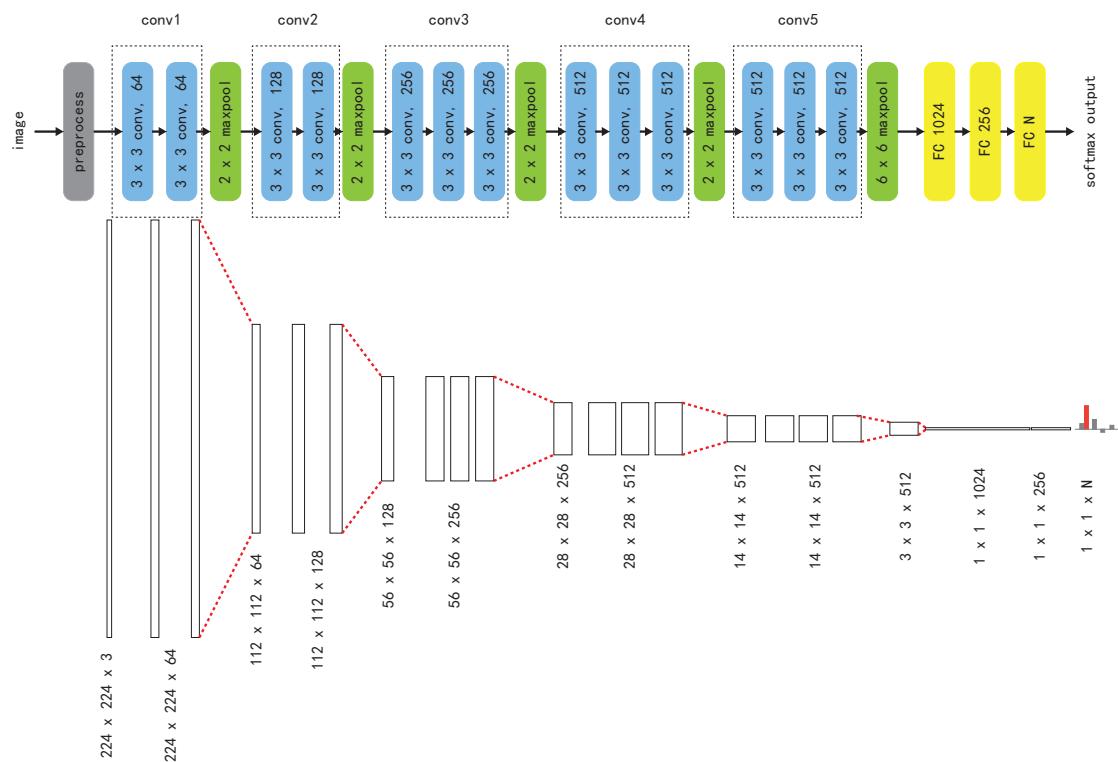


图 2.7 修改后的 VGG-16 网络结构与中间层输出

神经网络开始训练之前，需要初始化从头训练的层的权值，即 FC 1024、FC 256、FC N 的权值。本文的实验中，使用 Xavier Glorot 等人在 2010 年提出的 Xavier 方法^[22] 初始化网络权值。Xavier 方法是一种标准化 (normalization) 方法，其目的是为了如第 2.1 节所讨论的，避免网络的输出在训练过程中发生溢出或消

失。如果网络某一层的输入维度为 n , 输出维度为 m , Xavier 初始化方法即是使得网络中的权值满足均匀分布

$$W \sim U\left[-\sqrt{\frac{6}{n+m}}, \sqrt{\frac{6}{n+m}}\right] \quad (2.9)$$

理论推导表明, 在输入满足一定条件的情况下, 这种形式初始化的网络权值可以避免输出溢出或消失的问题。相关实验结果也表明 Xavier 初始化方法可以有效加快网络收敛速度。此方法的另一个优势是对于任意尺寸的网络, 都有一套统一的方法进行初始化, 而不需要根据具体情况手动调整。

为了可视化神经网络的中间隐藏层的输出, 本文在 ILSVRC-2012 数据集上训练完成 VGG-16 网络后, 选取了一些常见物体的图片作为输入, 将网络中间层的输出保存为单通道图片。其中一个物体输入网络后, 第 1 个和第 2 个卷积模块的输出结果如图 2.8 所示。实际中两个卷积模块的输出通道数分别是 64 和 128, 图中只随机选取了其中一小部分, 最左侧为输入图片的 RGB 通道图。可以看到, 不同的卷积核对于图片的不同部分进行了提取, 有些关注物体的边缘, 有些关注物体的颜色纹理, 也有些关注背景。经过多层卷积层的特征提取后, 物体的抽象特征就可以被有效提取出来, 在这些抽象特征上全连接层可以很容易地进行图像分类。对深度神经网络中间层输出结果以及卷积核权值的可视化是卷积神经网络领域的研究热点之一。2017 年 Olah 等人提出了一套实用的方法, 通过对卷积神经网络的输入进行优化, 可视化了网络中不同的神经元和卷积层关注的特征^[23]。这类研究有助于帮助人类理解神经网络的工作过程, 而不是将其看作黑箱。实验表明, 卷积神经网络前部的卷积层注重于基本特征, 如图像的边缘、基本的几何形状等; 后部的卷积层则注重于更为抽象的特征。有了对神经网络如何逐步提取图像特征的理解, 人类就可以设计出更有效的网络结构。

2.3.2. 训练数据集的采集与处理

和 ImageNet 竞赛的场景不同, 服务机器人需要识别的物体种类更少, 但是更加精细。例如, 在竞赛所用的数据库中, “orange” 分类下包括了各种大小、形状和颜色的橙子图片。但是在家庭环境中机器人需要将不同的橙子分开。这为训练数据的采集带来了很大的问题。首先, 物体的图片很难从网络上得到, 因为网络上搜索到的图片和实际中的物体往往有细微的区别, 这些物体的图片需要机器人自身进行采集; 其次, 每个物体只能获得十分有限数目的图片, 而且这些图片之间往往比较类似。为了获得泛化性能良好的网络模型, 数据扩充过程将十分关键。

数据采集的另一个问题是需要将物体从背景中分割出来。这可以由人力手工完成, 但是会花费大量的时间。本文利用了三维点云信息来完成物体的分割。

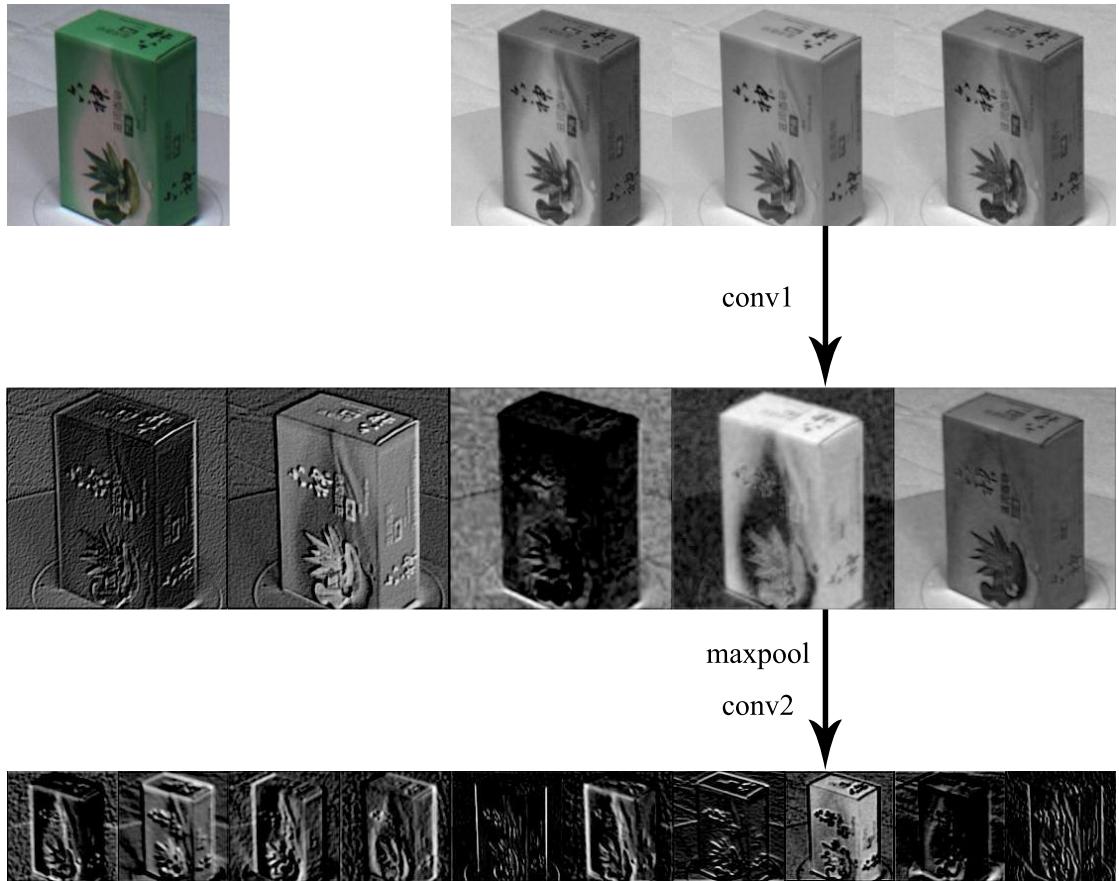


图 2.8 VGG-16 网络的前两个卷积模块输出的特征图



图 2.9 可佳机器人及其视觉传感器

可佳机器人及其视觉传感器系统如图 2.9 所示，视觉传感器由一个微软 Kinect 和一个高分辨率二维相机构成。两个相机可以采用一块印刷有已知图案的平板进行校准^[24]，使得 Kinect 视野中的任一像素 (s_k, t_k) 都可以和二维相机中的像素点 (s_c, t_c) 相对应。家庭环境中大部分情况下物体都放置在平面上，如桌子、书架、柜子等家具上，利用 RANSAC^[25] 算法可以很容易地从整个视野的三维点云中提取属于平面的点，而这些点上部的点进行欧几里得聚类就是各个独立物体的点。

云。从这些点云可以计算得到物体在 Kinect 视野中的像素空间上的包围盒，然后从上文所述的像素对应关系将包围盒四个角的像素对应到二维相机的图像中，就可以从二维相机的图像中分割出物体的高分辨率的图片。在对分辨率要求较低的情况下，也可以直接使用 Kinect 的彩色图像。

采集图像时，将物体放置在一个转盘的中心点，使得转盘以恒定速率缓慢转动，机器人在不同的角度以固定的时间间隔拍摄二维图像和三维点云，并分割物体得到训练图片。这样就可以得到物体在各个视角下的图片，这些图片按照 4:1 的比例被随机分为训练集和测试集，训练集将被用于训练神经网络，测试集被用于对训练得到的模型进行验证。训练图片在输入神经网络前使用算法 2.1 进行了预处理。其中随机裁剪尽可能保留了图片的信息，例如一幅 400×500 的图片上采用 400×400 的方框进行裁剪。这些预处理每次训练循环时都会被执行一次，每次循环的训练数据都各不相同，理论上总的数据量是无限的。测试图片时，不对图片进行亮度和对比度的随机调节。

Data: 分割得到的物体图片

Result: 224×224 的 RGB 图片

```

1 foreach 物体图片 do
2   对图片进行正方形随机裁剪
3   以 50% 的几率对图片进行左右翻转
4   以 50% 的几率对图片进行上下翻转
5   对图片进行随机亮度调整
6   对图片进行随机对比度调整
7   对图片的像素值进行标准化
8   对图片进行重采样转为  $224 \times 224$  分辨率
9   将图片像素值从整形转化为浮点型
10 end

```

算法 2.1: 训练图片的预处理过程

2.3.3. 神经网络的训练过程

由于本文使用了微调方法训练神经网络，且训练集相对于网络规模较小，训练过程中使用了学习率递减的方法来防止过拟合。根据理论和实验的结果^[26]，训练时采用的迷你批次中的数据量越多，训练的学习率应该越大，二者成近似的线性关系。这是由于迷你批次越大，计算得到的梯度方向就越准确，此时即使采用较大的学习率，也不会导致参数的随机梯度下降发生很大的偏移。相反，如果迷你批次很小，计算得到的梯度具有很大的随机性，此时采用较大的学习率可能会导致神经网络的参数掉入某个局部最优中，影响后续的优化过程。在本文的实验

中，如果迷你批次的大小为 M ，选取初始学习率的基线为

$$l = \sqrt{M} \times 10^{-5} \quad (2.10)$$

训练的前两个循环为了防止网络出现过拟合，学习率分别设置为 $0.1l$ 和 $0.3l$ ，这种方法被称为网络训练的“热身 (warmup)”。在第三个循环，设置学习率为 $0.6l$ 。随后的循环根据当前模型在测试数据集上的表现动态调整学习率，当测试精确度达到 75% 时，将学习率设置为 $0.5l$ ；当精确度达到 85% 时，设置为 $0.25l$ 。不同训练策略的训练结果将在第 2.5 节中详细讨论。

实验选取的损失函数是网络的输出结果与真实值的交叉熵。将真实值和神经网络的输出看成两个离散的概率分布 y 和 \hat{y} ，则交叉熵为

$$H(y, \hat{y}) = - \sum_x y(x) \log \hat{y}(x) \quad (2.11)$$

交叉熵衡量了两个概率分布的相似程度，如果 $y = \hat{y}$ ，交叉熵最小。在每个迷你批次上使用 Adam 算法优化该损失函数。

2.4 实验环境

2.4.1. 实验所用相关软件库简介

可佳机器人使用开源机器人框架 ROS^[27]。该框架支持各种常见的硬件架构和操作系统，提供了丰富的 Python 和 C++ 接口，并内置了大量的常用功能和算法，是机器人学界最常用的控制框架之一。ROS 提供了一套多个应用间使用网络进行异步通信的框架，各个应用称为 ROS 节点，机器人控制系统的中心是一个主节点，称为 MASTER，其他各个模块以从节点的形式运行，称为 NODE。主节点负责协调各个从节点之间的交流，从节点之间可以很容易地通过网络通信。只要各个节点运行在同一个网络中，无论它们是否在同一台计算机上都可以协同运行而不需要在程序逻辑上做任何修改。节点之间的通信以结构化的 ROS 消息的形式进行，支持几乎所有常用的基本数据类型和图像、点云、地图、规划、语音等复杂数据类型。消息通过发送和接收的流程在 NODE 之间传递，发送出来的消息具有一个特定的字符串标记的主题，称为 topic，消息本身被保存在 MASTER 维护的每个 topic 对应消息队列中。接收消息的 NODE 可以根据 topic 从此队列中收回消息。消息除了含有数据外，还含有时间戳和编号，便于程序处理。

以可佳机器人进行物体抓取为例，其中涉及到的各个节点如图 2.10 所示。图中，方框口表示各个 ROS 节点，蓝色箭头 → 表示物理信息的流动，绿色箭头 → 表示 ROS 消息的流动。其中各个控制硬件的节点中包含了硬件驱动，Camera

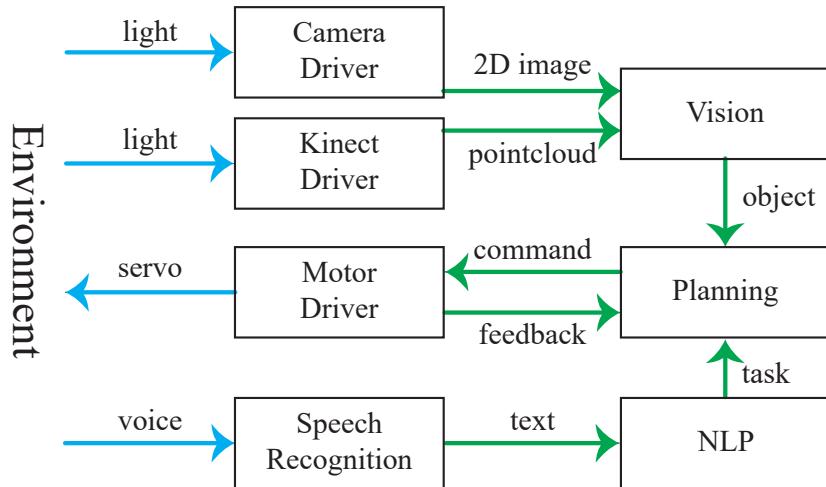


图 2.10 可佳机器人的 ROS 节点示意图

Driver 和 Kinect Driver 节点分别驱动二维相机和微软 Kinect，并将相机的输出转换为通用的图像和点云格式发送给视觉节点 Vision。视觉节点对视觉信息进行处理，完成物体的识别和定位，并将结果发送给决策节点 Planning。此时机器人的操作人以语音的形式向机器人提出一个要求，该要求被语音识别节点 Speech Recognition 转化为文本并发送给自然语言处理节点 NLP。自然语言处理节点分析该文本并转化为一系列任务，发送给决策节点。决策节点根据任务内容以及物体识别的结果规划出机器人需要完成的一系列动作，并发送给电机控制节点 Motor Driver。电机控制节点将动作转化为电机可以接收的电信号发送给电机，并且将电机的位置和力矩等反馈收集起来反馈给决策节点，决策节点根据这些反馈判断整个任务的进行情况。若任务已经完成，则开始接收执行下一个任务。

本文收集训练数据时，将拍摄图片作为可佳机器人的一个任务写入决策节点中。每次在转盘上更换待拍摄的物体后，操作人用语音告知机器人开始拍摄，机器人开始按照设置好的多个视角拍摄图片，并将图片保存在指定位置。图片的文件名使用物体的类别名称标注，便于后续处理。训练好网络模型后，将训练好的模型文件和识别模块整合到现有的视觉节点中，用于对物体进行实时识别。

本文采用 TensorFlow^[28] 作为神经网络框架。TensorFlow 是 Google 公司开发的基于静态计算图的计算框架，使用 Python 作为编程接口，可以在各种主流操作系统上运行，并且支持 CPU、GPU 等多种硬件。TensorFlow 的计算图基于张量（Tensor）。运算时首先定义一个具有输入输出的运算图，该计算图表示了张量的数据流，计算过程即是将张量输入该计算图然后取回结果的过程。一段简单的 TensorFlow 代码如下所示：

```

1 #!/usr/bin/python3
2 import tensorflow as tf
  
```

```

3
4 input_tensor = tf.placeholder(tf.float32, shape=[2, 1])
5 W = tf.constant([[-1.0, 0.0], [0.0, 1.0]])
6 output_tensor = tf.matmul(W, input_tensor)
7
8 with tf.Session() as sess:
9     print(sess.run(output_tensor, feed_dict={
10         input_tensor: [[2.5], [3.5]]}))

```

此段代码执行了矩阵乘法

$$\begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 2.5 \\ 3.5 \end{pmatrix} = \begin{pmatrix} -2.5 \\ 3.5 \end{pmatrix} \quad (2.12)$$

TensorFlow 提供了大量神经网络模型中常用的计算，如矩阵乘法、卷积、张量合并、激活函数等；也提供了各种实用的函数接口，如图像的解码编码、图像翻转等；并预先构建了很多常用的神经网络模型可以直接应用在新的项目当中。所以使用 TensorFlow 搭建卷积神经网络相当便捷，同时也有很高的灵活性。

2.4.2. 实验流程

可佳机器人使用一台运行 Ubuntu 14.04 x64 的笔记本电脑控制，该系统中安装了 ROS Indigo。训练图片采集完毕后，将图片放在一台安装了 Windows 10 x64 的工作站上进行训练。此工作站安装有一块 NVIDIA GTX 1060 3GB 显卡，运行 Python 3.6.4，TensorFlow 1.4.0。工作站安装了 CUDA 8.0 和 CUDNN 6.0 用于加速神经网络计算。由于训练所用数据很少，网络收敛很快，一般来说并不需要计算力很强的显卡。

由于 TensorFlow 和 ROS 都提供了 Python 接口，可以很容易地将卷积神经网络分类器整合到现有的视觉节点 Vision 当中。实验中首先将识别部分的代码加入视觉节点的现有代码中。训练完成后，将 TensorFlow 保存的网络模型文件放在可佳机器人的控制电脑中，然后运行视觉节点即可进行物体识别。对网络进行测试时，除了根据输出向量得到识别类别结果外，还根据输出的概率分布计算了香农信息熵

$$H = - \sum_i p(i) \log p(i) \quad (2.13)$$

其中，如果 $p(i) = 0$ ，则取 $p(i) \log p(i) = 0$ 。该信息熵反映了某次识别结果的可信度，越可信的结果其概率分布越集中在某个类别上，其信息熵越小。例如某个 5 分类问题的两个识别结果如表 2.1 中的 p_1 和 p_2 所示，可以看到 p_1 的分布更加“集中”，其计算得到的信息熵也较小。

	X_1	X_2	X_3	X_4	X_5	H
p_1	0.1	0.05	0	0.8	0.05	0.71
p_2	0.1	0.6	0.15	0.05	0.1	1.2

表 2.1 不同概率分布的信息熵

计算可信度是必须的，因为实际中机器人可能会遇到并不在训练集中出现过的物体。一般来说对这些未知物体的识别结果其可信度比较低。实际中，可以通过实验确定一个可信度阈值，低于此阈值的识别结果即被标记为未知物体。

2.5 实验结果与分析

2.5.1. 神经网络的训练过程

一次典型的数据采集拍摄了 16 个家庭环境中常见的物体，这些物体的尺寸、形状、表面纹理情况各不相同，如图 2.11 所示。为便于在文章中显示，拼接时对原图片进行了一定的缩放，实际中由于不同物体的形状和大小不同，训练图片的宽高比和分辨率也有很大的区别。使用第 2.3.2 小节中描述的方法对物体进行了拍摄和分割，每个物体拍摄了 150 张图片，其中 120 张用于训练，30 张用于测试。



图 2.11 实验所用物体

在前文叙述的硬件和软件上，训练中平均每张图片用时约 35 ms。实验中，分别选取了迷你批次大小为 4、8、16 对网络进行训练，训练中每个循环的学习率以前文所述的策略进行动态调整。其训练过程的学习率和在测试数据集上的识别精度如图 2.12 所示。其中横轴为训练的循环，每个循环中所有训练集图片都会被使用一次。绿色数据点 ● 是每个训练循环中使用的学习率，蓝色数据点 ● 是该次训练循环结束后，将当前网络模型应用在测试数据集上取得的识别精确度。

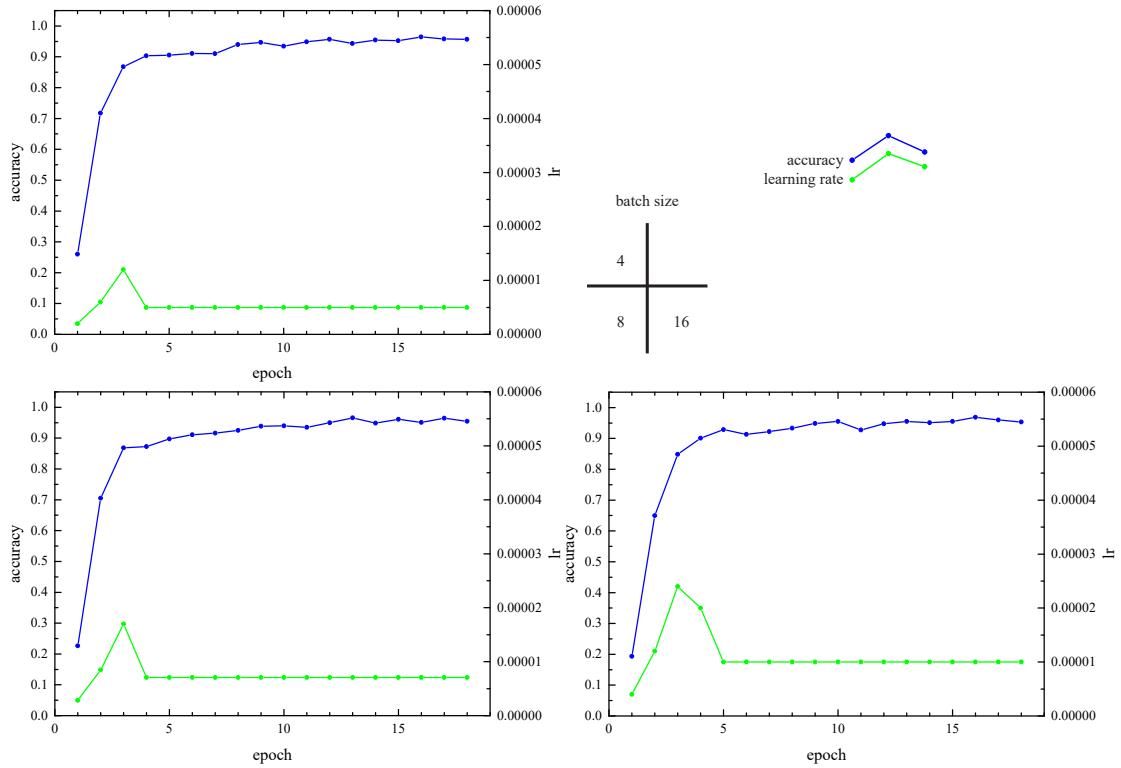


图 2.12 神经网络的训练过程

进行测试时，不对图片的亮度和对比度进行随机调整。

可以看到，相对于大数据集上的从头训练网络而言，本文采用的方法收敛速度非常快。在不同的迷你批次大小和学习率的设置下，神经网络在 5 个训练循环之后就可以达到 90% 以上的测试精度，并且在约 10 个循环后达到收敛，最终在测试数据集上的识别精度都可以达到 95% 以上。虽然此时可以实现很高的验证精度，但这并不意味着机器人使用训练好的模型在真实环境中也可以达到类似的精度，因为真实环境的光照条件和背景是无法预知的。下节将给出不同环境中的实际运行结果。

2.5.2. 在机器人的运行结果

在机器人上运行训练好的卷积神经网络分类器有两种方式。一种是直接在机器人的控制电脑上运行分类器。大部分情况下，控制电脑不具有高性能显卡，所以运行分类器较慢。另一种方式是在另一台安装有高性能显卡的计算机上运行分类器，该计算机和机器人控制电脑之间使用 ROS 消息通信。这种设置下，需要考虑消息传递带来的延迟，如果网络状况不佳，最终的识别速度可能会慢于在机器人控制电脑上运行分类器。本文尝试了两种方式，其中在控制机器人的电脑上运行分类器分类一张图片需要约 250 ms；在另一台安装有 NVIDIA GTX 1080 Ti 的工作站上，运行一次只需要约 15 ms，但是网络通信会带来 100 ms ~ 500 ms

不等的延迟，延迟时间和当前网络状况相关。综合来看，使用工作站运行分类器在时间上没有明显的优势。因为服务机器人执行任务的频率较低，且大部分时间花费在运动当中（例如可佳机器人抓取一个物体过程中机械臂的运动用时约为 15 s），该识别速率已经可以保证实时运行。接下来的实验都采用了机器人控制电脑直接运行分类器的方式。

训练数据集中的图片是在一个白色的转盘上拍摄的，圆盘放置在房间中央的桌子上，上方有白色荧光灯管产生的较为均匀的照明光照射。测试分类器在真实环境中的精确度时，选取了三个不同的环境，分别为房间中央的铺有白色桌布的桌面，房间中央没有桌布的黄色桌面，以及房间角落里的书架。第一种环境和训练数据采集的环境十分接近；第二种环境和收集数据的环境相比，光照条件类似，但是物体的背景颜色发生了变化；第三种环境下，照明度较低，机器人看到的物体亮度比收集数据的条件下暗得多。每种环境下，让机器人尝试识别所有已知物体，每个物体摆放时旋转几次使得机器人看到不同的方向。每个物体识别约 5 次，计算所有物体上总的识别成功率。不同环境下识别结果如表 2.2 所示。

环境	识别精确度
桌布	86%
桌面	80%
书架	68%

表 2.2 不同环境下机器人运行分类器的识别率

需要指出，由于真实环境中很多因素是无法控制的，例如一天中不同时间段的自然光照条件不同，以及物体周围的背景可能会散射特定颜色的光导致物体的颜色发生改变，而这些因素都会影响识别的结果。实验中可以观察到，对于同一个环境中的同一个物体，相邻两次识别的结果也可能不同。所以本文给出的结果只是多次实验中的一次有代表性的实验给出的结果。可以看到，在和原环境（即采集训练图片的环境）相似的环境下，识别成功率可以达到 85% 以上。光照条件不变，而背景改变时，识别率仍然可以达到 80%。但是当光照条件发生较大变化时，识别率降低到 70% 以下。实验结果表明光照对识别效果有较大的影响。另外，本文在训练时也尝试了不对训练图片进行随机亮度调整的处理方法。在这种情况下，识别器在测试数据集上仍然可以达到高于 90% 的识别率，但是在实际环境中，当光照发生较大变化时，成功率降低到了 30% 以下。这儿说明本文采用的数据扩充方法是有效的，通过随机调节亮度，将在较强光照下收集的数据一定程度上扩充到了弱光照条件下。

实验还比较了基于卷积神经网络的物体分类器和可佳机器人上原本的基于

特征的分类器的表现。原本的方法中，同样拍摄物体在转盘上的图片，然后结合了点云信息对物体完成分割。之后，对于每张训练图片，计算其整体的颜色直方图，并提取 SURF 特征点。这些特征以及该物体的类别被存储在模板文件中。对于任意待识别物体，同样计算其特征，然后和模板文件进行匹配，匹配度最高的特征对应的类别即为识别的类别。该方法存在一些问题。首先，由于模板文件只是简单的存储所有训练图片的特征，文件会随着训练数据集的增大线性增大，识别时间也会相应增加。当物体很多时，将花费很长时间。其次，该方法在提取训练集图片的特征时需要人工选取那些效果好的图片，去掉效果不好的那些，这个过程十分费时费力。本文采用的方法训练得到的网络模型文件不会随训练集的增大而增大，识别时间也保持不变。且整个过程的自动化程度较高，花费的时间较少。在机器人世界杯比赛中，之前的方法在相似环境中的识别率约为 80%，光照变化较大的环境中的识别率约为 50%，每次识别约耗时 100 ms。本文的方法在识别率上也超过了之前的基于特征的方法。

2.6 本章小结

本章详细介绍了基于卷积神经网络的图像分类器的网络结构、训练数据的采集方法、分类器的训练方法，并在机器人上进行了实际应用场景下的测试。为了解决小数据集带来的过拟合问题，本文使用了迁移学习、微调、学习率控制、数据增强等多种方法。机器人在实际环境中采集了数据进行了训练，并实现了较高的识别率。实验表明，该分类器对光照条件和背景的变化有一定的容忍度，且精度超过了之前使用的基于整体和局部特征的方法。该方法还结合了三维视觉信息实现了对物体的自动分割。本方法可以实时运行在可佳机器人上，并帮助其完成一些涉及物体识别的任务。

第3章 多摄像头物体定位与识别系统

3.1 三维视觉传感器简介

三维视觉传感器是在传统二维相机的基础上发展而来的。主流的三维相机工作时会投射具有一定特征的红外线到物体表面，然后通过反射回来的光成的像重构出物体表面的三维结构。典型的三维相机有微软公司于2010年上市的Kinect和2013年上市的Kinect 2.0，其主要结构分别如图3.1(a)和(b)所示。除了普通的RGB相机外，它们还有红外线投射和接收装置。

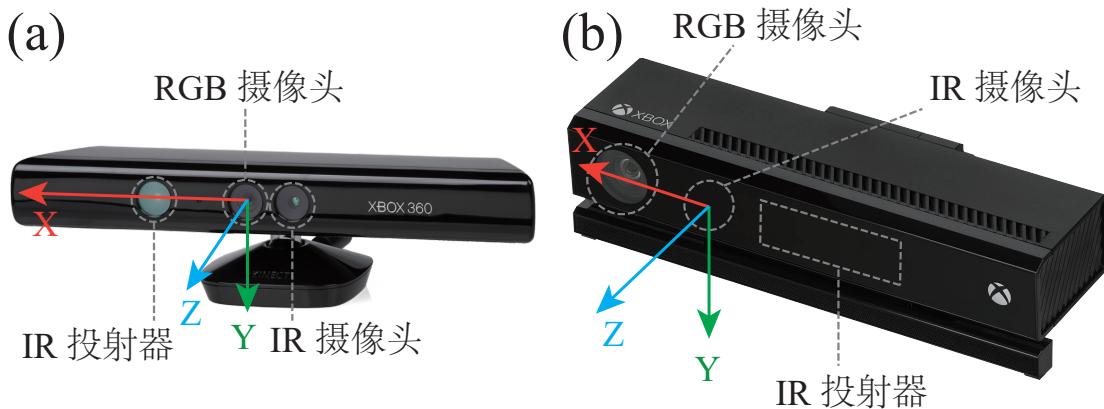


图3.1 微软 Kinect 传感器示意图

第一代Kinect传感器使用结构光重构物体表面的三维结构^[3]。红外线透射器发出构成了特定图案的红外光，这些红外光照射在物体表面后，反射回来在红外线接收器中成像。如果物体表面是正对相机的平面，那么红外光的图案不会发生变化；相对的，如果物体表面有一定的起伏，红外图案就会发生变形。从成像的变形图案中就可以计算出物体表面的三维结构。以图3.2中的情况为例。投射器发出均匀的黑白条纹光线，但是由于物体表面有起伏，接收器成像中的条纹是弯曲的。根据条纹的弯曲程度和间距等信息就可以计算物体表面的曲度。使用印刷有已知图案的平板可以在红外接收器的像素和RGB相机的像素间建立对应关系^[24]，这样就可以得到和彩色图案对应的三维信息，从而构建出RGBXYZ彩色三维点云。

第二代传感器Kinect 2.0采用了不同的飞行时间技术^[29]重构物体表面三维结构。红外线投射器会向各个方向发射出红外光线，这些光线被物体表面反射回接收器当中。测量光线从发射到接收经过的时间就可以计算出物体表面和传感器之间的距离。用上述同样的方法校准红外线接收器和RGB相机间的对应关系后，就可以为RGB图案中的每个像素加上深度信息，从深度信息同样可以重构出RGBXYZ点云。

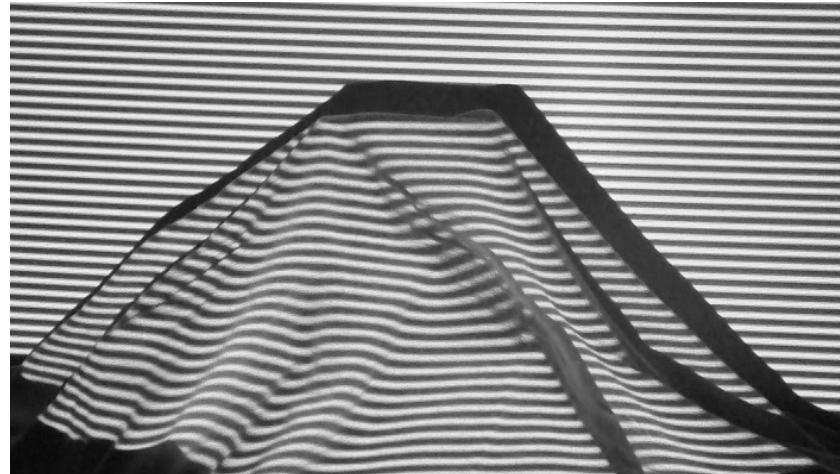


图 3.2 结构光的工作原理示例

三维点云是立体视觉系统处理算法的起点。从点云出发，可以计算各种整体或局部的立体特征；可以通过点云计算物体的中心点位置；可以将点云转化为多边形网格从而对物体进行三维建模。相对于二维图像，点云提供了丰富得多的且尺度不变的信息。而 Kinect 等商品化的三维视觉传感器能以较低的成本提供精度较高的三维点云，且可以在一定的环境下稳定工作，所以 Kinect 上市后不久就成为机器人领域最常见的视觉传感器之一，广泛应用于各种场景中。机器人世界杯 @Home 分组的比赛中，大部分参赛队伍的机器人都安装有 Kinect 或 Kinect 2.0。

三维视觉传感器的工作原理决定了它们具有一定的局限性。由于三维视觉传感器重建物体三维结构的基本假设是红外线接收器接收到的像是由红外线投射器发出的红外线在物体表面反射而来的，如果这个过程中有其他光源对红外线发生了干扰，就会导致三维结构重建的失败。因此，在室外强光环境下大部分三维传感器无法工作，因为阳光中的红外线会产生干扰，此时在红外线接收器中只能接收到过曝光的红外光成像。如果将两台 Kinect 指向同一个物体，它们发出的结构光会互相干扰，导致重构失败。此外，如果物体表面十分光滑，导致红外线发生了镜面反射，无法被接收器接收到，重构同样会失败。传统的二维相机只是被动地接收光信息，因此不会出现这些问题。此外，三维视觉传感器处理的数据量远多于二维相机，这对供电和数据传输提出了更高的要求。总体而言，精度越高的三维传感器成本也越高，调试过程越复杂，运行时也需要更多的处理时间。实际中，需要根据需求选取最合适的设备。本文采用了 Kinect 和 Kinect 2.0 作为三维视觉传感器。实验表明，它们的精度可以满足服务机器人的要求。为方便描述，在不产生误解的情况下，下文中将使用“摄像头”作为三维视觉传感器的简称。

3.2 多摄像头立体视觉系统

某一个时刻，一个三维传感器只能重构出当前摄像头视野的三维结构。如果希望得到某个物体的全貌，需要在不同的视角获取三维点云，然后将这些点云按照合理的方式拼接起来。一种方法是从点云计算局部特征点，如果两个点云有一定的重叠，就可以从重叠部分的对应特征点计算两个点云间的变换关系，从而完成拼接。这种方法在三维场景重建中有着广泛的应用^[30]，可以用于机器人导航等场景中。但是计算点云的局部特征十分复杂，而且在服务机器人的应用场景中会由于尺度较小产生很大的相对误差。

由三维视觉传感器的工作原理可知，其产生的点云中每个点的坐标都是相对于传感器本身而言的，该固定在摄像头硬件上坐标系称为摄像头坐标系，如图3.1中的XYZ标架所示。摄像头坐标系和外部坐标系之间的变换关系随着摄像头在空间中的位置和取向的变化而变化。拼接点云的关键即是建立不同摄像头坐标系之间的变换关系。如果两个摄像头之间的相对位置保持不变，该变换关系也保持不变。由于直接建立变换关系较为困难，可以先分别建立两个摄像头到某个外部坐标系之间的变换关系，然后将它们产生的点云分别变换到该外部坐标系当中，就可以完成点云的拼接。这种方法不需要对点云计算任何特征，且可以推广到更多个摄像头的情况。

现有 N 个不同的摄像头，各自的摄像头坐标系为 Φ_i ，另外定义一个外部坐标系为 Φ 作为基准。直接测量变换关系的各个角度和距离会产生很大的误差，所以本文采用了根据特征点进行参数优化的方法。 Φ_i 到 Φ 的变换关系可以用一个 4×4 的仿射变换矩阵表示

$$T_i = \begin{pmatrix} R_i & \vec{t}_i \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3.1)$$

其中 R_i 为一个 3×3 矩阵，表示旋转； \vec{t}_i 为一个三维向量，表示平移。该矩阵有 6 个自由度，在 Φ 中选取坐标已知的 M 个点，坐标标记为 P_k ，然后测量这些点在 Φ_i 中的坐标分别为 Q_k ，根据变换关系可以计算它们在 Φ 中的坐标为 $T_i Q_k$ ，通过最小化这些坐标和真实坐标的误差就可以计算出变换矩阵

$$\arg \min_{T_i} \sum_{k=1}^M E(P_k, T_i Q_k) \quad (3.2)$$

其中，函数 E 计算两个点之间的距离误差，实际中常取三维欧几里得距离的平方

$$E(P, Q) = (P_x - Q_x)^2 + (P_y - Q_y)^2 + (P_z - Q_z)^2 \quad (3.3)$$

从优化理论可知，为了减小误差，点的数目需要足够多，位置的选取应该具有代表性，且相隔一定的距离。对每个摄像头建立好变换关系矩阵后，在同一时刻收

集它们各自产生的点云 PC_i , 就可以拼接成点云 PC

$$PC = \sum_{i=1}^N T_i * PC_i \quad (3.4)$$

其中, 对点云的变换 $*$ 表示对点云中每个点的 XYZ 坐标应用变换矩阵 T_i , \sum 操作表示直接将各个点云中的点合并为一个新的点云。最终结果 PC 即是总体点云, 其坐标基于外部坐标系 Φ 。

上述校准过程的关键是需要有准确的坐标值 P_k 与 Q_k 。本文使用了 ALVAR 库中提供的二维码标签标记这些点^[31]。这些正方形的二维码标签具有特定的黑白图案, 很容易用简单的算法识别。实验中, 在一块平板上定义坐标系 Φ , 然后选取一些位置将不同的标签贴在板上, 制作而成的标定板如图 3.3 所示。每个摄像头从各自的点云中识别出各个标签的位置即为 Q_k , 从而得到变换矩阵。实验中需要固定各个摄像头和此标定板之间的相对位置, 然后进行校准。一般来说, 使得各个摄像头中标定板位于视野中心, 且距离摄像头距离适中时, 校准可以取得较高的精度。校准完成后, 将待识别物体放在视野当中, 就可以得到它的较为完整的点云 PC , 该点云被用于后续的定位和特征提取中。

3.3 物体的定位与三维特征的提取

对于典型的服务机器人面临的物体操作任务来说, 最有用的信息分别是: 物体的位置, 该信息决定了机器人运动的目标点, 此运动包含了机器人位置的移动以及机械臂的运动; 物体的尺寸, 该信息决定了机器人的硬件是否可以进行有效的操作, 太大或太小的物体都会导致操作失败; 物体的取向, 该信息决定了机器人机械臂末端的操作方式。另外, 有些机器人的机械臂末端具有吸盘等特殊结构, 此时需要在物体中寻找一个足够大的平面用于操作。总而言之, 有用的信息取决于机器人本身的工作方式。另外一些重要信息, 例如物体的硬度、重量、表面光滑程度、质量分布等, 由于很难通过视觉方法测量, 本文提出的方法暂不作涉及。

PC 中除了含有待定位和识别的物体的点之外, 还含有其他不需要的点, 例如摆放物体平面的点, 在点云上运行视觉算法前, 首先需要将这些不需要的点裁剪掉。由于物体一般只会摆放在一定范围之内, 将此范围之外的点去掉即可。另外, 如果定义物体摆放的平面的 Z 坐标为 0, 将所有 $z \leq 0$ 的点去掉即可将属于平面的点排除。此时就可以得到只由物体构成的点云 PC^{crop} 。三维摄像头的工作原理决定了物体正面的点的密度会高于物体侧面的点。由于点云实际上是对连续分布的物体表面的离散采样, 这种不均匀的分布在统计学上会导致各种计算偏差。为了消除这种不均匀性, 对 PC^{crop} 进行三维重采样, 使得每两个相邻

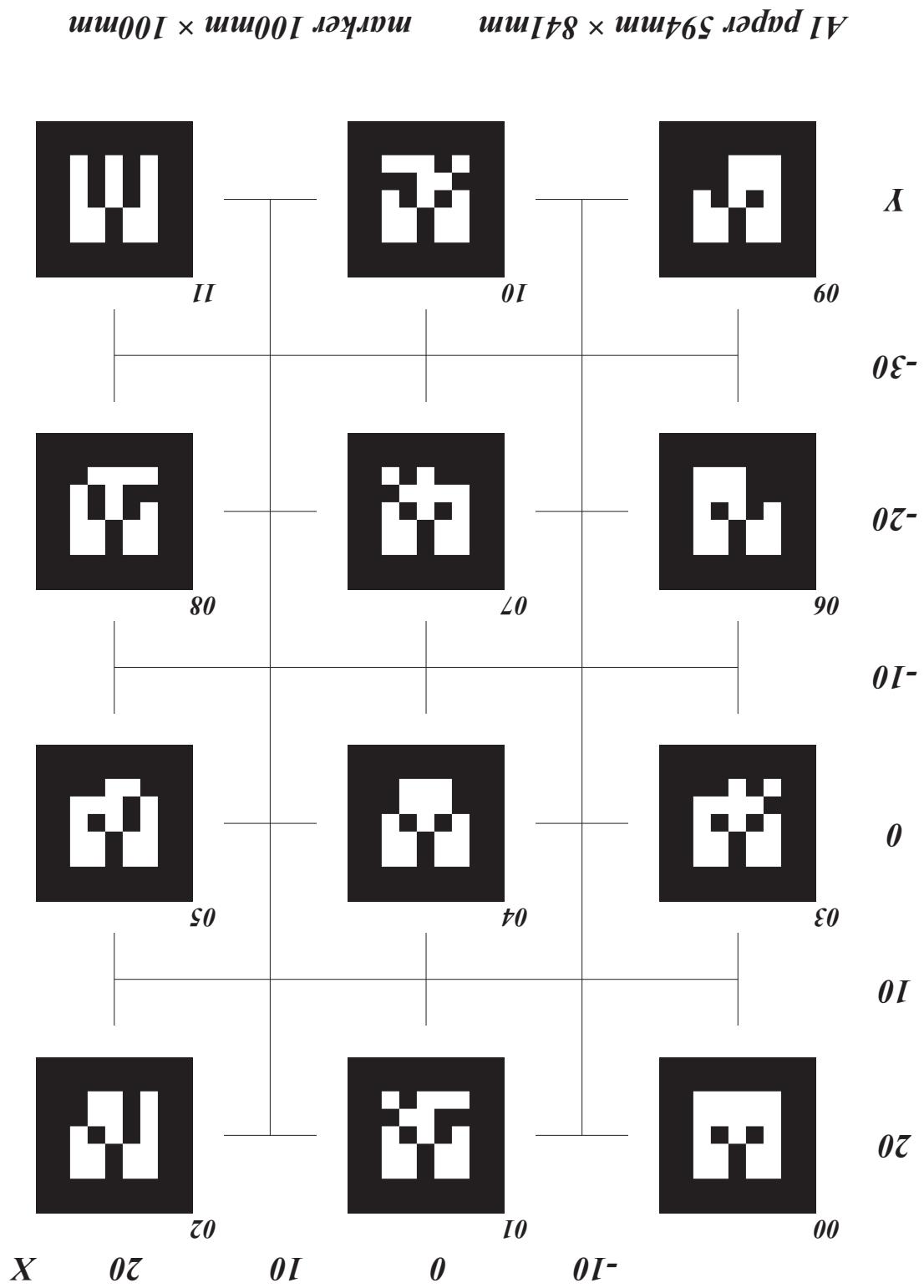


图 3.3 本文使用的标签板

点间的距离都接近于一个设定值。点云中除了含有多个物体外，还含有一些噪点。这些噪点悬浮在空中，是三维摄像头重构三维结构时产生的，不属于任何实际存在的表面。由于每个物体的表面的点之间距离较近，可以计算相邻点之间的

距离，然后根据距离对点云进行聚类，即可得到数个相互分离的只含有单个物体的点云 PC_i^{object} 。该点云即可以用于定位和特征提取。

物体的三维位置可以通过计算点云中 XYZ 坐标的均值得到

$$Centroid = \frac{1}{N} \sum_{i=1}^N (x_i, y_i, z_i) \quad (3.5)$$

如果点云是物体表面的均匀采样，该中心即为物体的几何中心点。物体的取向通过主成分提取进行计算。同样将点云看成物体表面的一系列采样，则可以计算其协方差矩阵

$$\Sigma = \begin{pmatrix} cov(X, X) & cov(X, Y) & cov(X, Z) \\ cov(Y, X) & cov(Y, Y) & cov(Y, Z) \\ cov(Z, X) & cov(Z, Y) & cov(Z, Z) \end{pmatrix} \quad (3.6)$$

其中，协方差定义为

$$cov(X, Y) = \frac{1}{N} \sum_{i=1}^N ((x_i - \bar{x})(y_i - \bar{y})) \quad (3.7)$$

该协方差矩阵的特征向量即代表了采样的三个主成分方向，也即物体的三个取向。为便于说明，以二维平面下的情况为例，如图 3.4 所示。坐标系 xOy 中的真实物体为红色线条 — 标出的椭圆形。在此物体上进行随机采样得到一系列点 (x_i, y_i) ，以蓝色圆圈 \circ 表示。计算这些点的协方差矩阵，并求出它的两个特征向量分别为 \vec{v} 、 \vec{u} 。以这些点坐标的均值为原点，以特征向量的方向为坐标轴，就可以建立一个新的坐标系 uCv 。此坐标系中两个轴分别沿着椭圆的长轴和短轴方向，代表了椭圆的取向。

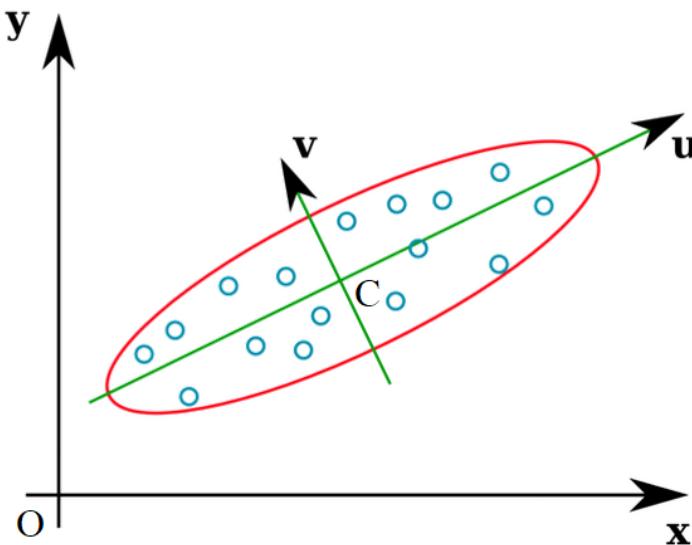


图 3.4 主成分提取示意图

得到物体的取向后，就可以计算物体的真实尺寸。首先将点云中点的坐标转换为特征向量定义的坐标系中的坐标，然后计算点云在坐标系的各个轴上的投影长度，这些值就代表了物体在各个取向方向的尺寸。此尺寸和物体在外部坐标系坐标轴上的投影长度不同，不会随着物体的旋转发生变化，因此更能代表物体的真实尺寸。根据尺寸和取向信息就可以计算物体在空间中占据的范围，即有取向的最小包围盒 OBB 。

物体表面的平面可以使用 RANSAC^[25] 方法计算。首先建立平面模型

$$ax + by + cz + d = 0 \quad (3.8)$$

然后调整模型参数，使得点云中有最多的点在该模型描述的平面的一定的误差范围内。如果这些点的数目足够多，就可以认为它们构成了一个平面。将它们从点云中去除，然后重复上述算法，就可以依次将点云中含有的平面提取出来。从平面上点的坐标可以计算平面的中心点和法向量。由于前述算法已经对点云进行了重采样，每两个相邻点间的距离是一个固定值，还可以通过点的数目计算平面的面积。上述算法的整个流程如算法 3.1 所示。

Data: 合并得到的点云 PC

Result: 每个物体的位置与特征

- 1 将 PC 中不在平面范围内的点裁剪，并去掉平面上的点，得到 PC^{crop}
- 2 对 PC^{crop} 进行重采样，得到 PC^{sample}
- 3 对 PC^{sample} 中的点进行聚类，得到多个子点云 $\{PC_i^{object}\}$
- 4 **foreach** $\{PC_i^{object}\}$ 中的 PC^{object} **do**
- 5 计算所有点的 XYZ 坐标的均值 $Centroid$
- 6 计算协方差矩阵，并计算特征向量 $\{\vec{v}_1, \vec{v}_2, \vec{v}_3\}$ 与特征值 $\{\lambda_1, \lambda_2, \lambda_3\}$
- 7 根据特征向量构成的坐标系，将 PC^{object} 转换到该坐标系中得到 PC^{PCA}
- 8 计算 PC^{PCA} 在坐标系各个轴上的投影长度为 $\{l_1, l_2, l_3\}$ ，并以 $Centroid$ 为中
心计算三维包围盒 OBB
- 9 提取 PC^{object} 中的各个平面 $\{Plane_k\}$
- 10 **foreach** $\{Plane_k\}$ 中的 $Plane$ **do**
- 11 计算平面的中心点 $Center$
- 12 计算平面面积 A
- 13 计算平面法向量 \vec{n}
- 14 **end**
- 15 **end**

算法 3.1: 物体的定位与特征计算过程

3.4 实验平台的搭建

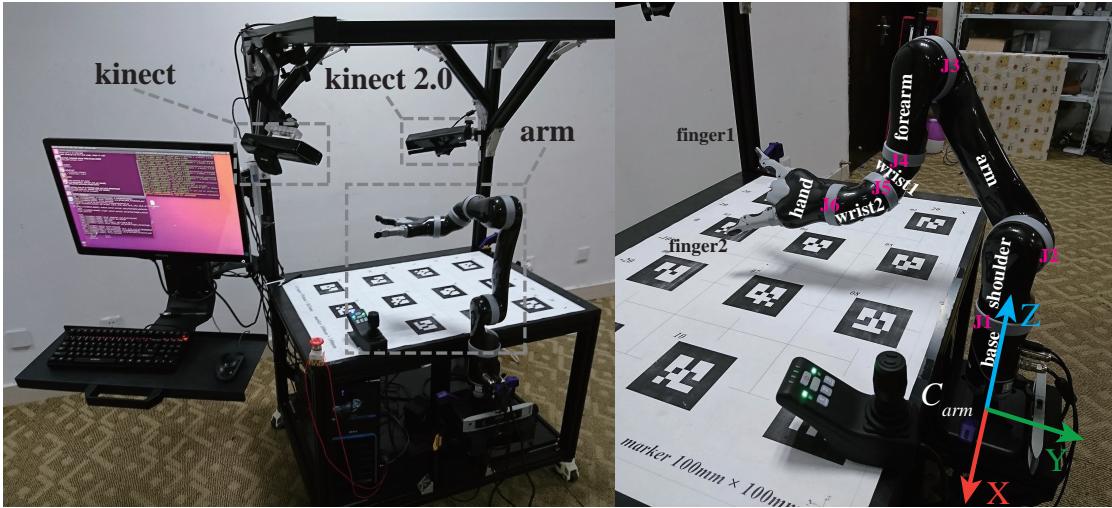


图 3.5 实验平台示意图

为了验证上述方法的有效性，本文搭建了一个模块化的实验平台进行实验。该平台如图 3.5 所示，主要由框架、视觉系统、手臂系统以及计算机构成。平台框架由铝合金部件拼接而成，中间和底部安装有硬塑料平面，分别用于放置物体和设备。多摄像头立体视觉系统由一台 Kinect 和一台 Kinect 2.0 构成，分别通过三自由度云台固定在框架上部，且指向平面的同一块区域。摄像头独立供电，数据线通过 USB 接口连接到计算机上。计算机运行 Ubuntu 16.04 x64 系统，并使用 ROS Kinetic 控制整个平台。

为验证前文所述定位和特征提取算法的实用性，在实验平台的侧面安装了一台 6 自由度机械臂，用于根据结果对物体进行操作。该机械臂为 Kinova Robotics 公司生产的 MICO²，由 6 个转动关节连接的 7 个部件构成，最末端的部件上安装有两根可以独立控制开合的手指。机械臂的每个关节都有独立的角度控制，并可以反馈任意时刻关节上的力矩大小。根据机械臂的尺寸和形状建立物理模型，可以通过每个关节的角度计算出机械臂的状态，从状态中可以得到机械臂末端的位置和取向。反过来，给出一个需求的末端位置，就可以规划出机械臂末端达到该位置时各个关节的角度，从而完成对机械臂的控制。实验时，采用机械臂驱动程序中提供的模型文件和 MoveIt!^[32] 完成对手臂的动作规划。手臂独立供电，并以 USB 接口与计算机通信。

和三维摄像头类似，对机械臂进行动作规划时，采用的都是某个位置相对于机械臂的坐标，该坐标系称为机械臂坐标系 Φ_{arm} ，其原点位于机械臂的基座中心，如图 3.5 的 XYZ 标架所示。由于视觉系统给出的所有结果都基于外部坐标系 Φ ，同样需要建立 Φ 和 Φ_{arm} 之间的变换关系。和摄像头坐标系不同，可以通过调整安装位置使得 Φ 和 Φ_{arm} 的坐标轴互相平行，此时只需要对坐标进行平移就

可以完成变换。平移距离可以直接测量得到，即有变换矩阵

$$T_{arm} = \begin{pmatrix} I & \vec{t} \\ 0 & 0 & 1 \end{pmatrix} \quad (3.9)$$

其中 I 为 3×3 的单位矩阵， \vec{t} 为平移向量。

为了对机械臂的运行精度进行评估，实验中还使用了一套外部的动作捕捉系统测量机械臂末端的位置。该系统由 OptiTrack 公司开发，由多个红外线摄像机构成，每个红外相机可以发出红外线，被表面覆盖有全反射材料的球形标记物反射后成像。将多个相机安装固定好后以一定方法进行校准后，就可以从多张图片中计算出标记物的三维坐标。如果将三个标记物固定在一起，则可以定义一个 6 自由度的刚体，系统可以测量出该刚体的位置和取向信息。摄像机和标记物如图 3.6 所示。实际中，相机大小约为 15 cm，标记物圆球的直径约为 0.5 cm，定位精度小于 1 mm，定位速率约为 120 帧/秒。



图 3.6 动作捕捉系统使用的相机和标记物

3.5 实验结果与分析

本文在安装并固定好视觉系统后，对系统进行了校准，如图 3.7 所示。其中分别标出了外部坐标系 Φ 和其中一个摄像头的坐标系 Φ_1 。绿色标出的二维码标签是在摄像头视野中完整出现并被识别的标签， Φ 的原点位置的标签用红色标记。随后选取了一些家庭环境中常见的有代表性的物体进行了定位和特征提取实验，并使用机械臂根据定位与特征提取的结果对物体进行了操作。实验用的计算机安装有一块 Intel Core i7 7700K 处理器，实验中处理一帧图像的时间约为 1 s。

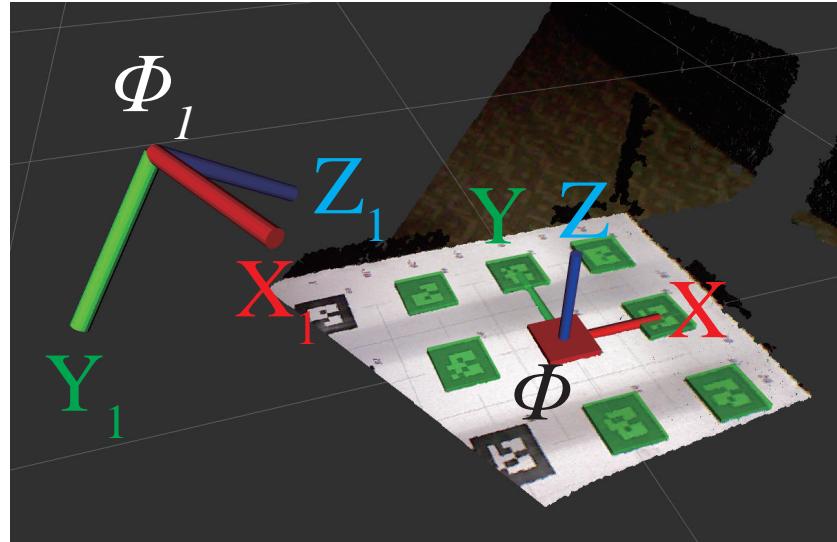


图 3.7 视觉系统校准过程示意图

3.5.1. 物体定位实验

实验选取了两个形状较为规则的物体进行了物体定位实验。物体 1 为一锥台形的无盖水杯，物体 2 为一四棱柱形的饮料，如图 3.8 所示。选取规则形状的物体的理由是这类物体可以精确测量其真实位置。由于外部坐标系 Φ 就定义在物体下方的标定板上，将物体放置在平面上后，测量物体中心点在标定板上的位置即可得到物体的真实位置。如果物体是不对称的，其真实中心点的位置将会很难确定。由于机器人进行物体操作时，物体在水平面中的位置比物体的高度更为重要，且由于物体的底部不能被视觉系统观测到，其定位高度并不能代表物体的真实情况，本文接下来将只测量和讨论物体定位结果在水平平面中的精度和分布情况。



图 3.8 物体定位实验所用的物体

对于每个实验物体，选取了 6 个有代表性的位置，每个位置进行了连续 $N = 200$ 次位置测量。物体的真实摆放位置记为 (x_0, y_0) ，测量值分别记为 (x_i, y_i) ，

$i = 1 \sim N$ 。计算每个测量值与均值的二维欧几里得距离值的方均根

$$\sigma = \sqrt{\frac{\sum_{i=1}^N ((x_i - \bar{x})^2 + (y_i - \bar{y})^2)}{N}} \quad (3.10)$$

σ 即为二维平面内定位结果的标准差。真实位置、测量结果的均值与标准差如图 3.9 所示。其中，以叉号 \times 标记真实位置，以蓝色圆点 \bullet 标记测量结果的均值 (\bar{x}, \bar{y}) ，括号中的数值是均值和真实值的偏差。

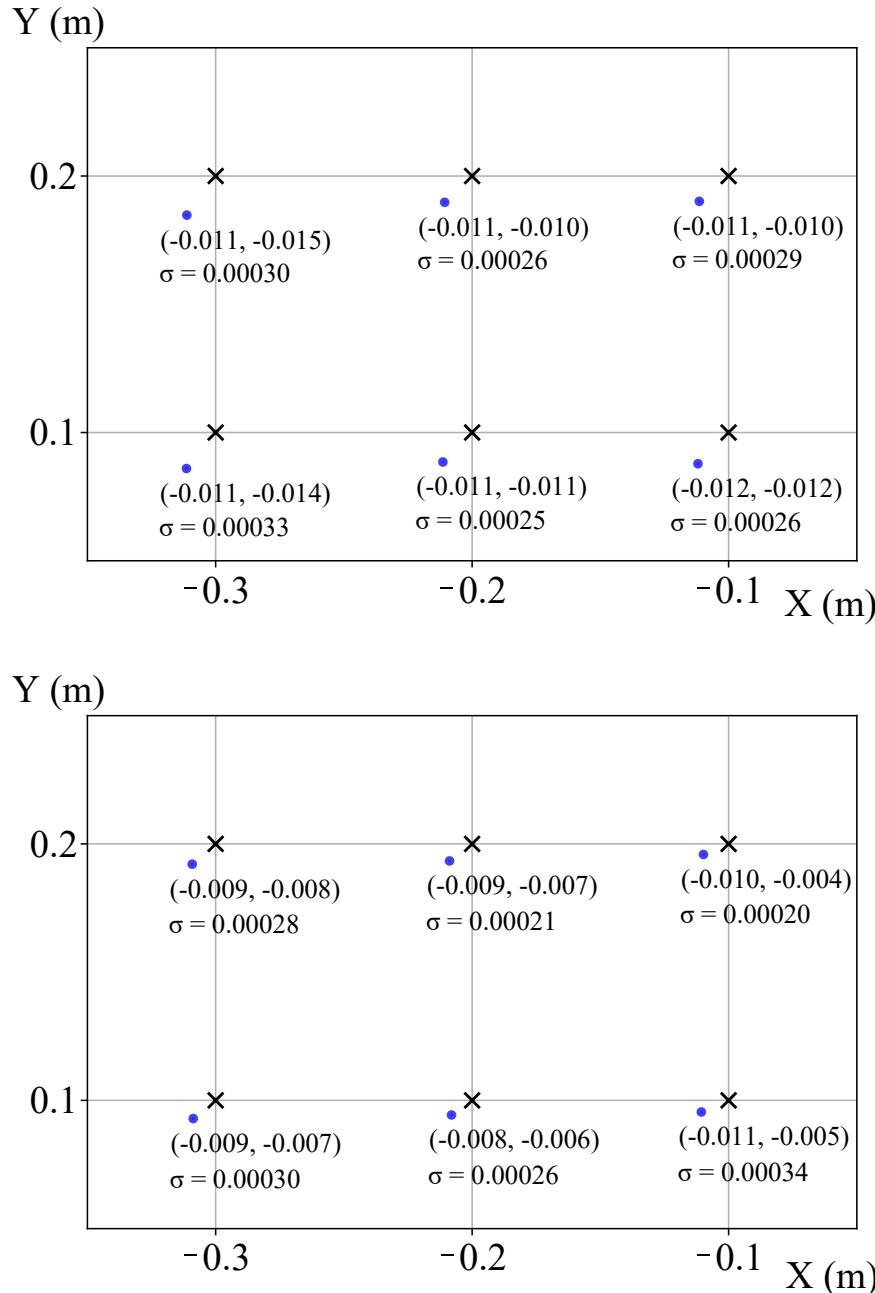


图 3.9 物体定位实验结果

可以看出，对于不同的实验物体的不同位置，测量结果均值和真实位置的偏差基本是一致的，且标准差基本也一致。根据对 Kinect 传感器定位误差的分

析^[33]，物体约偏离 Kinect 的视野中心，则点云中点的位置误差越大。实验结果表明，由于采用了两个相对放置的 Kinect 传感器，这种误差一定程度上发生了抵消，使得在平面的一定范围内误差呈现出一致性。表现出来的一致误差属于系统误差，可能来自于视觉系统的校准误差。计算所有 2400 次测量结果与各自对应的真实结果的平均偏差为

$$(\Delta x, \Delta y) = (-0.010, -0.009) \text{ m} \quad (3.11)$$

然后以此偏差对定位结果进行修正。由于此偏差是在不同的物体和位置上是保持一致的，每次校准视觉系统后，只需选取规则物体进行一次偏差测量即可。对位置偏差进行修正之后，重新计算物体在水平平面内定位结果的平均距离误差为

$$\bar{\epsilon} = 0.003 \text{ m} \quad (3.12)$$

此误差已经可以满足服务机器人进行物体操作时对定位精度的要求，第 3.5.4 小节中将通过实验来证明这一点。

实验还分析了误差在平面内的分布情况。根据物体 2 放置在 $(-0.1, 0.2) \text{ m}$ 处的 200 个测量结果，绘制分布直方图如图 3.10 所示，其中以每个格子的颜色深浅表示分布数目的多少。图中所有的格子中的分布数之和少于 200，因为几个偏差较大的结果未画出。可以看出分布基本以均值为中心成旋转对称。统计在不同范围内的测量值占 200 个测量值的比例如表 3.1 所示。可以看出，此分布十分接近正态分布的特征。这说明测量值的误差是典型的随机误差，可以通过多次测量取均值的方法有效减小。

范围	$\leq \sigma$	$\leq 2\sigma$	$\leq 3\sigma$
测量结果的分布比例 (%)	64.5	96.0	100
正态分布的分布比例 (%)	68.3	95.4	99.7

表 3.1 不同范围内的分布比例

3.5.2. 三维特征提取实验

实验中选取了 3 类不同的物体，使用前文所述算法进行了特征提取，并对结果进行了可视化。物体的照片与特征提取结果分别如图 3.11 左侧和右侧所示，左侧图片的拍摄视角与两个摄像头的其中一个相同。右侧图中，绘制出了经过裁剪、重采样和聚类后的点云，XYZ 标架标记出外部坐标系 Φ ，绿色线条 — 画出了物体的包围盒 OBB ，白色箭头 \Rightarrow 标记了物体的三个取向，箭头上的数字表示该方向物体的尺寸，单位为 m。

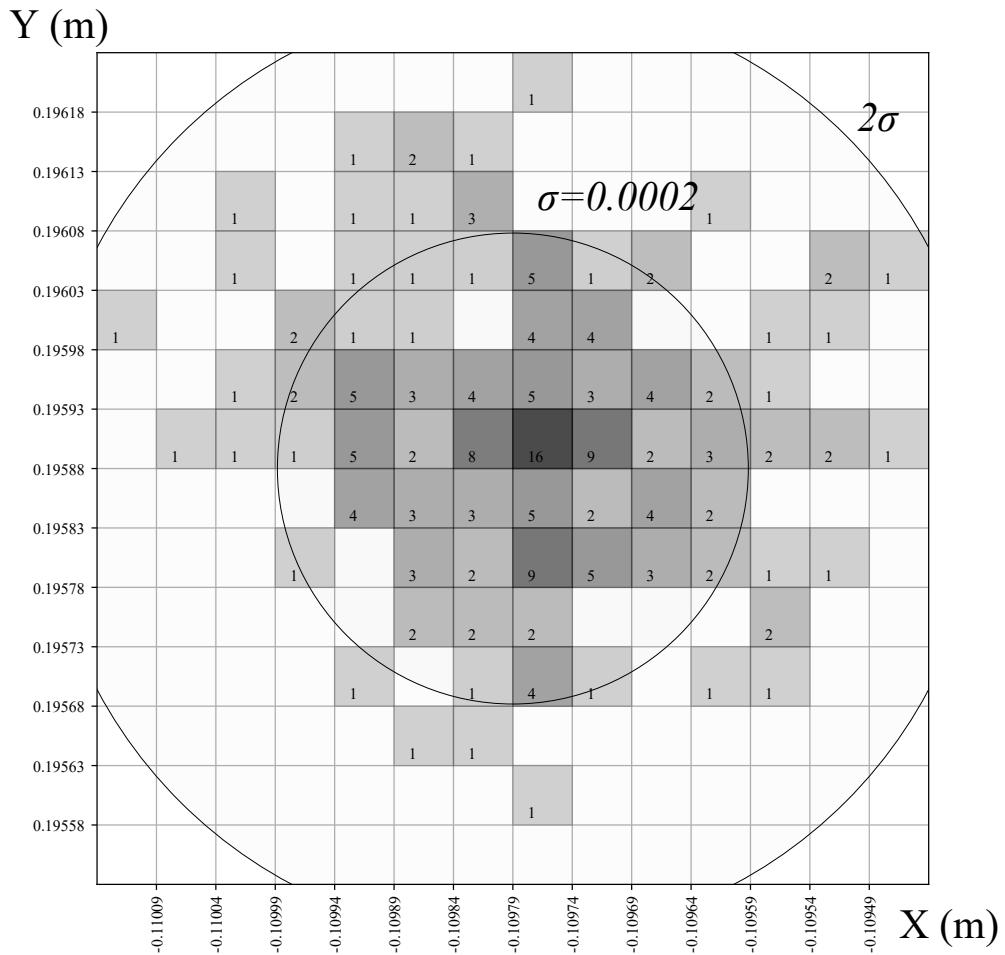


图 3.10 物体定位实验测量结果的分布情况

第一类物体都是旋转对称体。直观上来看，旋转对称体的取向是该物体旋转轴的方向，这和特征提取结果一致。物体的三个尺寸值分表代表了物体在轴向的长度，和物体的半径，这也符合物体的描述特点。第二类物体是几个四方棱柱形的物体，一般而言，这类物体使用长宽高来描述，其取向分别为各条棱的方向。而特征提取的结果符合该特点，三个特征向量的方向沿着棱的方向，且尺寸分别为物体的长宽高。第三类物体是形状较为不规则的物体。这类物体难以用简单几何体描述，但是实验结果仍然基本符合人类对它们的认识。计算得到的包围盒基本是空间中可以取到的最小包围盒，而三个取向方向基本和用于描述这类不规则物体尺寸所用的三个方向一致。实验表明，前文所述的算法可以有效地提取人类和机器人感兴趣的三维特征，这些特征将被用于对物体进行更有效更智能的操作。

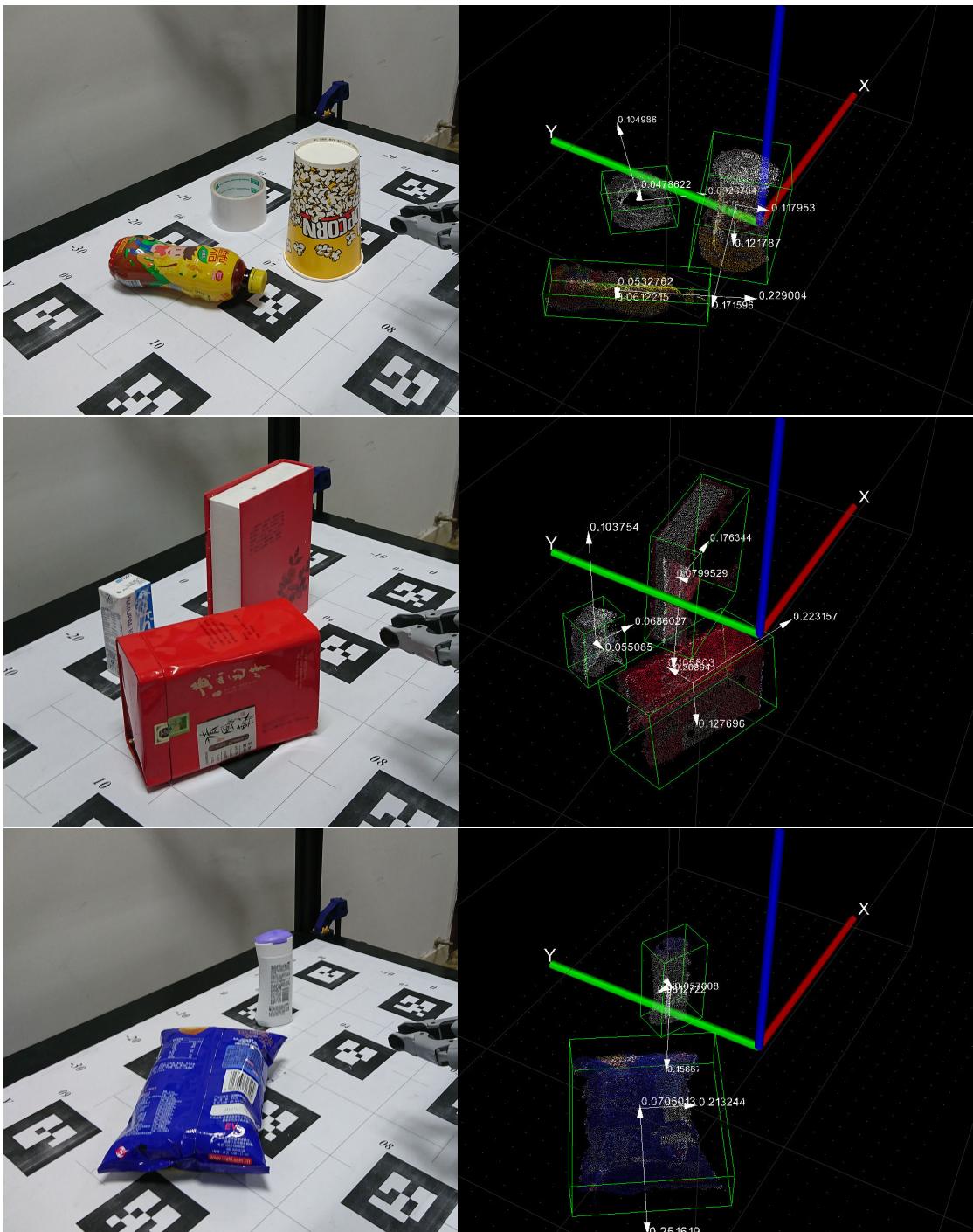


图 3.11 不同物体的三维特征提取结果

3.5.3. 机械臂精度评估

将前文所述的标记物构成的刚体固定在机械臂末端，然后控制机械臂在空间中沿着特定长度的直线移动，分别用动作捕捉测量运动开始和结束时刚体的位置。实验发现，手臂末端实际运动的距离和指定距离的误差约为 2 mm。此误差略小于视觉系统对物体定位的误差。此结果表明，后续使用机械臂进行物体操作实验时，机械臂的精度不会导致导致操作失败。如果操作出现失败，基本可以

认为是视觉系统的定位或者特征提取的造成的。

3.5.4. 物体操作实验

大部分场景下，服务机器人进行物体操作的第一步都是对物体进行抓取。由于物体具有一定的取向，抓取过程应该具有一定的方向。例如，抓取一个水杯时，需要抓在水杯的侧面，这样才能便于完成后续的倒水等操作。这就要求物体以不同的取向摆放在平面上时，机器人需要以不同的方式进行抓取。此外，机器人操作某一个物体时需要避免触碰到其他无关物体。本文在平面上放置了多个物体，物体的形状、尺寸、取向各异。视觉系统对物体完成定位和特征提取，随后机械臂根据这些信息对物体进行抓取。4个典型的抓取动作如图3.12所示。图中(a)、(b)、(c)是物体在侧面被抓取的情况。若物体侧面的尺寸在可以被机械臂抓取的范围内，控制手臂前端垂直于物体最长轴，即最大尺寸对应的特征向量方向，平行于尺寸适于抓取的方向，抓取物体的中心点。这种策略假设物体质量分布均匀，此时可以实现最稳定的抓握，且随后的运动过程中物体在机械臂上不会发生旋转和滑动。图(d)测试了视觉系统提取物体点云中平面的结果。由于机械臂上并未安装吸盘等操作平面的部件，控制机械臂以垂直于平面的方向触碰平面中心点，模拟对平面的吸取操作。

实验中还测试了其他一些物体，整个实验中机械臂对物体进行连续抓取的视频请访问 <http://t.cn/R8MIQWM> 观看。实验中总共测试了十余个常见物体的100次抓取，未出现抓取失败的情况。这表明本章提出的视觉系统可以有效地对物体进行定位和特征提取，配合机械臂可以进行有效地物体操作。

3.6 本章小结

本章详细介绍了使用多个三维摄像头获得物体完整点云并进行定位和提取特征的方法。为了获取完整点云，使用标定板预先校准了摄像头坐标系之间的变换关系。随后，根据点云对物体进行了定位和特征提取。为了检验硬件和算法的有效性和精度，搭建了一套模块化的实验平台，使用机械臂进行了物体操作实验。实验结果表明，视觉系统的定位精度可以满足物体操作的需要。提取出的三维特征符合人类对物体的认知，并且可以有效应用在物体抓取和其他操作当中。

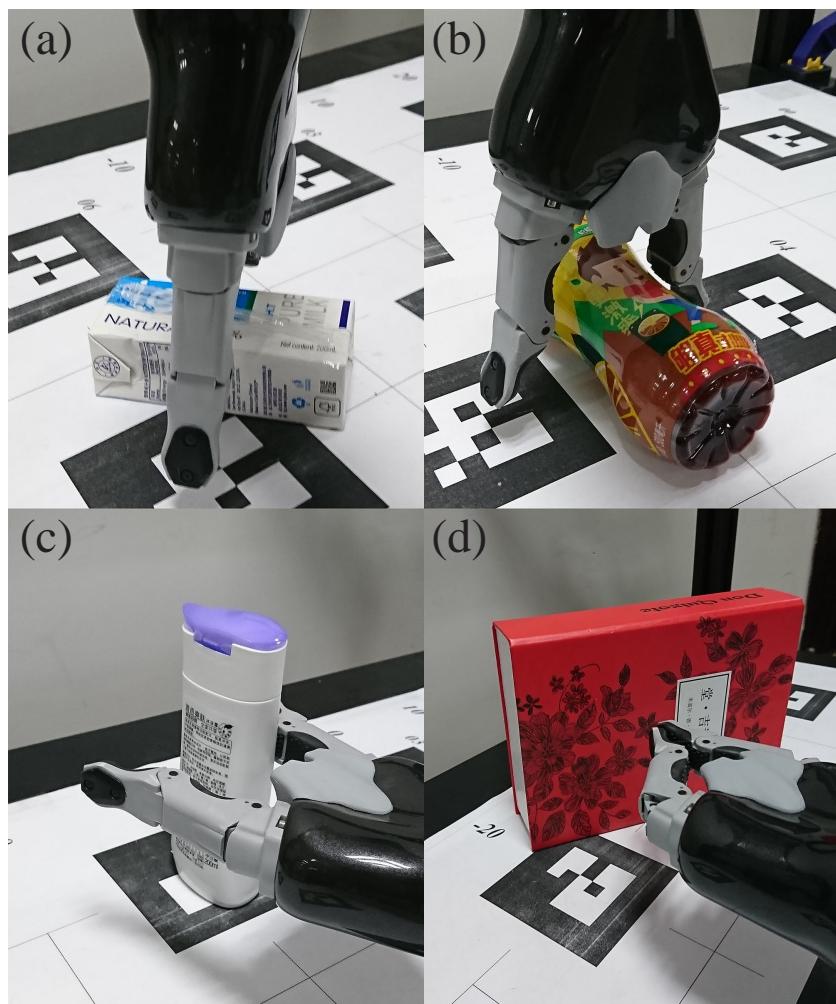


图 3.12 机械臂对物体进行抓取

第4章 总结与展望

4.1 本文工作总结

物体识别一直是机器人视觉领域最重要的研究课题之一。本文面向实际应用场景，提出了基于深度学习的物体分类方法和基于多个三维摄像头的物体定位和识别方法。由于应用场景的不同，需要对现有的方法做较多的修改和创新，才能取得有实用价值的结果。

在标记好类别的图像数据集上的图像分类问题一定程度上已经被解决。现有的最好的分类器的分类正确率已经超过了人类。但是，这些分类器的使用场景都是有严格限制的，而实际中机器人面对的是复杂多变，无法预测的环境。机器人还需要更为精细地分辨物体，而这些物体的预标注数据集是十分有限，甚至需要临时采集的。此时传统的从头训练神经网络的方法会导致严重的过拟合。在这种场景下，需要对卷积神经网络进行修改，减小网络规模；同时采用迁移学习和微调的思路，利用大数据集训练网络提取物体特征的能力，然后对输出的特征训练分类器进行分类。同时，为了应对复杂多变的环境，本文模拟环境的变化，对训练图片进行了随机调整。调整后的图片一定程度上模拟了不同环境中采集到的图片，起到了数据扩充的作用。在训练网络时，采用了热身和学习率规划的方法进一步减小了过拟合。同时，采用了三维摄像头提供的点云信息帮助分割物体和背景。最终的分类器对于环境变化有一定的鲁棒性，识别精度超过了之前使用的基于特征的分类器，且运行速度可以满足服务机器人的要求。

三维物体在二维平面上成像时必然伴随着信息的丢失，在需要知道物体的尺寸、三维形状的场景下，使用三维视觉是必要的。三维摄像头一次只能看到物体的一个侧面，获取物体的完整点云需要进行点云的匹配拼接。基于特征点的点云拼接方法计算复杂，且精度有限。本文使用图案已知的二维码标签，预先校准多个摄像头之间坐标系的变换关系来完成点云匹配。对物体的完整点云，提出了一套预处理和特征提取方法，有效地完成了物体定位，物体尺寸、取向的计算，以及点云中平面的提取。这套方法的精度可以满足服务机器人对物体操作的要求。实验表明，这些三维特征与人类对物体的认知相符合，并成功指导机械臂完成了连续的物体操作。

本文提出的方法主要着眼于服务机器人，尤其是家庭服务机器人的应用场景。这些方法综合了机器人视觉领域多个方向的成果，并可以集成到现有的服务机器人系统当中，提高机器人的智能度和表现。

4.2 未来工作展望

本文的工作仍然处于初步阶段，还有很多可以改进或进一步研究之处。采用一些新技术可能可以进一步提高算法的表现。

本文设计的卷积神经网络分类器只涉及对裁剪好的图片进行分类，而图片的裁剪是通过利用三维摄像头提供的点云信息完成的。这种方法依赖于校准好的三维摄像头和二维相机，且物体图像的分割步骤需要耗费较多的计算资源。最近几年提出的基于区域的卷积神经网络模型，例如 R-CNN^[34]、Fast R-CNN^[35]、Faster R-CNN^[36]，具有直接从图片中定位物体并识别的能力。这类方法首先在图片中裁剪区域候选，评估候选区域属于物体的可信度，然后对可信的区域运行卷积神经网络分类器进行分类识别。和卷积神经网络分类器类似，这些模型需要用方框标记好物体位置，且给出物体类别的数据集进行训练。如果使用三维摄像头的点云数据对物体进行标记，然后使用 R-CNN 进行训练，就可以得到能够检测和识别物体的分类器。此分类器可以运行在不具有三维摄像头的机器人上。但是，R-CNN 的运行需要更多计算资源，所以可能会产生识别过慢的问题。具体实现与评估可以在未来的工作中进行尝试。

本文的方法得到物体的完整点云后，只计算了位置、取向、尺寸、包含的平面等信息，这实际上只利用了点云信息的很小一部分。用类似于提取平面的方法，可以从点云中寻找圆柱面、拐角、尖角等等部位的信息，这些信息都可以帮助机器人更好地操作物体。现有的机器人操作物体的策略是根据人类经验人工构建的，这些策略可能不是最适合于机器人的策略。未来可以采用物理建模的方法，计算在何种情况下机器人的机械臂末端可以最紧密地贴合物体，从而得到最佳的抓取策略。也可以使用强化学习的方法，在模拟器中模拟机器人操作物体的过程，训练出成功率最高的策略。此外，中国科学技术大学多智能体实验室也开发了新一代的带有多个吸盘的柔性机器人抓取器^[37]，此抓取器可以利用物体的表面形状信息，找到最适合的抓取位置和方式进行抓取。

参 考 文 献

- [1] RUSSELL S J, NORVIG P. Artificial intelligence: A modern approach[M]. 2nd ed. [S.I.]: Pearson Education, 2003.
- [2] INGLE D, GOODALE M, MANSFIELD R. Analysis of visual behavior[M]. [S.I.]: MIT Press, 1982.
- [3] ZHANG Z. Microsoft kinect sensor and its effect[J]. IEEE MultiMedia, 2012, 19(2): 4–10. DOI: 10.1109/MMUL.2012.24.
- [4] ROBERTS L G. Outstanding dissertations in the computer sciences: Machine perception of three-dimensional solids[M]. [S.I.]: Garland Publishing, New York, 1963.
- [5] HARALICK R, SHANMUGAM K, DINSTEIN I. Texture features for image classification [J]. IEEE Transactions on Systems, Man, and Cybernetics, 1973, 3(6).
- [6] LOWE D G. Object recognition from local scale-invariant features[C]//ICCV '99: Proceedings of the International Conference on Computer Vision-Volume 2 - Volume 2. Washington, DC, USA: IEEE Computer Society, 1999: 1150–.
- [7] BAY H, TUYTELAARS T, GOOL L V. Surf: Speeded up robust features[C]//In ECCV. 2006: 404–417.
- [8] RUMELHART D E, HINTON G E, WILLIAMS R J. Learning representations by back-propagating errors[J]. Nature, 1986, 323(6088): 533–536.
- [9] LECUN Y, BOTTOU L, BENGIO Y, et al. Gradient-based learning applied to document recognition[C]//Proceedings of the IEEE. 1998: 2278–2324.
- [10] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. Imagenet classification with deep convolutional neural networks[M]//PEREIRA F, BURGES C J C, BOTTOU L, et al. Advances in Neural Information Processing Systems 25. [S.I.]: Curran Associates, Inc., 2012: 1097–1105.
- [11] SRIVASTAVA N, HINTON G, KRIZHEVSKY A, et al. Dropout: A simple way to prevent neural networks from overfitting[J]. Journal of Machine Learning Research, 2014, 15: 1929–1958.
- [12] SIMONYAN K, ZISSERMAN A. Very deep convolutional networks for large-scale image recognition[J]. CoRR, 2014, abs/1409.1556.
- [13] SZEGEDY C, LIU W, JIA Y, et al. Going deeper with convolutions[C]//Computer Vision and Pattern Recognition (CVPR). 2015.
- [14] HE K, ZHANG X, REN S, et al. Deep residual learning for image recognition[J]. CoRR, 2015, abs/1512.03385.
- [15] IOFFE S, SZEGEDY C. Batch normalization: Accelerating deep network training by reducing

- internal covariate shift[J]. CoRR, 2015, abs/1502.03167.
- [16] RUSU R B, BRADSKI G R, THIBAUX R, et al. Fast 3d recognition and pose using the viewpoint feature histogram.[C]//IROS. [S.l.]: IEEE, 2010: 2155–2162.
- [17] SCOVANNER P, ALI S, SHAH M. A 3-dimensional sift descriptor and its application to action recognition[C]//MM '07: Proceedings of the 15th ACM International Conference on Multimedia. New York, NY, USA: ACM, 2007: 357–360. DOI: 10.1145/1291233.1291311.
- [18] RUSU R B, COUSINS S. 3d is here: Point cloud library (pcl)[C]//In Robotics and Automation (ICRA), 2011 IEEE International Conference on. [S.l.]: IEEE: 1–4.
- [19] PAN S J, YANG Q. A survey on transfer learning[J]. IEEE Trans. on Knowl. and Data Eng., 2010, 22(10): 1345–1359. DOI: 10.1109/TKDE.2009.191.
- [20] KAISER L, GOMEZ A N, SHAZEER N, et al. One model to learn them all[J]. CoRR, 2017, abs/1706.05137.
- [21] RUSSAKOVSKY O, DENG J, SU H, et al. ImageNet Large Scale Visual Recognition Challenge[J]. International Journal of Computer Vision (IJCV), 2015, 115(3): 211–252. DOI: 10.1007/s11263-015-0816-y.
- [22] GLOROT X, BENGIO Y. Understanding the difficulty of training deep feedforward neural networks[C]//In Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS' 10). Society for Artificial Intelligence and Statistics. 2010.
- [23] OLAH C, MORDVINTSEV A, SCHUBERT L. Feature visualization[J]. Distill, 2017. DOI: 10.23915/distill.00007.
- [24] ZHANG Z. A flexible new technique for camera calibration[J]. IEEE Trans. Pattern Anal. Mach. Intell., 2000, 22(11): 1330–1334. DOI: 10.1109/34.888718.
- [25] FISCHLER M A, BOLLES R C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography[J]. Commun. ACM, 1981, 24 (6): 381–395. DOI: 10.1145/358669.358692.
- [26] GOYAL P, DOLLÁR P, GIRSHICK R B, et al. Accurate, large minibatch SGD: training imagenet in 1 hour[J]. CoRR, 2017, abs/1706.02677.
- [27] QUIGLEY M, CONLEY K, GERKEY B P, et al. Ros: an open-source robot operating system [C]//ICRA Workshop on Open Source Software. 2009.
- [28] ABADI M, AGARWAL A, BARHAM P, et al. TensorFlow: Large-scale machine learning on heterogeneous systems[EB/OL]. 2015. <https://www.tensorflow.org/>.
- [29] SARBOLANDI H, LEFLOCH D, KOLB A. Kinect range sensing[J]. Comput. Vis. Image Underst., 2015, 139(C): 1–20. DOI: 10.1016/j.cviu.2015.05.006.
- [30] HENRY P, KRAININ M, HERBST E, et al. Rgb-d mapping: Using kinect-style depth cameras for dense 3d modeling of indoor environments[J]. Int. J. Rob. Res., 2012, 31(5): 647–663.

- DOI: 10.1177/0278364911434148.
- [31] SILTANEN S, HAKKARAINEN M, HONKAMAA P. Automatic marker field calibration [C]//2007.
- [32] SUCAN I A, CHITTA S. Moveit![EB/OL]. <http://moveit.ros.org>.
- [33] CHOO B, LANDAU M J, DEVORE M D, et al. Statistical analysis-based error models for the microsoft kinect™ depth sensor[C]//Sensors. 2014.
- [34] GIRSHICK R, DONAHUE J, DARRELL T, et al. Rich feature hierarchies for accurate object detection and semantic segmentation[C]//CVPR '14: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition. Washington, DC, USA: IEEE Computer Society, 2014: 580–587. DOI: 10.1109/CVPR.2014.81.
- [35] GIRSHICK R. Fast r-cnn[C]//ICCV '15: Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV). Washington, DC, USA: IEEE Computer Society, 2015: 1440–1448. DOI: 10.1109/ICCV.2015.169.
- [36] REN S, HE K, GIRSHICK R, et al. Faster r-cnn: Towards real-time object detection with region proposal networks[C]//NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1. Cambridge, MA, USA: MIT Press, 2015: 91–99.
- [37] LIN N, WU P, TAN X, et al. Design and analysis of a novel sucked-type underactuated hand with multiple grasping modes[C]//International Conference on Robot Intelligence Technology and Applications. 2017.

致 谢

在中国科学技术大学多智能体实验室度过的近三年的时间里，我得到了多位老师和同学的指导和帮助。在这里，我希望向他们表达我诚挚的谢意。

首先要感谢我的导师，也是实验室的创始人，中国科学技术大学计算机学院的陈小平教授，是他将我引入了人工智能和机器人的大门。自我 2015 年加入实验室以来，陈老师一直对我进行了悉心的指导，让我从一个门外汉成长为一个研究者。陈老师以极大的决心和毅力从无到有建立了多智能体实验室，开发了可佳机器人，并领导实验室多次在国际和国内的重大赛事中夺得桂冠。我作为可佳机器人研究组的成员，参与了 2015 年到 2017 年间的三次机器人世界杯，以及多次国内赛事和交流活动，并从中学到了很多课堂上无法接触到的知识。这些活动开阔了我的眼界，锻炼了我的能力，磨练了我的意志，也增进了我和实验室其他成员间的感情。陈老师也乐于和我们分享他的研究心得与人生感悟，让我们受益良多。陈老师三年来对我的指导将是我一生的宝贵财富。

实验室的吉建民老师和吴锋老师作为科研道路上的前辈和实验室的开创成员，一直对我的学习和研究给予了帮助。当我在研究中遇到瓶颈时，他们总能提供宝贵的建议，常让我有豁然开朗之感。实验室的程敏师兄、陈凯师兄、赵哲师兄、陈羸峰师兄、卢栋才师兄、唐可可师兄、王宁扬师兄和刘松师兄在学习、研究和生活中给予了我宝贵的帮助和关心，谢谢你们！感谢和我一起参加比赛的帅威、刘江川、周锋、王希平、唐冰、陈广大、崔国伟、郑魁松、张钊、林楠、晋忠孝等各位同学，和你们合作解决问题的经历将让我回味终生。

自从 2010 年进入中国科学技术大学以来，我不仅学习到了知识，还认识了一群优秀的青年人才，感谢各位为我传道授业解惑的老师，感谢一直以来陪伴我的各位同学和朋友。感谢我的本科班主任李娜颖老师，她帮助初入校园的我适应了大学的学习和生活，我读研后也给予了我大量的帮助。感谢本科毕业论文的指导老师，中国科学技术大学微尺度国家实验室的赵爱迪教授，虽然我毕业后没有继续从事化学领域的学习，从赵老师那里学习的科研方法仍然一直引导着我。感谢 2015 级科学硕士班的班主任王行甫老师和教学秘书张荣老师，他们在各种事务上给予了我很大的帮助。和你们在合肥一起度过的这 8 年是我一生最快乐的时光。

最后感谢一直在背后支持我的父母和家人，无论我做出什么决定，遇到了何种困难，他们一直是我最坚实的后盾。

张泽坤

2018 年 3 月 26 日于安徽合肥

在读期间发表的学术论文与取得的研究成果

已发表或待发表论文

1. 张泽坤, 唐冰, 陈小平 *. 面向物流分拣的多立体摄像头物体操作系统 [J].
计算机应用, 2018. (**ZHANG Zekun, TANG Bing, CHEN Xiaoping***. Object manipulation system with multiple stereo cameras for logistics applications[J]. Journal of Computer Applications, 2018.) (已接收, 择期发表)

其他成果与活动

1. 在 2015 年 RoboCup 机器人世界杯比赛（合肥）中作为蓝鹰队核心成员获得 Benchmarking Service Robot 分组冠军，获得 @Home 分组亚军。
2. 在 2016 年 RoboCup 机器人世界杯比赛（莱比锡）中作为蓝鹰队核心成员获得 @Home 分组第三名。
3. 在 2017 年 RoboCup 机器人世界杯比赛（名古屋）中作为蓝鹰队核心成员获得最佳操作奖。