

2021-2022 学年春季学期
《人工智能进展》(0869EY01)

课程报告

成绩

序 号		学 院	计算机工程与科学学院
学 号	20123101	姓 名	李昀哲
报告题目	智能机器人对非线性移动目标的跟踪和预测		
课程 报告 成绩	1、选题（20%）：有前沿性和应用前景。		
	2、报告主要部分（60%）： (1) 对所研讨领域的相关研究工作进行较全面的综述，相关研究要做到查新、查全，有论述有评价； (2) 报告主要内容叙述条理清楚、内容详细； (3) 有相关技术的案例分析，鼓励对研讨内容的动手实践； (4) 案例分析不直接粘贴所有代码； (5) 代码有注释或说明； (6) 鼓励有原创性工作（技术或算法）； (7) 有个人想法、观点表达清楚。		
	3、书写格式（20%）： (1) 书写规范、用词正确、无明显的错别字； (2) 图表、代码清晰规范，格式协调、效果好； (3) 参考文献书写（不少于 10 篇参考文献，英文文献不少于 2 篇）、引用规范合理。		
报告评语：			
教师签名：			
日期： 年 月 日			

智能机器人对非线性移动目标的跟踪和预测

学号 20123101 姓名 李昀哲

摘要：本课程报告基于本人正在参与的全国大学生机器人大赛（RoboMaster 机甲大师赛）研发的项目，结合自身经历和对智能机器人进展的调研成果：介绍、分析人工智能浪潮下，智能机器人的国内外发展现状；从机器人视觉这一关键技术为切入，介绍此项技术的核心任务以及本人近期项目中实际研发、使用的对非线性移动目标的跟踪和预测算法，通过工业相机、移动运算平台联合工作，将计算结果和机械结构通过通信有机交互，实现智能机器人感知周边环境、“自主”预测、精确打击目标的特点。此项技术的进一步发展可以有效应用于智能安保、挖掘勘探等应用领域，结合无人机、雷达等技术，还将于军工领域有潜在发展可能。

关键词：智能机器人；机器人视觉；目标跟踪；移动预测

1 引言

在科技水平和市场需求的共同推动下，智能机器人的应用不断向日常领域、新兴领域拓展。在工业领域，智能机器人逐步代替传统人力生产，极大地降低了人力资源和成本，也能适应时代需求下更高精度、更快速度的生产要求；在面向医疗、家庭等领域的智能机器人也不断进步，由前些年的智能扫地机器人到近年基于交互式机器人的智能语音机器人。随着功能的不断丰富，智能机器人不仅展现了高速的科技发展，更从不同程度上解决、缓解了：工业中部分加工风险大、难度高的问题；生活中老人、孩子，尤其是现在疫情之下因隔离原因无人沟通、心里抑郁的问题；医疗领域微创手术难度大、精细度要求高的问题。

由 ABI Research 调查研究结果显示，亚太地区的智能机器人市场份额正逐步加大，重心也由欧美转向亚太，份额占比从原先的 19% 提升至 57%。我国的政策也旨在培养多学科、高科技交叉融合人才，智能机器人正是这样的载体之一，近些年，国家提出“中国制造 2025”战略发展目标连同“十四五”发展战略规划^[1]，充分体现了我国大力发展智能机器人的决心和态度。



图 1 ABI Research 亚太地区智能机器人市场份额调查

随着智能机器人的不断普及，这也逐渐成为当代青少年、大学生的科技启蒙，小到 LEGO™ 的积木拼搭、简单程序控制，大到大疆 DJI 以高校为平台，推出的全国大学生机器人 RoboMaster 机甲大师赛，制作一个

自己的机器人已经成为触手可及的科技。我正是这样的教育模式下的受益者，第一次接触机器人是初二游学 Stanford 时，见到表哥的一个 Mechanical Engineering 机器人项目，当时我就找到了我的兴趣所在，这种团队合作、设计创新的方式吸引了我。高中阶段有幸参加了学校的机器人社团，开始对机械结构、蓝牙通信、程序设计有了初步的了解，还通过自己和团队的努力，历史性地连续两年代表中国参加在底特律的 FIRST 世界锦标赛。大学阶段，仍旧保持初心，成功考入心仪的计算机专业，同时加入了上海大学的 RoboMaster 机器人战队。不断地自己动手尝试、设计，根据需求完善自己的结构、程序，对于工科学生来说，这无疑是最好的实践方式，在实践中创新，而非在淹没在书本的教条中，我认为，我国的教育也正随着智能机器人的发展大跨步的前进。

2 国内外发展现状分析

世界第一台智能机器人 Shakey 研制于 1966 年至 1972 年间的 Stanford 研究所，如图 2 所示，Shakey 首次全面应用了人工智能技术，装备了电子摄像机、三角测距仪、碰撞传感器以及驱动电机等，能简单解决感知、运动规划和控制问题。在今天看来，智能机器人 Shakey 简单又笨拙，但不可否认的是，它是当时将 AI 应用于机器人中最为成功的案例，证实了许多人工智能领域严肃科学的结论，在实现过程中获得的成果也影响了很多后续研究。



图 2 Shakey Bot 和创始人 Charlie Rosen

早在 1967 年，我们的近邻日本就从率先发展机器人的美国引进技术，于 1968 年研制第一台川崎机器人，虽然那时以工业机器人为主，但较早的技术研发和使用，为日本智能机器人产业奠定坚实基础。

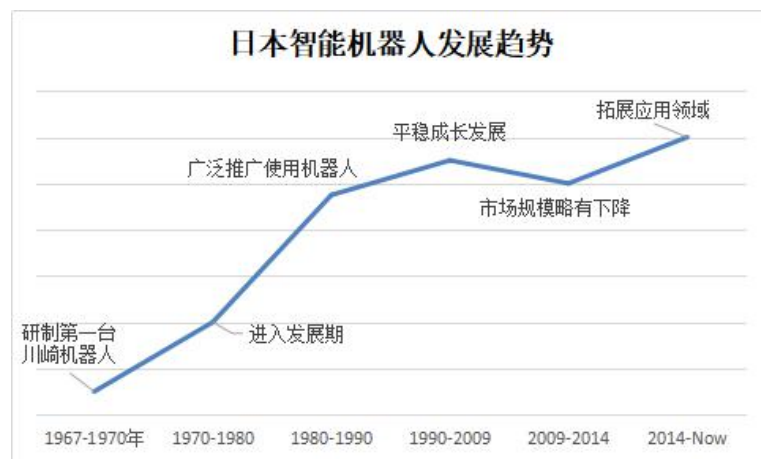


图 3 日本智能机器人发展趋势

如图 3 所示，为日本智能机器人发展的趋势折线图。在 2014 年，以日本、美国、英国为首的国家，就结合自己的生产力，就让全球的工业机器人的销售额增长了 11.5%^[1]。在欧洲等地区，也启动了“SPARC”研发计划，并且投入了较大的资金和精力。智能机器人在工业、医学、农业、建筑业、军事等领域都有所应用。因此，探究关键部件和核心技术的发展，完成机器人网络化，实现更好的交互方式，保证人与机器之间信息的协调性与个性化发展方式，具有重要意义^[2]。

而通过调研发现，我国从 2010 年开始，才积极投入到机器人产业的发展中，虽然起步较晚，但发展速度惊人，如图 4 所示，从 2012 年-2017 年，5 年来的平均增速接近 60%，市场规模和需求双双占据全球首位。智能机器人如前文提到，是多学科、高科技交叉融合的产物，作为工业后发展国家，中国在产业基



图 4 中国近 10 年智能机器人发展趋势

础上相对较弱。在机器人产业中，上游核心零部件直接影响机器人性能，且占本体成本 70% 以上，但目前我国核心零部件仍依赖进口，如图 5 所示，我国在传感器市场所占份额极少，因此将在零件成本上花费较多的资金，且无法掌握和使用核心技术。相比于国外如今更低的元件价格、电子器件、传感器，更大的数据集，我国这些方面的现状谈不上乐观。同时，我国机器人先进技术（包括自动化、通信和计算机领域），同样发展滞后，虽正在大跨步前进，但尚未实现全球领先，与国外成熟的工业、家用智能机器人相比，仍有不小差距，高端市场显著缺失。

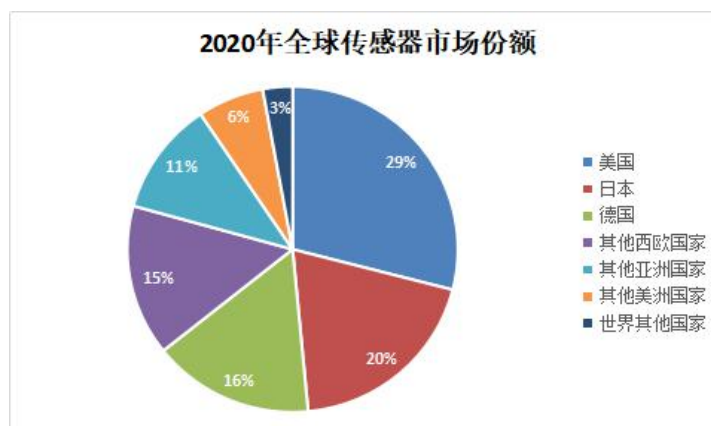


图 5 2020 年全球传感器市场份额

目前的智能机器人电子商务模式普遍为 B2B 或 B2C，相较于国外的私人企业家利用自身资源发展前沿技术，如：火星勘探、高智能类人机器人等，中国的 B（usiness）端，貌似对于市场尚未明确应用方向，本质来说，还是受到人机交互、功能仿生、智能感知等技术的限制。而 C（ustomer）端又面临着产品价格高，功能安全与信息安全难保证的问题，高价格门槛使得服务机器人很难进入大众化消费时代。

但尽管如此，我国还是在全力加强交叉学科人才培养，改善教育体制，加大政策扶持，市场潜力正在快速上升。近年，利用机器人视觉、定位技术等，我国研究人员也取得众多成果，如表 1 所示：华中科技大学的王敏等人将超声波测距与视觉技术相结合，用梯度算子描述物体边缘特征，通过超声技术完成定位，实现工件的自动识别抓取^[3]。西南科技大学的张华等人提出基于双目视觉的机器人自定位方法，通过提取图像 Harris 角点特征计算 Sobel 响应，采用绝对误差累计的最小窗口完成匹配，使用改进的 RANSAC 算法优化特征匹配，最后通过高斯牛顿迭代方法获取机器人位姿，完成机器人的自定位研究^[4]。北京交通大学的 Haikuo Shen 等人将激光测距与双目视觉系统相结合，提出多传感器信息融合的自适应算法，提高了多传感器的利用效率和测量精度^[5]。哈尔滨工业大学的韩博通过决策融合和全局最小生成树完成对目标工件的识别与匹配，并通过视差图获取到工件三维坐标，完成目标工件的识别与定位^[6]。

表 1 近年智能机器人基于机器人视觉国内部分成果

研究者	单位	研究成果
王敏、黄心汉等	华中科技大学	工件的自动识别抓取
张华等	西南科技大学	通过高斯牛顿迭代方法获取机器人位姿
Haikuo Shen 等	北京交通大学	多传感器信息融合的自适应算法
韩博等	哈尔滨工业大学	目标工件的识别与定位

Tesla, Google, Boston Dynamics, Honda's Asimo 等企业都是智能机器人领域耳熟能详的领军企业，从相关文献、自媒体平台等了解到他们的工业机器人效率高、功能强；高级智能机器人已可以完成跑、跳、翻滚等动作，还能根据交互、周边环境变化做出自己的推理，所需的人类控制需求逐渐减弱。如表 2 所示，同样以智能机器人的机器人视觉为例，外国研究人员同样做出如目标跟踪、目标抓取等成果。

表 2 近年智能机器人基于机器人视觉国外部分成果

研究者	单位	研究成果
Asada Minoru 等	日本大阪大学	预测未知运动物体的运动，完成避障
Plinio Moreno 等	葡萄牙里斯本技术大学	完成对目标物体的抓取动作
Umar Manzoor 等	杜兰大学	实现机器人的三维感知
法国研究人员	Aldebaran Robotics 公司	引导机器人实现抓取

从各类技术文献中，也不难发现，国内外目前聚焦于导航与定位、路径规划、人机接口设计、机器人视觉、多传感器信息融合、智能控制等技术的突破。通过调研期间文献检索发现，如图 6 所示，目前的国内外智能机器人路径规划、机器人视觉已有成熟稳定的技术基础；人机接口设计也在逐步发展；智能机器人控制在 Web Of Science 网站积累的技术文献较多，而国内技术文献积累较少；多传感器融合技术的文献积累在国内外均是占比最少的技术，这也是当代智能机器人追求高智能化的重点技术。

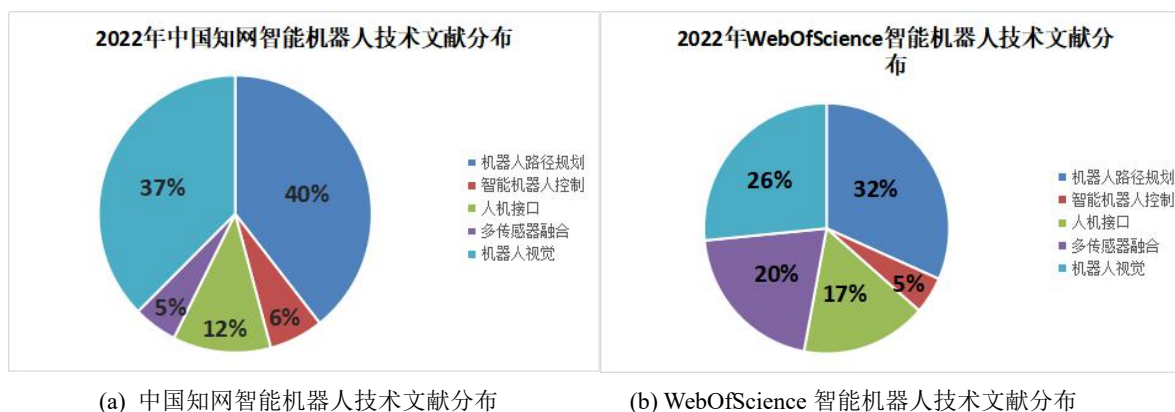


图 6 国内外文献网站智能机器人技术文献分布

3 智能机器人对目标的跟踪及预测关键技术

人工智能是发展智能机器人的关键，智能机器人之于一般机器人的差别，就在于智能机器人会通过类人的感知系统，对周边的环境进行感知从而通过推理产生下一步动作的指令。而人工智能的三大核心技术：感知技术、认知技术、行为控制技术又是重中之重。其中的感知技术中较为成熟和应用较广的是机器人视觉技术，由上述综述中的内容也可以看出机器人视觉在科研领域也有长足的技术积累。

机器视觉是一门复杂的技术，其融合了视觉系统构建与标定、数字信号处理、图像处理算法、计算机应用技术等多方面知识，通过传感器获取图像信息，经信息处理后将图像信息转化为数字信号，再进行显示或控制动作。如图 7 所示^[7]，视觉技术目前被广泛地应用到工业生产、航空航天、城市交通、医学医疗、农业 采摘等相关领域^[8-9]。由上述表 1，表 2 的机器人视觉领域的成果可以得出，对于移动目标的跟踪和预测有较好的应用前景和技术积累。

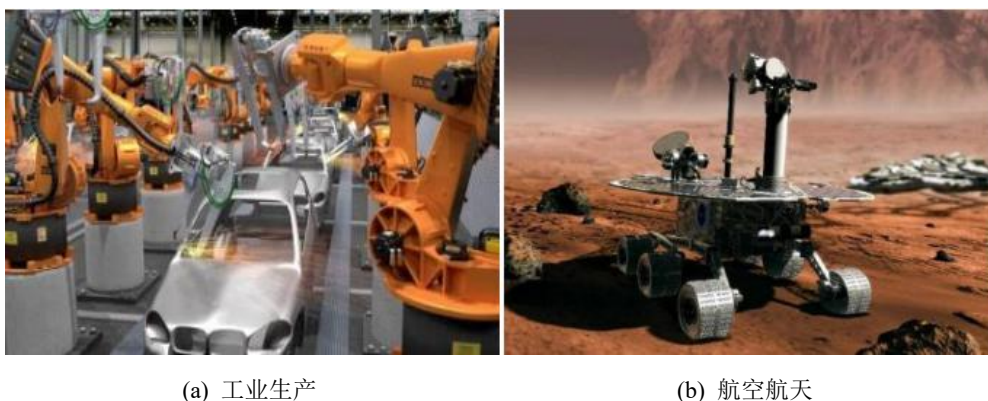


图 7 机器人视觉技术在相关领域中的应用

因此，结合个人兴趣点、技术应用前景、个人项目等因素，本节将介绍机器人视觉中，对移动目标的跟踪及预测技术。将从相机的标定技术和目标跟踪预测算法两个角度介绍，并简要介绍在智能机器人上的部署。

3.1 镜头的选择与相机标定

对于不同环境条件、功能需求的机器人视觉应用场景，图像的所需的条件也是不同的，因此我们常关心作为机器人“慧眼”的镜头和相机选择。例如对于电子警察就需要长焦镜头捕捉远距离目标；对于疫情之下的“数字哨兵”就需要短焦识别近距离物体。而对于相机而言，其参数涉及像素、传感器等，传感器分为 CMOS 和 CCD，目前较常使用的是 CMOS 全局快门。实际使用时，对于曝光时间、增益、白平衡、相机帧率等相机参数调整，来适应不同的需求。

我们往往也对识别到物体在三维空间中的信息：如角度、距离等十分关心，而这些信息的来源都与相机有关。因此，在开发智能机器人相关的视觉算法之前，选择合适的相机和建立相机成像几何模型尤为重要。

机器人视觉的基本任务就是从相机获取的图像信息出发计算三维空间中物体的几何信息，并由此重建和识别物体，而空间物体表面某点的三维几何位置与其在图像中对应点之间的相互关系是由相机成像的几何模型决定的，这些几何模型参数就是相机参数。简单来说，标定是为了能够从空间点的像素坐标映射到世界坐标，这是 3D 立体视觉必须经过的过程。

成像模型一般包括实体、镜头、感光元件和输出，实体通过镜头将光点投射到感光元件上，感光元件将光信号转化为电信号得到数字图像。数据图像常以矩阵的形式在计算机中进行操作，因此标定的过程往往就是进行空间扭曲、矩阵运算的过程。

这里直接给出由世界坐标系到像素坐标系的转化，其原理如图 8 所示，在此不做过多的推导，本质就是根据刚体变换和平移进行矩阵乘法。

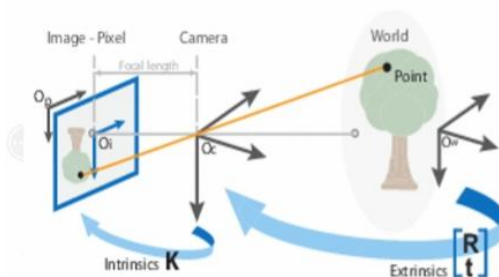


图 8 相机模型成像原理

世界坐标系到像素坐标系的关系，满足以下关系：

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K[R|t] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

其中, $[u, v, 1]^T$ 表示图像点, K 表示相机内参, R 表示相机旋转, t 表示相机平移, $[x, y, z, 1]^T$ 表示空间点。

标定的意义在于得到上述 K ，相机内参矩阵。而相机的位姿（旋转和平移）往往是通过机器人完成装配后，相机和陀螺仪的相对位姿确定的。标定的方法有许多种，目前比较常用的是张正友标定法。标定工具的选择有：**OpenCV 标定**、**Matlab Camera Calibrator 标定工具**等。

3.2 目标跟踪和预测算法

由于智能机器人常处于较为复杂的环境中，观测到的环境信息量也较大，我们常关心根据不同的功能需求对特定的目标进行跟踪。传统的目标跟踪算法，如表 3 所示，常采用光流法、卡尔曼滤波、粒子滤波等方法，但局限性也较为明显，集中体现在对复杂环境的处理效率不够。这就使得这些手段无法较好地应用于面向高智能的机器人，因此，对于跟踪速度和精度的不同要求，常考虑使用滤波类算法和深度学习类算法。

表3 传统目标跟踪预测方法分析

方法名称	计算方法	局限
光流法	基于特征匹配、频域信息	计算量大，实时性较差
卡尔曼滤波	利用动态信息去除噪声	对非线性系统、系统噪声不符合高斯分布的预测较差
粒子滤波	基于强化学习的优化	需要大量样本数量，环境较复杂时，算法复杂度较高

基于相关滤波(Correlation Filter)的目标跟踪预测算法是目标跟踪领域的一大研究热点，为目标跟踪领域带来重大变革。相关滤波源于信号处理领域，相关表示两信号相似性的高低，两信号越接近，相关响应越高。相关滤波跟踪算法就是通过建立一个相关滤波器，来寻找响应值最高的目标位置。2010 年, Bolme 等人率先将相关滤波与目标跟踪相结合，提出了 MOSSE 跟踪算法^[10]，以其高速的跟踪速度和良好的跟踪性能。

深度学习作为近年来的热门技术手段，在众多领域都能看到它的身影。深度学习凭借优异的特征建模

能力在目标跟踪领域也取得了广泛应用。基于深度学习的目标跟踪大致可分为两类，一类是采用卷积神经网络提取目标特征，然后与其他跟踪方式进行融合来实现目标跟踪。另一类是训练出端到端的神经网络模型，目标跟踪的所有步骤均由神经网络来实现。基于相关滤波和深度学习的方法侧重点不同，如表 4 所示。

表 4 不同跟踪预测算法特点对比

方法	跟踪速度	精度
传统目标跟踪预测	慢	低
基于相关滤波目标跟踪预测	快	较高
深度学习目标跟踪预测	较快	高

处理的基本流程如图 11 所示，其中的具体步骤涉及的模型和方法会因项目需求不同而有不同设计。

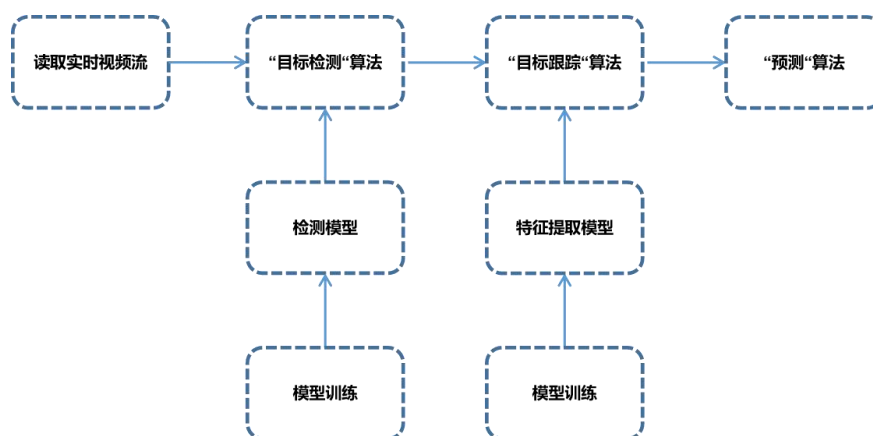


图 11 跟踪预测算法基本流程

读取视频文件后，对于每帧，每个检测到的目标，先利用卷积神经网络提取表观特征（可以理解为该目标的一种特征编码），然后计算每个目标特征之间的余弦距离，进行关联匹配。同时又分为以下三种情况：若跟踪成功，则在上一帧的 n 个目标中找到本次检测到的目标；若新目标出现：在上一帧中的 n 个目标中未找到本次检测到的目标；若目标消失，在上一帧中存在某个目标，但这一帧中没有与之关联的目标。

识别到多目标的情况对于智能机器人而言实际应用中非常普遍，下一节将根据近期实践的项目介绍实际应用中使用的算法和决策手段。

3.3 代码在智能机器人上的部署

首先是运算平台。运算平台是智能机器人的“大脑”，而日常的程序测试和开发一般在个人电脑上进行，但实际机器人使用时无法、也不能将个人电脑附载在机器人上，因此需要更小规模但同样具备优秀计算能力的运算平台。企业产品会对运算平台有更高的要求 and 设计，但对于学生阶段的开发而言，一般使用已有的运算平台，如：Nvidia Jetson nx、Intel NUC 系列等。完成依赖库、环境的配置后即可使用。

其次是串口通信。串口通信一般只需要三根线：地线（非必须），发送线，接收线。串口是差分信号，资源占用比较少，串口通信的三个重要参数：波特率、数据位、停止位，有时还有奇偶校验位。通过运算平台发送数据给控制板，对机械结构（如云台、底盘等）进行控制。

最后，需要考虑由于供电问题导致的电压较低，算力下降的问题。针对不同的项目需求做出相应的调整，对于预测而言，算力、帧率下降会有较大的影响。故对于保证下限电压，就需要优化供能方面的设备。

4 智能机器人对移动目标的跟踪及预测的实现

4.1 项目简介

全国大学生机器人 RoboMaster 机甲大师赛，是共青团中央和 DJI 大疆发起的机器人赛事，采用射击对抗的形式，针对机器人技术的各类领域激发不同专业大学生的深入探索和学术研究，旨在培养前沿的机器人技术人才。赛事对抗内容主要为各高校自主设计机器人（包含机械、电控、视觉等方面），赛场以 FPV 第一视角形式，多兵种（步兵、无人机、英雄等）协同作战，与对手射击、战略对抗，夺取战略点赢下比赛。官方宣传视频可以更快了解此项赛事：[【宣传片】三分钟带你了解 万千青年工程师的舞台——RoboMaster 机甲大师赛 哔哩哔哩 bilibili](#)

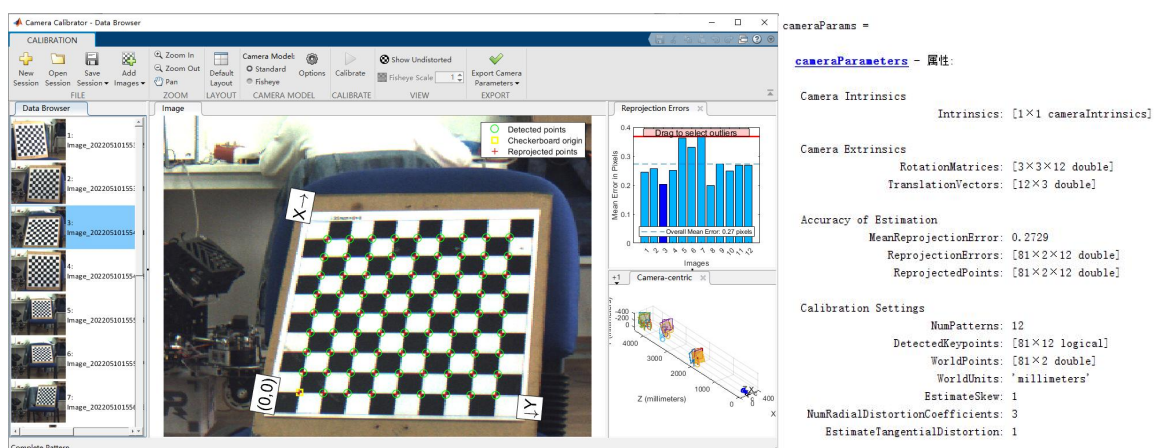
我所负责的部分为视觉算法，主要职责为：识别所需击打的目标（不仅局限于敌方机器人、还包括战场战略资源的控制）并对其的运动进行预测，将云台所需转动的角度发送给控制板，控制板通过嵌入式算法对机械结构进行控制完成发弹。

原先我仅把它当作计算机视觉，但实际接触后发现，虽说算法是基于计算机视觉，但整个开发的过程是以机器人为载体的，是实际应用的产品。因此还需要考虑更多的环境、性能因素，往往会产生意料之外的局限，比如：如何处理电源状态不好时帧率降低的问题、如何处理移动运算平台和控制板通信不同步等问题。这些都是分散却必需的知识和技术，因此，我认为在这个项目上我对机器人视觉有了初步的了解，也为今后的学习和工作打下坚实的基础，同时这也是我所热爱的一件事，希望可以在 6 月的比赛中为上大创造辉煌。

4.2 相机标定实践

为建立相机成像几何模型，就需要相机的内参矩阵。获取方式即对相机进行标定，标定的方式有很多种，此处使用“张正友标定法”^[1]。标定方式并不困难，主要在于对 Matlab 标定工具的熟悉和对标定原理的大致了解（如：考虑图像畸变，以及拟合指数的问题）。

如图 12 所示，为对于海康 Hik-Vision 工业相机的标定，采用 Matlab Camera Calibrator 工具。通过所需距离范围内对于棋盘的拍摄，选择畸变参数和指数系数即可得到相机的各类内参矩阵。根据需求导出 yaml 等格式的数据文件即可导入程序使用。



(a) Matlab Camera Calibrator 标定工具

(b) 标定得到的内参矩阵

图 12 使用 Matlab 对海康工业相机的标定

4.3 运动目标移动预测

无论是实际应用还是本项目中，我们通常对目标进行跟踪和预测时，常对它的运动方式比较关心，线性、非线性往往采用不同的预测方式。因此本节中将先对识别到的目标的运动方式、参数进行确定；下一节中根据运动方式进行预测。

4.3.1 匀速高速非线性运动目标预测

击打目标中包括“前哨站”的旋转装甲板，如图 13 所示，其旋转策略为 $0.4 (\pm 0.005)$ round/s.

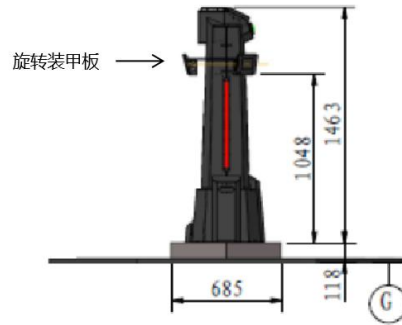


图 13 非线性匀速高速运动装甲板示意图

4.3.1.1 运动方式评估

首先通过识别目标中心前后帧的矢量计算，得到大量离散点，使用 Python 拟合近似曲线。由于旋转装甲板为本赛季新添加的元素，而对一些已有场地元素的预测有了一定的经验，故：将旋转装甲板的运动曲线和小陀螺（即：自旋机器人）以及“哨兵”（在轨道线性运动的机器人）的运动曲线进行对比，如图 14 所示，发现旋转装甲板的运动周期较小，且目标与中心的距离仅在 1 米左右，故无法看作线性运动。

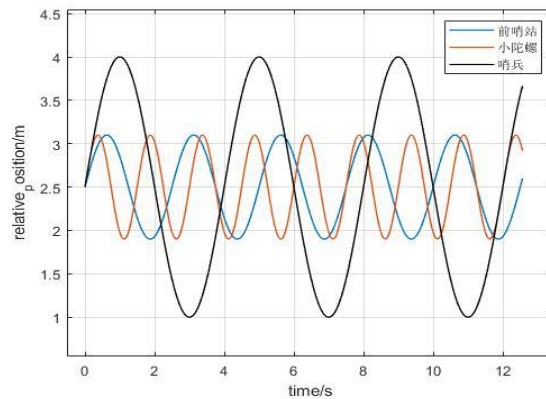


图 14: 哨兵、旋转装甲板和舵轮小陀螺运动曲线

由于实际旋转速度存在误差，为预测得到更准确的旋转速度，故考虑观测半个周期的数据拟合更准确的旋转速度。初期考虑实用 Fourier Transform，但实验结果因只能观测到一块装甲板而使频谱图并不能准确反映速度。因此，直接考虑时域上，两个最大值之间的时间差。峰值点的计算，会首先滤除和均值的标准差相差较大的点，已得到尽可能准确的周期数值，以确定旋转速度。

4.3.1.2 预测和击打策略

实验中，尝试通过跟踪目标的方式进行击打，但由于高速的旋转，目标出现在视野中的时间极短，考虑发弹延时，跟踪目标的方法会导致机器人云台来回摆动的情况，极大影响射击、预测效率。因此，由于旋转装甲板具有良好的机械限制，只需找出旋转的中心位置，当满足：

$$|x_{center} - x_{t_{communication} + t_{orbit} + t_{algorithm}}| < \varepsilon$$

即可自动发弹，上述公式中， x_{center} 为旋转中心的 X 坐标， x_t 为计算通信延时和算法延时后，目标旋转位置的 X 坐标，延时几乎可以精确测量。

4.3.2 变速非线性运动目标预测

击打目标中还包括“能量机关”的旋转装甲板，击打时，随机一块扇叶将被点亮，2s 内成功击打，会随机亮起下一块扇叶，直到全部扇叶点亮视为击打成功。如图 15 所示，其旋转策略为平面旋转，旋转速度近似满足方程： $spd = a * \sin(\omega * t) + b$ ，其中 spd 单位为 rad/s。

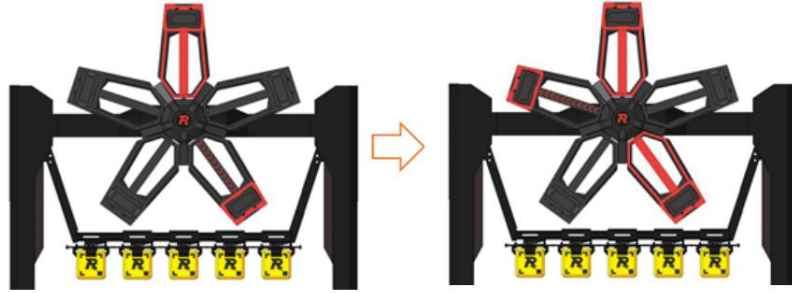


图 15 非线性变速运动装甲板示意图

由于旋转近似三角函数，参数较多，需要对旋转曲线进行拟合。曲线拟合尝试用最小二乘法作为工具，其基本思路为：令

$$f(x) = a_1 \varphi_1(x) + a_2 \varphi_2(x) + \cdots + a_m \varphi_m(x)$$

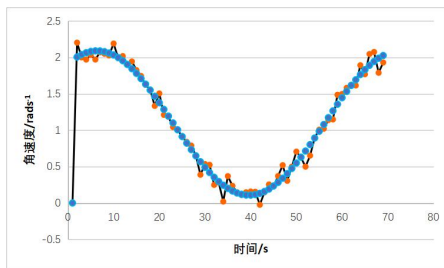
其中， $\varphi_k(x)$ 是事先选定的一组线性无关的函数， a_k 是待定系数，拟合准则是使 y_i 与 $f(x_i)$ 的距离的平方和最小，从而得到最小二乘拟合计算公式：

$$\min_x \frac{1}{2} \sum_i \rho_i \left(\|f_i(x_{i_1}, \dots, x_{i_k})\|^2 \right)$$

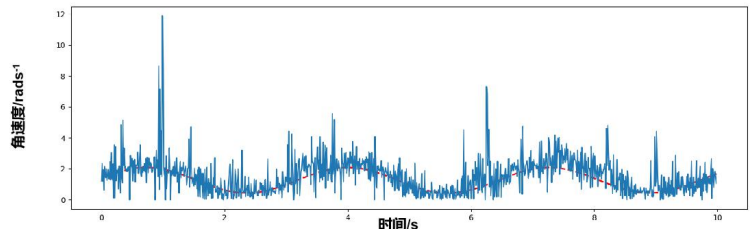
以下将对各个量进行介绍：

表达式 $\rho_i(\|f_i(x_{i_1}, x_{i_2}, \dots, x_{i_k})\|^2)$ 称为残差块 (Residual Block)， ρ_i 是损失函数 (Loss Function)， $f_i(\sim)$ 是代价函数 (Cost Function)，依赖于参数块 (Parameter Block)，即： $[x_{i_1}, x_{i_2}, \dots, x_{i_k}]$ 。

对于参数块而言，在大多数拟合、优化问题中，多个标量可能同时出现。例如，一个平移矢量的三个分量和定义摄像机姿态的四元数的四个分量。我们把这样三元组或四元组称为参数块。当然，Parameter Block 也可以只是一个参数，如这里的运动方程拟合就是如此，故无需过多考虑。



(a) 理论拟合效果



(b) 实际拟合效果

图 16 理论拟合效果

在“能量机关”运动方程拟合中，经过测试，残差块的各函数选择如下：

代价函数选择三角残差 (Trigonometric Residual)；损失函数选择柯西损失 (Cauchy Loss)。

经过一定数据量的测试，确定了理论的可行性，如图 16(a) 所示。投入项目使用时，选择使用算法优化库 Ceres Solver。目前对于能量机关的运动方程拟合，已达到可观的效果，如图 16(b) 所示。预测击打仅需对各类延时做出计算加入偏移量即可，在此不过多赘述。

预测部分核心算法部分代码见附录。

5 总结

智能机器人的四部分关键技术：本体机构、感知、决策和执行技术，使之将涵盖多个学科的研究，如：机械工程、人工智能、控制科学、计算机、材料等，应用领域涉及制造业、物流业、医疗、生活服务、航空航天、海洋、军工等领域。自 2020 年新冠疫情爆发以来，医疗机器人的开发逐步提上日程，近日在上海大学校园内看到的“医疗物资搬运机器人”、新闻中报道的“数字哨兵”都是最新的成果。因此，在智能机器人各项技术方面的突破和创新刻不容缓，国家发布的中国制造 2025 计划也旨在推进这些技术的发展。

在 Science 杂志发布的机器人十大挑战和方向中，智能机器人是其中一个重要方向，子类方向包括脑机接口、社会交互、医疗等。关键技术中，智能感知方面，需要设计完整的机器人视觉感知系统，将三维识别、SLAM 技术等加以应用；智能控制技术方面，柔顺控制、多机器人协调控制是当下热点，广泛应用基于深度学习、强化学习的控制技术。除此以外，还需要稳定有效的机械关节、能适应各种极端环境的制造材料、稳定持久的供电能源，这些则更多地由其他学科的专家关心。前两个方面的技术是我们计算机专业学生主要研究的领域，如前所述，虽说二者是我们的研究重点，但其他的相关技术也应有一定的涉猎，才能帮助我们更好地协同开发高性能的智能机器人。

在 5G 到来的时代背景下，在通讯速率提高，宽带、时延减少的云空间下，未来机器人的应用将面向网络协同化制造，基于柔性的自动化生产线进行小批量、多品种和个性化的制造。其需要具备数字化、网络化、智能化的特征。网络化协同制造可以满足个性化的生产与服务模式，以及复杂多变的作业环境与任务，是未来工业向智能化生产发展的趋势。智能机器人将在其中扮演重要作用。

最后，我认为，对前沿科技的发展而言，就我个人经验来说，仅从高校阶段才进行培养或许是有些滞后的，我国在初中、高中阶段的各类科技竞赛项目，往往多由家长、机构代劳，很多成果显然不是那个年龄段能够完成的，而国外从初中、甚至小学，就有体系化的机器人比赛，鼓励青少年动手，去体会自己和他人的差距从而发现问题，学习他人思路，激发自己的思考。但就个人体会而言，一味地照搬国外显然会出现大批量机构涌现、买奖等情况，因此，或许在体系上可以做出些调整，从教育部近期提出的双减政策等，也是在鼓励这方面的动手。总而言之，问题是在不断发现中改善的，技术是在不断积淀中突破的，相信我国的整个科技体系会逐渐完善，终有一天会成为科技强国。

6 参考文献

- [1] 刘军民.人工智能发展现状和趋势报告会在京举行高文院士:人工智能要与实体经济深度融合[J].环境与生活, 2019(5):102-103
- [2] 何桂梅.智能机器人及其发展趋势展望[J].信息记录材料,2021,22(12):5-6.
- [3] 王敏,黄心汉. 基于视觉与超声技术机器人自动识别抓取系统[J]. 华中科技大学学报(自然科学版),2001(01):73-75
- [4] 张华,夏菠. 基于双目视觉的机器人自定位方法研究[J]. 西南科技大学学报(自然科学版),2015,43(S1):104-108.
- [5] Haikuo S, Kaihua Z, Afson N, A noncontact positioning measuring system based on distributed wireless networks[J]. Peer-to-Peer Networking and Applications, 2017,10(3):823-832
- [6] 韩博. 基于双目立体视觉的工件识别与定位关键技术研究[D].哈尔滨工业大学,2018.
- [7] 李泽辰. 基于双目视觉的工业机器人目标识别与定位系统研究[J]. 2021,25(1):11-12
- [8] 陈英. 机器视觉技术的发展现状与应用动态研究[J]. 无线互联科技, 2018.15(19):147-148
- [9] 王飞. 机器视觉技术发展及其工业应用[J]. 电子技术与软件工程,2018(16):246-246
- [10] D. S. Bolme, J. R. Beveridge, B. A. Draper and Y. M. Lui, "Visual object tracking using adaptive correlation filters," 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 2544-2550.
- [11] Zhengyou Zhang, "Flexible camera calibration by viewing a plane from unknown orientations," Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, pp. 666-673 vol.1.
- [12] 李长文, 初磊, 毛俊超. 可预测弹道目标的鱼雷射击参数计算方法[J]. 水下无人系统学报, 2017, 25(3): 267-271.

7 附录（预测移动目标核心算法部分代码）

以下代码为上海大学 RoboMaster 战队项目中本人开发部分代码，项目暂未开源
故以下代码仅作为本课程研讨内容“智能机器人技术进展”的拓展实践，请勿开源。

```
// 预测非线性变速运动物体（能量机关）算法流程
// @author Yunzhe Li 李昀哲

SendPacket predictor::rune::RunePredictor::Run(const PowerRune &power_rune, AimModes aim_mode) {
    rune_ = power_rune;
    LOG(INFO) << "Rune predictor mode: big.";
    state_.UpdatePalstance(rune_, fitting_data_); // 更新旋转角速度
    state_.UpdateAngle(rune_.RtpVec());          // 更新当前帧角度
    PredictAngle(aim_mode);                       // 预测角度
    PredictPoint();                               // 预测击打点
    output_data_.Update(debug_, rune_, predicted_angle_, predicted_point_, fixed_point_);
    state_.CheckMode();
    // 返回预测值给控制板，由控制板控制云台
    return {output_data_.yaw, output_data_.pitch, output_data_.delay,
            static_cast<int>(output_data_.yaw + output_data_.pitch + output_data_.delay)};
}

// 预测非线性匀速高速运动物体（前哨站旋转装甲板）算法流程
// @author Yunzhe Li 李昀哲

SendToOutpostPredictor OutpostDetector::Run(const Battlefield& battlefield)
{
    SendToOutpostPredictor send_to_outpost_predictor;
    detected_armors_in_this_frame_.clear();
    auto facility = battlefield.Facilities();
    if (facility[color_][Facility::kOutpost] == nullptr)
    {
        DLOG(INFO) << "No outpost armor founded";
        return send_to_outpost_predictor;
    }
    detected_armors_in_this_frame_ = facility[color_][Facility::kOutpost]->BottomArmors(); // 维护识别到的装甲板

    if(clockwise_ <= 7 && clockwise_ >= -7 && !is_checked_clockwise) // 判断旋转方向
        IsClockwise();
    else
        DecideComingGoing(); // 决策“将要来”和“将要走”的装甲板
    FindBiggestArmor();       // 寻找最大装甲板作为击打中心
    // 发送计算数据
    send_to_outpost_predictor.UpdateInfo(going_center_point_, outpost_center_, coming_center_point_,
                                         clockwise_, center_distance_, battlefield.BulletSpeed(), shoot_point_);
    return send_to_outpost_predictor;
}
```



```

// 拟合方程时，基于 Ceres Solver 的三角残差块定义。下一页介绍详细算法
// @author Yunzhe Li 李昀哲
/// @brief Trigonometric residual (cost) function package for Ceres solver.
struct TrigonometricResidual {
/**
 * @param _x Independent (input) variable.
 * @param _y Dependent (output) variable.
 * @param _rotational_direction 1 is clockwise, -1 is counterclockwise.
 */
  TrigonometricResidual(double _x, double _y, int _rotational_direction) :x(_x), y(_y), rotational_direction(_rotational_direction) {}
/**
 * @brief Calculate trigonometric (cost) residual.
 * @tparam T Reserved for Ceres solver.
 * @param [in] a Amplitude.
 * @param [in] omega Palstance.
 * @param [in] phi Initial phase.
 * @param [out] residual Calculation result.
 * @return Always true, reserved for Ceres solver.
 */
  template<typename T>
  bool operator()(const T *const a, const T *const omega, const T *const phi, T *residual) const {
    // residual[0] = y - (-1.0) * rotational_direction * (a[0] * sin(omega[0] * x + phi[0]) + 2.090 - a[0])
    residual[0] = y + (a[0] * sin(omega[0] * x + phi[0]) + 2.090 - a[0]) * (double) rotational_direction;
    return true;
  }
  double x; ///< Independent variable.
  double y; ///< Dependent variable.
  int rotational_direction; ///< 1 is clockwise, -1 is counterclockwise.
};

```

```

// 拟合方程算法
// @ author Yunzhe Li 李昀哲
void predictor::rune::FittingData::Fit(bool debug, RotationalSpeed &rotational_speed) {
    if (!outdated)
        return;
    if (!ready) {
        // Preparing, wait for some time.
        if (palstance.size() >= kPreparePalstanceDataNum) {
            ready = true;
            palstance.clear();
        }
    } else {
        // Collect data and fit.
        if (palstance.size() >= kCollectPalstanceDataNum) { // 观测一定量的数据
            for (double &speed: palstance)
                speed = speed / 180 * CV_PI;
            ceres::Problem problem;
            for (int i = 0; i < kResidualBlockNum; ++i)
                problem.AddResidualBlock(
                    new ceres::AutoDiffCostFunction<TrigonometricResidual, 1, 1, 1, 1>( // 使用上一页的残差块
                        new TrigonometricResidual(
                            time[i], palstance[i],
                            rotational_speed.rotational_direction)),
                    new ceres::CauchyLoss(0.5),
                    &rotational_speed.a,
                    &rotational_speed.w,
                    &rotational_speed.p);
            ceres::Solver::Options options;
            options.max_num_iterations = 300;
            options.linear_solver_type = ceres::DENSE_QR;
            options.minimizer_progress_to_stdout = true;
            if (debug) {
                ceres::Solver::Summary summary;
                ceres::Solve(options, &problem, &summary);
                DLOG(INFO) << summary.BriefReport();
                DLOG(INFO) << "Fitting: a = " << rotational_speed.a
                    << ", w = " << rotational_speed.w
                    << ", phi = " << rotational_speed.p;
            }
            ready = false;
            outdated = false;
        }
    }
}

```