



单位代码 10006

学 号 15031108

分 类 号 V324

密 级 公开

北京航空航天大学  
BEIHANG UNIVERSITY

# 毕业设计(论文)

## 基于图像的多旋翼飞行器追踪打击控制

院（系）名称	自动化科学与电气工程学院
专 业 名 称	自动化
学 生 姓 名	哈泽辰
学院指导教师	全权
工业部门导师	李鑫

2019 年 5 月

I、毕业设计（论文）题目：

基于图像的多旋翼飞行器追踪打击控制

II、毕业设计使用的原始资料（数据）及设计技术要求

原始资料： Tracking a Moving Ground Object using Quadcopter UAV in a Presence of Noise.

技术要求： 设计基于视觉的追踪打击控制算法，使多旋翼飞行器可以最终打击到目标，并且在打击过程中目标在相机视域内。

III、毕业设计（论文）工作内容

设计基于图像的多旋翼近距离追踪打击控制算法；

比较多种基于视觉的目标追踪算法的速度和精度，选用 KCF 方法；

利用 Simulink 虚拟现实工具箱对基于图像的追踪打击控制系统进行视景仿真，证明算法可行性；并通过控制变量法多维比较后，得到目标不同运动轨迹、运动速度对控制效果的影响。

IV、主要参考资料

Chaumette F, Hutchinson S. Visual servo control. I. basic approaches[J]. Robotics & Automation Magazine, 2006, 13(4): 82-90.

全权. 多旋翼飞行器设计与控制[M]. 电子工业出版社: 北京, 2018:227.

Y. Wu, J. Lim, M. Yang, Online object tracking: a benchmark[C], Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2411-2418.

J. F. Henriques, R. Caseiro, P. Martins and J. Batista. High-Speed Tracking with Kernelized Correlation Filters[C], IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 3, pp. 583-596, 1 March 2015.

自动化科学与电气工程学院自动化专业类 150321 班

学生 哈泽辰

毕业设计（论文）时间： 2018 年 12 月 16 日至 2019 年 6 月 2 日

答辩时间： 2019 年 05 月 31 日

成 绩：\_\_\_\_\_

指导教师：\_\_\_\_\_

兼职教师或答疑教师（并指出所负责部分）：

\_\_\_\_\_  
\_\_\_\_\_

\_\_\_\_\_系（教研室） 主任（签字）：\_\_\_\_\_

## 关于学位论文的独创性声明

本人郑重声明：所呈交的论文是本人在指导教师指导下独立进行研究工作所取得的成果，论文中有关资料和数据是实事求是的。尽我所知，除文中已经加以标注和致谢外，本文不包含其他人已经发表或撰写过的研究成果，也不包含本人或他人为获得北京航空航天大学或其它教育机构的学位或学历证书而使用过的材料。与我一同工作的同志对研究所做的任何贡献均已在论文中做出了明确的说明。

若有不实之处，本人愿意承担相关法律责任。

学位论文作者签名：\_\_\_\_\_

日期：      年      月      日

## 学位论文使用授权书

本人完全同意北京航空航天大学有权使用本学位论文（包括但不限于其印刷版和电子版），使用方式包括但不限于：保留学位论文，按规定向国家有关部门（机构）送交学位论文，以学术交流为目的赠送和交换学位论文，允许学位论文被查阅、借阅和复印，将学位论文的全部或部分内容编入有关数据库进行检索，采用影印、缩印或其他复制手段保存学位论文。

本论文不涉及任何保密内容。若有涉密内容，本人愿意承担相关法律责任。

学位论文作者签名：\_\_\_\_\_

日期：      年      月      日

学院指导教师签名：\_\_\_\_\_

日期：      年      月      日



## 基于图像的多旋翼飞行器追踪打击控制

学 生：哈泽辰

指导教师：全权

校外导师：李鑫

### 摘要

多旋翼飞行器因其体积小、重量轻、造价低，具有好的机动性能和操纵性能而才成为一种被广泛使用的现代工具，在搜索、航拍、监视等领域表现出出色的特征优势，更被应用到军事行动等特殊领域。现代战争中已经开始将多旋翼投入到狭窄空间和街头楼巷的作战任务中，但是通常需要经验丰富的操纵手控制其对目标实现精确打击，这种方法有高风险性、低易用性等弊端，所以本文希望提出一种可靠的基于图像的伺服控制方法，使多旋翼能够自主飞行精确打击目标。

本文主要工作如下：

首先建立多旋翼通道模型和视觉伺服模型，作为后续算法设计的数学基础；其次通过实验综合比较多种成熟的在线追踪算法的性能，并选用表现最出色的 KCF 完成基于视觉的目标追踪环节；第三是核心章节——设计基于视觉的追踪打击控制算法，通过解耦雅阁比矩阵将视觉伺服模型简化成纵向和横侧向通道模型，分别设计两个通道中的控制算法并证明算法的收敛性；最后基于设计环节，在 MATLAB 上利用虚拟现实工具箱对视觉伺服算法进行视景仿真验证，并分析仿真结果。

在实验环节，本文通过调整控制参数，考虑不同的目标轨迹和运动速度，进行了多组视景仿真。仿真结果分析表明，本文的设计方法可以实现多旋翼飞行器基于图像的精确打击控制，并探讨了目标不同运动轨迹、不同运动速度对控制效果的影响。

**关键词：**多旋翼飞行器，视觉伺服控制，基于视觉的目标追踪，精确打击



## Image Based Attacking Control of Multicopters

Author : Ha Zechen

Tutor : Quan Quan

Co-Tutor : Li Li

### Abstract

The multicopters have become widely used modern tools, due to their small size, light weight, low cost, good maneuverability, and stability, showing excellent features in the fields of search, aerial photography, surveillance, etc. The multicopters are also applied to special areas such as military operations. In modern warfare, multicopters have been put into the battle missions of narrow spaces and street buildings, but the tasks usually require experienced remote pilots to make precision attacks, which has many disadvantages such as high risk and low ease of use, etc. This dissertation aims to propose a reliable image-based servo control algorithm so that multicopters are able to autonomously make precision attacks.

The main work of this dissertation is shown as follows:

Firstly, the multicopter channel model and the visual servo model are established for the following algorithm design. Secondly, comparing the performance of various mature online visual tracking algorithms comprehensively, the Kernelized Correlation Filter (KCF) is selected. The third part, a key part of this dissertation, is about image-based servo control algorithm. The visual servo model is simplified as a longitudinal and a horizontal channel model with the interaction matrix decoupled. Then, the control algorithms for the two channels are designed, and the convergence is further proved. Finally, based on the design link, the dissertation uses virtual reality toolbox on MATLAB to verify the visual servo algorithm by visual simulation and the results are analyzed.

In the simulation process, by adjusting the control parameters, multiple groups of visual simulations are performed, where different target trajectories and velocities are considered. The simulation results show that the design method can realize the image-based precision attacks based on multicopters. The control performance of different target trajectories and velocities are discussed.



Keywords: multicopters, visual servo control, visual tracking, precision attack



## 目录

1 绪论 .....	1
1.1 研究背景 .....	1
1.2 研究意义 .....	2
1.3 国内外研究现状 .....	3
1.3.1 基于视觉的目标追踪算法 .....	3
1.3.2 视觉伺服概论 .....	3
1.3.3 带有视场角约束的制导律 .....	5
1.3.4 小结 .....	6
1.4 研究目标与研究内容 .....	6
1.4.1 研究目标 .....	6
1.4.2 研究内容 .....	6
1.4 文章组织结构 .....	9
2 多旋翼视觉伺服模型 .....	10
2.1 多旋翼飞行控制刚体模型 .....	10
2.1.1 假设 .....	10
2.1.2 刚体运动学模型 .....	11
2.1.3 刚体动力学模型 .....	11
2.1.4 多旋翼飞行控制刚体模型 .....	13
2.2 视觉成像模型 .....	13
2.2.1 坐标系定义 .....	14
2.2.2 针孔模型 .....	14
2.3 视觉伺服模型 .....	16
2.3.1 视觉伺服公式 .....	16
2.3.2 雅阁比矩阵 .....	17
2.4 本章小结 .....	18
3 基于视觉的目标追踪算法分析 .....	19
3.1 基于视觉的目标追踪算法比较与选取 .....	19





3.2 KCF 算法原理和优势 .....	21
3.2.1 算法原理 .....	21
3.2.2 算法优势 .....	23
3.2 本章小结 .....	24
4 基于视觉的追踪打击控制 .....	25
4.1 多旋翼简化的线性模型 .....	25
4.2 视觉伺服模型解耦 .....	27
4.3 多旋翼的视觉伺服模型 .....	28
4.4 纵向通道控制器设计 .....	30
4.5 横侧向通道控制器设计 .....	32
4.6 仿真测试和性能分析 .....	33
4.7 本章小结 .....	37
5 仿真平台和综合实验 .....	39
5.1 仿真环境搭建 .....	39
5.2 综合控制系统框架 .....	41
5.2.1 子模块构成 .....	41
5.2.2 仿真框架 .....	44
5.3 仿真测试和性能分析 .....	46
5.4 本章小结 .....	52
总结和展望 .....	53
总结 .....	53
展望 .....	53
致谢 .....	54
参考文献 .....	55

## 1 绪论

### 1.1 研究背景

多旋翼无人机正快速成为一种适用范围广泛的现代工具，它在搜索、远程遥感、救援、航拍、监视等领域已经表现出出色的特征优势<sup>[1]</sup>，更被应用于军事行动等特殊领域。小型多旋翼无人机由于体型小、质量轻、可垂直起降、安全性好、机动灵敏、结构简单等特点，在现代战争中开始被应用于狭窄空间和街头楼巷的作战任务中，到现在更是已经推广到更多场景，频频出现在叙利亚、利比亚战场上<sup>[2]</sup>。

目前主流采用遥控多旋翼，需要经验丰富的操纵手操控多旋翼无人机实现对目标精确打击。然而这种方式存在诸多不足，首先完成打击需要经验丰富的操纵手，降低了多旋翼作战的易用性和广泛应用潜力，大大增加操纵手的培训成本；并且几乎只能对侦查过的固定目标进行打击，例如对入侵的多旋翼无人机这种灵活目标这种方法就难以奏效；在复杂地形中需要保持多旋翼处于操纵手视域之内，操纵手的人身安全威胁增加，因此一个合理的需求是制定一种可靠的方法控制飞行器自主飞行<sup>[3]</sup>。

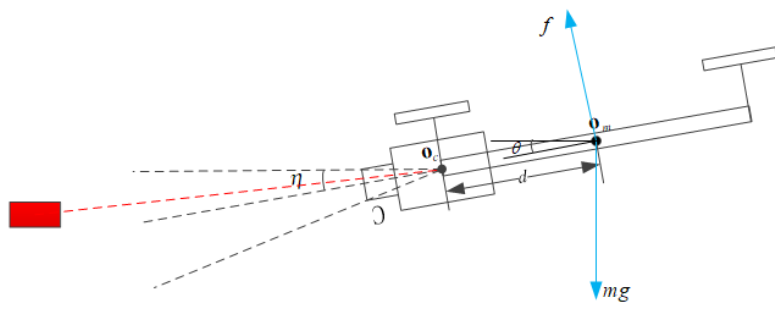


图 1.1 俯仰平面目标打击示意图

本文考虑使用基于视觉的目标追踪打击控制满足这种需求，俯仰平面目标打击示意图如图 1.1 所示，红色块为目标，小型多旋翼无人机的机架上固连着摄像头获取图像。在机载处理器处理信息的情况下，通过摄像头获取图片，使用基于视觉的目标追踪算法捕获目标在图像中的位置坐标，利用视觉伺服算法控制飞行器运动使得目标始终在摄像头视域之内并向中心收敛，完成一个控制闭环。这种新型作战方式脱离了操纵手的限制，减少大规模培训操纵手所需的经济成本和时间成本，对于普通士兵更加易用，对移动的



灵活目标更具威胁，应用场景更加广泛，并且保护了操纵人员的安全。

## 1.2 研究意义

相较于传统的单兵打击方式，利用多旋翼飞行器这种现代工具进行打击无疑是更加优秀的，多旋翼飞行器打击是一种远程打击，所以可以有效降低士兵所受到的危险，并且适用于更加广泛的地形环境，可以打击士兵所不能及的目标区域，多旋翼飞行器还在一定程度上具有视距外打击能力，增强了打击的隐蔽性和突然性。

使用操纵手控制多旋翼无人飞行器打击是目前战场主流作战方式，大多需要操控手操控到达指定地点完成任务，操纵手在靠近目标的地方是非常危险的，采用视觉伺服进行近距离跟踪控制打击脱离了对于经验丰富的操纵手的需求，保护了操纵手的安全，同时降低了培训操纵手的经济和时间成本，面向普通士兵使用更加友好，提高这种新型作战方式的易用性和推广性，降低军队列装难度，并且该方法对于灵活移动物体具有有效打击能力。

传统的多旋翼飞行器一般采用全球定位系统（GPS）和惯性测量单元（IMU）提供飞机的位姿信息。但是GPS信号穿透性弱，定位精度受信号强度影响，在山区、森林、洞穴以及建筑物繁多的城市等复杂地形和室内都不具有很高的可靠性，尽管INS/GPS组合导航方法通过互补二者特性提升了准确度和实时效率，但其相对于机动性小目标的定位精度依然难以满足需求，因此借助视觉传感器获取图片感知局部环境是一种有效的代替方案<sup>[4]</sup>。

在目前现有传统控制多旋翼方法中，大多只适用于打击侦查过的固定目标，或者不灵活的移动目标，对于例如入侵己方领空的多旋翼无人机这种灵活机动性极高的目标不具有打击能力，而使用视觉伺服的方法可以打击灵活目标，增加了打击目标范围。

在跟踪目标并且打击和设计制导律方面，对于导弹的研究比较充分，然而多旋翼飞行器具有导弹所不具备的优势，小型多旋翼无人机体型小、造价低、易操作、易维护、易携带、隐蔽性高，适用于严峻环境，所以适用于在狭窄复杂的环境，灵活机动的任务中，并且容易单兵携带，所以研究小型多旋翼的制导打击是具有现实意义的。



### 1.3 国内外研究现状

多旋翼最初多使用 GPS (Global Positioning System) 进行控制,但是考虑到 GPS 信号被环境影响很大,研究人员考虑使用视觉信息来获取多旋翼的位置信息。现在已经有了一些多旋翼的视觉控制应用。事实上,因为在很多多旋翼平台上已经有视觉系统,方便地为闭环控制系统提供相对位置信息,一定程度上解决了系统在 GPS 信号不可靠区域的应用。

例如,文献<sup>[5]</sup>利用基于图像的视觉伺服算法实现直升机自主悬停在一个固定标定上空,相似的应用还有<sup>[6]</sup>中通过实时的视觉伺服算法实现直升机在固定标定上降落,文献<sup>[7]</sup>中使用视觉伺服算法延道路飞行。这些应用的目标都是固定目标或者缓慢目标,本文希望利用视觉伺服算法追踪一些移动方式不同的目标。

#### 1.3.1 基于视觉的目标追踪算法

在线视觉追踪旨在实景中追踪目标位置。广义来讲,一种视觉追踪算法包含两个组成部分:一种运动模型,描述物体随时间的状态并预测其可能的状态;一个观测模型,描述目标的外表特征并在每一帧进行预测。一些学者已经证实了,观测模型比运动模型更重要。视觉追踪是计算机视觉的诸多应用中一种基础的任务类型,并且已经被研究了数十年。尽管研究者们已经提出了很多实现方法,稳定有效的视觉追踪任然是一个艰巨的挑战。视觉追踪的困难在于目标运动、特征变化、目标结构变化和摄像头运动。目前的算法正逐步解决这些问题。

传统的追踪算法专注于从在线学习算法或者指定的特征中研究稳定的表征模型。这些算法包括 IVT<sup>[8]</sup>, MIL<sup>[9]</sup>, APGL<sup>[10]</sup>, KCF<sup>[11]</sup>, ASLAS<sup>[12]</sup>。

在近几年,深度学习因为成功通过多层非线性变换实现了特征提取,在计算机视觉领域取得了重要进步。有一些使用深度学习的追踪算法,比如 MDNet<sup>[13]</sup>, STCT<sup>[14]</sup>, SiameFC<sup>[15]</sup>已经证明了他们提升追踪效果的潜力。

本文将比较一些成熟的、可以在线使用的视觉追踪算法在视景仿真中的效果,并选用一种已经成熟的追踪算法作为完成视觉伺服系统的必要模块。

#### 1.3.2 视觉伺服概论

视觉伺服指的是利用计算机视觉数据对机器人的运动进行控制,依赖于控制理论、

图像处理和计算机视觉等技术。图像数据由摄像头获取，摄像头可以直接固连在多旋翼上，这样摄像头的位姿和多旋翼位姿耦合；也可以将摄像头固定在一个工作空间周围，这样可以在惯性系下观察获取多旋翼位姿。根据这两种不同的方式，我们将前一种称为于图像的视觉伺服（Image-Based Visual Servo, IBVS），后一种称为基于位置的视觉伺服（Position-Based Visual Servo, PBVS）

所有基于视觉伺服的方案，目的都是减小图像误差  $\mathbf{e}(t)$ 。误差  $\mathbf{e}(t)$  表示为

$$\mathbf{e}(t) = \mathbf{s}(\mathbf{m}(t), \mathbf{a}) - \mathbf{s}^* \quad (1.1)$$

其中，矢量  $\mathbf{m}(t)$  是图像的测量值； $\mathbf{s}(\mathbf{m}(t), \mathbf{a})$  是视觉特征点矢量； $\mathbf{s}^*$  为期望特征值； $\mathbf{a}$  为摄像头固有参数。

在 PBVS 中， $\mathbf{s}$  定义为一系列三维参数，必须从图像中估计；而在 IBVS 中， $\mathbf{s}$  图像中直接可用的一系列特征点，即跟踪点在图像平面中的坐标，可以在图像中直接读出<sup>[16]</sup>。

在基于位置的视觉伺服中，视觉传感器被利用视觉算法通过获取深度信息，构建三维空间的方法，实现位置估计，因而被认为是一种三维传感器。PBVS 方法的缺点在于鲁棒性差，在图像中的测量结果微小错误会引发严重的位姿估计偏差。多旋翼飞行器 PBVS 的基本结构如图 1.2 所示。

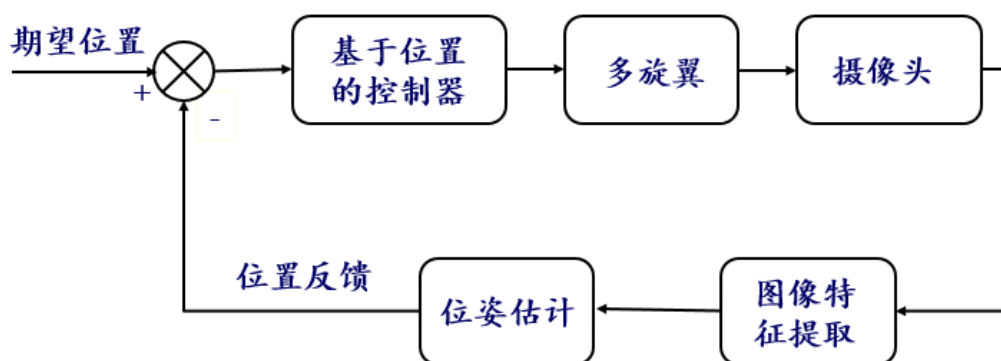


图 1.2 PBVS 基本结构

在基于图像的视觉伺服中，视觉传感器直接从图像空间中提取特征，被认为是一种二维传感器，相较于 PBVS，IBVS 被认为是具有显著的鲁棒性的，对于图像误差。IBVS 的缺点在于，当移位过大时摄像头会达到局部最小值或跳过交互矩阵的奇点，从而不可预知摄像头的运动。多旋翼飞行器的 IBVS 基本结构如图 1.3 所示<sup>[17]</sup>。

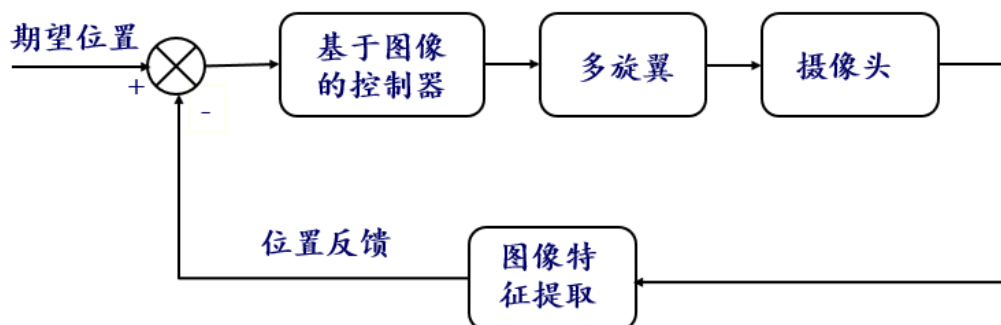


图 1.3 IBVS 基本结构

PBVS 和 IBVS 的主要优缺点总结如下表 1.1

表 1.1 PBVS 和 IBVS 优缺点总结

特点	PBVS	IBVS
优点	可以直观地在直角坐标空间定义目标的运动，符合现有机器人的工作方式	无需三维空间定位、对摄像机和机器人标定不敏感；计算量较小
缺点	控制精度很大程度上依赖于位姿估计精度，而位姿估计精度依赖于摄像机和机器人的标定精度等；此外计算较大	伺服控制器复杂且缺乏适应性；需要额外的传感器获取深度信息；移位过大会导致不可预知摄像头运动

本文使用基于图像的视觉伺服，在图像中获取目标的图像坐标系坐标，利用交互矩阵建立摄像头的速度、角速度和图像目标偏差的矩阵运算关系，从而完成接下来的控制闭环。

### 1.3.3 带有视场角约束的制导律

多旋翼基于视觉伺服的制导可以参考有视角约束的制导问题，因为多旋翼打击过程中需要保持目标在摄像头视域之内。可以通过研究导弹导引头带视角约束的末制导律设计，得到四旋翼飞行器带有视角约束制导律设计问题的解决思路。

文献<sup>[18]</sup>首次对导引头视场约束进行了讨论，该问题使用最优加权函数解非线性最优控制问题。文献<sup>[19]</sup>仅考虑导引头视场约束，设计了复合制导律，第一段被称为原制导律，第二段为恒定目标视角制导律。在此基础上，文献<sup>[20]</sup>设计了带有状态变量不等式约束的复合制导律，第一段制导律目标是导引头达到最大视角，第二段制导律目标是导引头视角始终保持在边界值内，第三段制导律目标是保证导弹以期望攻击角成功



拦截目标。

### 1.3.4 小结

多旋翼飞行器平台可以容易地获取摄像头的图像信息，经过视觉伺服算法建立摄像头速度、角速度与图像坐标系中目标偏差的矩阵运算关系，因为摄像头固连在机架上，摄像头的运动状态可以近似的认为就是飞行器的运动状态。为了目标向图像中心收敛，可以考虑参考导弹的带视角约束的制导问题解决方案，据此获得期望的飞行器控制向量，可以完成控制闭环，基于图像的视觉伺服。

## 1.4 研究目标与研究内容

本节将列出本文的研究目标与研究内容，下面将分别描述。

### 1.4.1 研究目标

研究目标是实现多旋翼飞行器基于视觉伺服的近距离攻击控制，设计制导律并且完成整个过程的仿真。具体可以分为以下几个目标：

- 1、建立多旋翼飞行控制刚体模型，多旋翼通道模型，视觉成像和视觉伺服模型。
- 2、选取基于视觉的目标追踪算法，根据数学模型分别设计俯仰平面和水平面上视觉伺服实验公式
- 3、在 MATLAB 上设计控制器并设计示意仿真实验，验证公式的正确性并比较偏差，分析示意实验中各参数的作用和影响，指导之后实验中的参数选取。
- 4、基于推导的公式需求，修改实验室搭建的 MATLAB/Simulink 平台多旋翼模型输入向量和一系列内部模块，使用 VRML 搭建仿真环境，并用结合 VR 工具库获取图像，完成 MATLAB/Simulink 上的三维视觉伺服仿真。
- 5、评价方法完成情况和效率。

### 1.4.2 研究内容

#### 1.4.2.1 研究内容简介

本文描述的应用场景是，具有固连在其机架摄像头地多旋翼无人飞行器利用视觉伺服算法追踪以不同方式运动的目标。

研究流程：多旋翼飞行器平台可以容易地获得图像信息，利用已有的基于视觉的目标追踪算法追踪目标在图像坐标系中的坐标，将图像坐标偏差输入到视觉伺服控制器中，控制器输出多旋翼飞行器的控制向量，改变多旋翼位姿，因为摄像头与飞行器固连，即改变了摄像头视角和目标的图像坐标位置，从而完成反馈，形成控制闭环，令目标向图像中心收敛并追踪目标移动。控制整体流程框架如图 1.4。

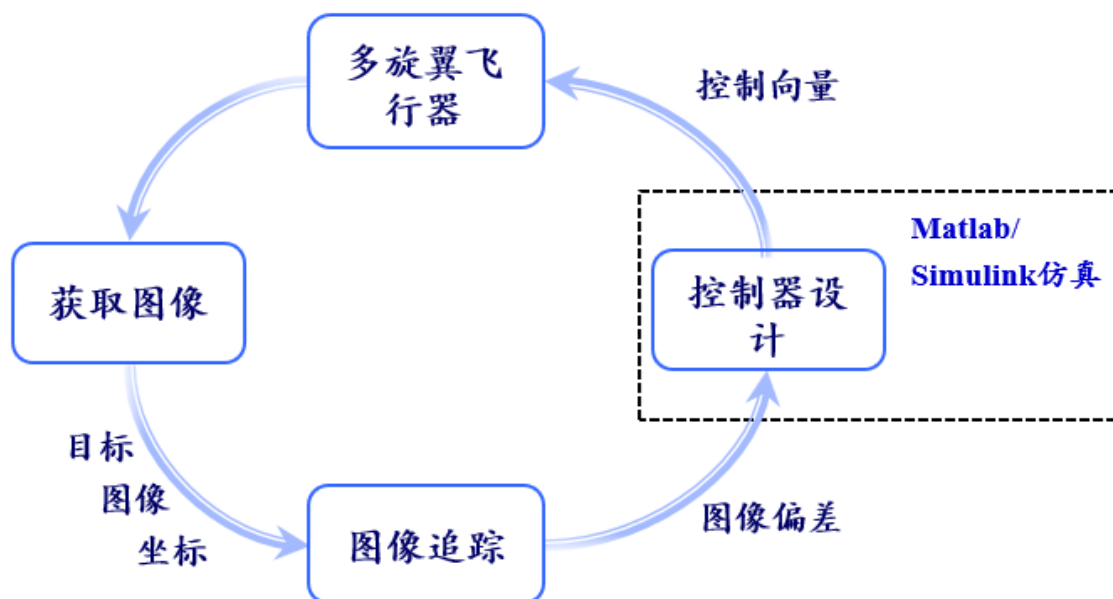


图 1.4 控制整体框架

本文重点在于控制算法，即视觉伺服部分的设计、编写、仿真和效果评价，因为目前视觉跟踪算法已经很多并且很成熟，所以不作为本文研究的重点，只需要在应用时选取已有基于视觉的目标追踪算法。

#### 1.4.2.2 研究内容解析

##### 1、建立数学模型

(1) 研究多旋翼飞行器运动模型和通道模型，对非线性模型新型适当线性化简化和线性化处理，最终得到多旋翼的水平通道模型、高度通道模型和姿态模型。

(2) 研究视觉成像模型，通过针孔模型确定物理图像坐标系和世界坐标系转换关系。

(3) 研究视觉伺服模型，通过摄像机模型推导图像坐标系坐标和相机坐标系坐标转换关系，得到雅阁比矩阵，并根据水平竖直分解后的运动公式，得到水平竖直化简后的





雅阁比矩阵。

## 2、设计视觉并推导伺服控制公式

(1) 选取 KCF 基于视觉的目标追踪算法，了解算法原理并实现算法。

(2) 在俯仰平面上，根据已建立的多旋翼通道模型和视觉伺服模型，推导设置参数，使目标在图像中可以向中心收敛。

(3) 在水平面上，根据已建立的多旋翼通道模型和视觉伺服模型，推导设置参数，使目标在图像中可以向中心收敛。

## 3、MATLAB 示意仿真和结果分析

(1) 根据俯仰平面和水平面设计的视觉伺服公式，在 MATLAB 上编写控制器，进行示意仿真。

(2) 分析示意仿真结果，分析参数在公式中的意义和对结果的影响，对之后的仿真参数整定起指导作用。

## 4、搭建 MATLAB/Simulink 平台模型

(1) 基于实验室搭建的多旋翼 Simulink 模型和多旋翼运动学公式，根据实验中输入控制向量需求，修改多旋翼模型。

(2) 使用 VRML 搭建 3D 仿真环境，根据需求模拟真实的环境，设置目标运动轨迹。

(3) 使用 MATLAB/Simulink 中的 VR 工具库获得 VR 显示图像，并进行必要的图像处理，使用图像跟踪算法得到每帧目标点图像中的坐标位置。

(4) 搭建视觉伺服算法模块作为控制器，并将返回的控制向量反馈给多旋翼完成控制闭环。

## 5、MATLAB 仿真

(1) 在 MATLAB / Simulink 平台上按照制导律编写控制器，输入图像坐标、已知量和多旋翼飞行控制参数，输出飞行器控制向量。

(2) 在 MATLAB 平台上完成仿真全过程，改变目标的运动方式，检验在目标不同运动方式和速度下的视觉伺服算法。

(3) 绘制每次仿真中目标和多旋翼飞行器的运动轨迹，根据飞行器轨迹计算控制评价指标，比较对应的目标和多旋翼参数，得到一些结论对算法有一些定性认知。



## 1.4 文章组织结构

本文分章节详细介绍了多旋翼飞行器利用视觉伺服算法追踪移动目标整个系统的设计和仿真过程。其中第 1 章主要介绍了研究背景和研究内容，并说明了国内外对基于视觉的目标追踪算法、视觉伺服、带视觉约束的控制律设计的研究现状。第 2 章在数学基础上对本研究课题进行支撑，介绍了多旋翼飞行控制刚体模型、视觉成像模型和视觉伺服模型，并明确了每个模型的输入输出关系。第 3 章主要介绍了视觉跟踪算法，当第一帧的目标图像坐标系位置标定出时，可以通过本算法持续获得目标的坐标信息。第 4 章首先简化多旋翼刚体模型和视觉伺服模型到横侧向通道和纵向通道，建立多旋翼通道模型与视觉伺服通道模型的关系，然后分别在两个通道中设计基于视觉的追踪打击控制算法，最后通过示意仿真比较全面地了解了算法中的参数影响。在本章中，将多旋翼在三维空间中的运动拆分成俯仰平面和水平面的运动分别进行控制设计，以简化模型和运算，然后进行平面中的示意仿真，定性地得到每个参数对控制的影响程度和参数的合理范围，为之后视景仿真的参数整定打下基础。第 5 章基于第 2，3，4 章的设计过程进行 VR 场景中的仿真实验，首先是搭建虚拟场景，然后在 Simulink 中编写控制器，搭建控制闭环，改变目标的运动方式验证多旋翼视觉伺服算法，最后对仿真过程进行评价。最后纵观全文，对本文做出总结，对不足之处做出以后的期望和改进方向探讨。

## 2 多旋翼视觉伺服模型

准确的模型是系统分析和控制的基础。一方面,建模过于繁琐,会导致控制算法设计复杂;另一方面,建模过于简单,会使得模型脱离实际,对控制效果带来消极影响。本章中,首先基于若干简化假设,建立多旋翼刚体运动学模型和刚体动力学模型,从而得到多旋翼飞行控制的刚体模型;然后以针孔模型为例,介绍视觉成像的数学模型;最后介绍了视觉伺服的定义和三维空间中雅阁比矩阵推导过程。

### 2.1 多旋翼飞行控制刚体模型

如图 2.1 所示,多旋翼飞行控制刚体模型主要包括如下两个部分。

#### 1、刚体运动学模型

运动学与质量和受力无关,只研究位置、速度、姿态和角速度等变量。多旋翼运动学模型的输入为速度和角速度,输出位置和姿态。

#### 2、刚体动力学模型

动力学既涉及运动又涉及受力情况,与物体的质量和转动惯量有关。牛顿第二定律、动能定理和动量定理等常被用来研究物体之间的相互作用。多旋翼动力学模型的输入为拉力和力矩(俯仰力矩、滚转力矩和偏航力矩),输出为速度和角速度。刚体运动学模型和动力学模型一起构成了常用的多旋翼飞行控制刚体模型。

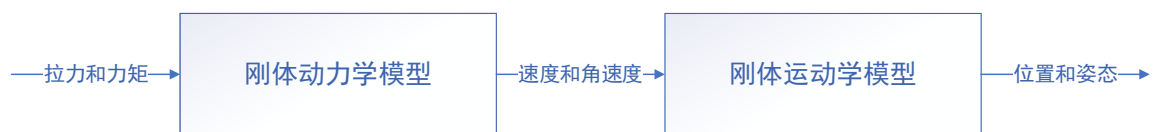


图 2.1 多旋翼飞行控制刚体模型层次结构

#### 2.1.1 假设

为了简便,在对多旋翼建模时,做出如下假设<sup>[20]</sup>。

**假设 2.1** 多旋翼是刚体。

**假设 2.2** 质量与转动惯量不变。

**假设 2.3** 多旋翼几何中心与重心一致。

**假设 2.4** 多旋翼只受重力和螺旋桨拉力,其中,重力沿  $O_e z_e$  轴正方向,而螺旋桨拉

力沿  $O_b z_b$  轴负方向。

**假设 2.5** 奇数标号的螺旋桨为逆时针转动，而偶数标号地螺旋桨为顺时针转动。

### 2.1.2 刚体运动学模型

令多旋翼的重心为  ${}^e \mathbf{p} \in \mathbb{R}^3$ ，则

$${}^e \dot{\mathbf{p}} = {}^e \mathbf{v} \quad (2.1)$$

其中， ${}^e \mathbf{v} \in \mathbb{R}^3$  表示多旋翼的速度。姿态运动学模型分 3 种：欧拉角模型、旋转矩阵模型和四元数模型，本文中计划采用欧拉角模型。

姿态变化率与集体角速度关系为

$$\begin{bmatrix} \omega_{x_b} \\ \omega_{y_b} \\ \omega_{z_b} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \cos \theta \sin \phi \\ 0 & -\sin \phi & \cos \theta \cos \phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (2.2)$$

进一步，有

$$\dot{\boldsymbol{\Theta}} = \mathbf{W} \cdot {}^b \boldsymbol{\omega} \quad (2.3)$$

其中，

$$\boldsymbol{\Theta} \triangleq \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}, \quad \mathbf{W} \triangleq \begin{bmatrix} 1 & \tan \theta \sin \phi & \tan \theta \cos \phi \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix} \quad (2.4)$$

基于**假设 2.1**，结合式 (2.1) 和 (2.3)，可得

$$\begin{cases} {}^e \dot{\mathbf{p}} = {}^e \mathbf{v} \\ \dot{\boldsymbol{\Theta}} = \mathbf{W} \cdot {}^b \boldsymbol{\omega} \end{cases} \quad (2.5)$$

### 2.1.3 刚体动力学模型

#### 1、位置动力学模型

**假设 2.4** 表明，本章仅考虑桨盘水平的多旋翼。基于**假设 2.4**，对多旋翼进行受力分析，有

$${}^e \dot{\mathbf{v}} = g \mathbf{e}_3 - \frac{f}{m} {}^e \mathbf{b}_3 \quad (2.6)$$

其中,  $f \in \mathbb{R}_+ \cup \{0\}$  表示螺旋桨总拉力的大小, 该拉力是单向的 (此处不考虑变桨距而产生负拉力的情况);  $g \in \mathbb{R}_+$  为重力加速度。直观上, 虽然拉力方向向上, 但是由于定义的  $O_b z_b$  轴取向向下为正方向, 因此式 (2.6) 中的  $-\frac{f}{m} {}^e \mathbf{b}_3$  表示拉力矢量, 进一步可以得到

$${}^e \dot{\mathbf{v}} = g \mathbf{e}_3 - \frac{f}{m} \mathbf{R} \mathbf{e}_3 = g \mathbf{e}_3 + \mathbf{R} \frac{{}^e \mathbf{f}}{m} \quad (2.7)$$

其中,  ${}^e \mathbf{f} \triangleq -f \mathbf{e}_3$ 。因为

$${}^e \mathbf{v} = \mathbf{R} {}^b \mathbf{v} \quad (2.8)$$

结合式 (2.7) 和 (2.8), 得到

$${}^b \dot{\mathbf{v}} = -[{}^b \boldsymbol{\omega}] \times {}^b \mathbf{v} + g \mathbf{R}^T \mathbf{e}_3 - \frac{f}{m} \mathbf{e}_3 \quad (2.9)$$

## 2、姿态动力模型

基于假设 2.1~假设 2.3, 在机体坐标系内建立姿态动力学方程如下:

$$\mathbf{J} {}^b \dot{\boldsymbol{\omega}} = -{}^b \boldsymbol{\omega} \times (\mathbf{J} {}^b \boldsymbol{\omega}) + \mathbf{G}_a + \boldsymbol{\tau} \quad (2.10)$$

其中,  $\boldsymbol{\tau} \triangleq [\tau_x \ \tau_y \ \tau_z]^T \in \mathbb{R}^3$  表示螺旋桨在机体轴上产生的力矩,  $\mathbf{J} \in \mathbb{R}^{3 \times 3}$  表示多旋翼的转动惯量。对于  $n_r$  旋翼来说,  $\mathbf{G}_a \triangleq [G_{a,\phi} \ G_{a,\theta} \ G_{a,\psi}]^T \in \mathbb{R}^3$  表示陀螺力矩。陀螺力矩的符号与螺旋桨旋转方向有关。根据假设 2.5 以及坐标系的定义, 单个螺旋桨的角速度矢量为  $(-1)^k \varpi_k {}^b \mathbf{b}_3$  ( $k=1, \dots, n_r$ ), 其中  $\varpi_k \in \mathbb{R}_+$  表示第  $k$  个螺旋桨的角速度 (rad/s)。因此, 单个螺旋桨产生的螺旋力矩表示为

$$\begin{aligned} \mathbf{G}_{a,k} &= {}^b \mathbf{L}_k \times {}^b \boldsymbol{\omega} \\ &= J_{RP} (-1)^k \varpi_k {}^b \mathbf{b}_3 \times {}^b \boldsymbol{\omega} \\ &= J_{RP} ({}^b \boldsymbol{\omega} \times \mathbf{e}_3) (-1)^{k+1} \varpi_k \end{aligned} \quad (2.11)$$

其中,  $J_{RP} \in \mathbb{R}_+ (\text{N} \cdot \text{m} \cdot \text{s}^2)$  表示整个电机转子和螺旋桨绕转轴的总转动惯量。这里的推导用到了  ${}^b \mathbf{b}_3 \times {}^b \boldsymbol{\omega} = -{}^b \boldsymbol{\omega} \times {}^b \mathbf{b}_3$ ,  ${}^b \mathbf{b}_3 = \mathbf{e}_3$ 。对于  $n_r$  旋翼来说, 有

$$\mathbf{G}_a = \sum_{k=1}^{n_r} J_{RP} ({}^b \boldsymbol{\omega} \times \mathbf{e}_3) (-1)^{k+1} \varpi_k \quad (2.12)$$

进一步, 因为

$${}^b\boldsymbol{\omega} \times \mathbf{e}_3 = \begin{bmatrix} \omega_{y_b} \\ -\omega_{x_b} \\ 0 \end{bmatrix} \quad (2.13)$$

可以得到

$$\begin{cases} G_{a,\phi} = \sum_{k=1}^{n_r} J_{RP} \omega_{y_b} (-1)^{k+1} \varpi_k \\ G_{a,\theta} = \sum_{k=1}^{n_r} J_{RP} \omega_{x_b} (-1)^k \varpi_k \\ G_{a,\psi} = 0 \end{cases} \quad (2.14)$$

由上式可以看出, 偏航通道上不产生陀螺力矩。

#### 2.1.4 多旋翼飞行控制刚体模型

联立式 (2.5)、式 (2.7) 和式 (2.10), 可以得到多旋翼飞行控制刚体模型欧拉角表示:

$$\begin{cases} {}^e\dot{\mathbf{p}} = {}^e\mathbf{v} \\ {}^e\dot{\mathbf{v}} = g\mathbf{e}_3 - \frac{f}{m}\mathbf{R}_b^e\mathbf{e}_3 \\ \dot{\boldsymbol{\Theta}} = \mathbf{W} \cdot {}^b\boldsymbol{\omega} \\ \mathbf{J} \cdot {}^b\dot{\boldsymbol{\omega}} = -{}^b\boldsymbol{\omega} \times (\mathbf{J} \cdot {}^b\boldsymbol{\omega}) + \mathbf{G}_a + \boldsymbol{\tau} \end{cases} \quad (2.15)$$

这些模型同时包含机体坐标系和地球坐标系。一方面, 希望在地球坐标系下表示多旋翼的位置和速度, 方便飞控手更好地确定飞行位置和飞行速度, 而且这种表示与 GPS 的测量时一致的; 另一方面, 在机体坐标系下, 拉力与力矩的表示非常直观, 而且传感器的测量也是在机体坐标系下表示。多旋翼飞行控制刚体模型的显著特点表现在  $-f/m \cdot \mathbf{R}\mathbf{e}^3$ , 意味着方向始终和  $O_b z_b$  轴负方向一致。

将多旋翼飞行刚体控制使用数学表达出来, 利于后续对多旋翼控制算法的设计、编写和实现。

## 2.2 视觉成像模型

在建立摄像机模型之前, 首先对视觉模型中常用到的诸多坐标系进行明确的定义。



### 2.2.1 坐标系定义

世界坐标系是就是惯性坐标系，在视觉模型中一般用于描述目标以及相机的绝对位置坐标，一般使用的是三维直角坐标系。

相机坐标系使用的是三维直角坐标系， $z$  轴方向指出屏幕，垂直于摄像机镜头，摄像机镜头光心作为原点  $O$ ，平面  $xOy$  平行于屏幕。

图像像素坐标系，坐标的原点在图像的左上角，两条坐标轴分别指示行数和列数，衡量单位使用图像像素，是对产生的图片的描述

图像物理坐标系同样是直角坐标系，坐标建立和图像像素坐标系一致，不同在于衡量单位是毫米，是对焦平面的测量。

齐次坐标：齐次坐标就是一个  $n$  维向量的  $n+1$  维矢量表示方法，其集合意义可理解为三维空间的剖视图，即某一维是常数的三维坐标，作为二维平面。

### 2.2.2 针孔模型

对于假想的针孔摄像机进行符号表示： $f$  是摄像机焦距， $Z$  是摄像机到物体的距离， $X$  是物体长度，则  $x = f * X / Z$ 。假设世界坐标系原点在光心，即和相机坐标系重合，如图 2.2，然后将成像平面理想化到光心的前方，这样方便进行计算。

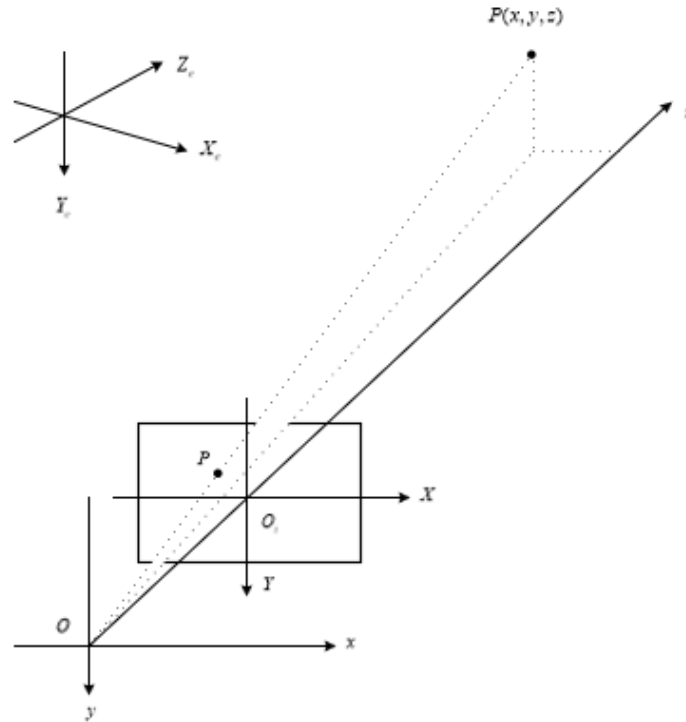


图 2.2 相机坐标系与物理坐标系映射关系

此时，给一个世界坐标系中的点  $\mathbf{P}(x, y, z)$ （单位是 mm），则由于假设，该点在相机坐标系下坐标也是  $\mathbf{P}(x, y, z)$ ，它经过投影变换产生的成像平面上图像物理坐标系下的点坐标为  $\mathbf{p}(u, v)$ （单位是 mm），易得以下关系：

$$u = f \frac{x}{z} \quad (2.16)$$

$$v = f \frac{y}{z} \quad (2.17)$$

如果用齐次坐标表示，可以表示成

$$\mathbf{p} = \mathbf{M}\mathbf{P} \quad (2.18)$$

其中

$$\mathbf{p} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \quad \mathbf{M} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{P} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.19)$$

再假设图像像素坐标系的原点在图像中心，即和图像物理坐标系重合。若  $\mathbf{p}(u, v)$ （单位



mm) 对应的图片上的像素的点为  $\tilde{\mathbf{p}}(\tilde{u}, \tilde{v})$  (单位像素, 位于图像像素坐标系下), 由成像平面到图像平面是感光元件在起作用。由于感光元件像素的横纵往往不同, 所以用  $s_x$  示感光元件横向每毫米多少像素, 即单位是像素/mm, 用  $s_y$  表示感光元件纵向每毫米多少像素, 即单位是像素/mm。则有

$$\tilde{u} = us_x \quad (2.20)$$

$$\tilde{v} = vs_y \quad (2.21)$$

带入上式 (2.20) 和 (2.21), 即

$$\tilde{\mathbf{p}} = \tilde{\mathbf{M}}\mathbf{P} \quad (2.22)$$

其中

$$\tilde{\mathbf{p}} = \begin{bmatrix} \tilde{u} \\ \tilde{v} \\ 1 \end{bmatrix}, \quad \tilde{\mathbf{M}} = \begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \tilde{\mathbf{P}} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (2.23)$$

## 2.3 视觉伺服模型

### 2.3.1 视觉伺服公式

视觉伺服指的是利用计算机视觉信息控制机器人的运动。所有基于视觉伺服的方案, 目的都是减小误差  $\mathbf{e}(t)$ 。误差  $\mathbf{e}(t)$  表示为<sup>[22]</sup>

$$\mathbf{e}(t) = \mathbf{s}(\mathbf{m}(t), \mathbf{a}) - \mathbf{s}^* \quad (2.24)$$

式中, 矢量  $\mathbf{m}(t)$  为一连串的图像测量值 (图像特征点或者物体的中心点);  $\mathbf{s}(\mathbf{m}(t), \mathbf{a})$  视觉特征点矢量;  $\mathbf{a}$  摄像头固有参数;  $\mathbf{s}^*$  期望特征值。

采用 IBVS (基于图像的视觉伺服) 时,  $\mathbf{s}$  一组可以在图像中立即获得的特征点, 为了设计出一个速度控制器, 需要求出  $\dot{\mathbf{s}}$  摄像头速度的关系。设  $\mathbf{v}_c = (\boldsymbol{\omega}, \mathbf{v})$  其中  $\mathbf{v}$  摄像头的瞬时线速度,  $\boldsymbol{\omega}$  摄像头的瞬时角速度, 则  $\dot{\mathbf{s}}$  和  $\mathbf{v}_c$  的关系是

$$\dot{\mathbf{s}} = \mathbf{L}_s \mathbf{v}_c \quad (2.25)$$



其中  $\mathbf{L}_s \in \mathbf{R}^{k \times 6}$  称为交互矩阵或者雅阁比矩阵, 根据公式 (2.24) 和 (2.25), 可以得到摄像头速度和  $\dot{\mathbf{e}}$  的关系:

$$\dot{\mathbf{e}} = \mathbf{L}_e \mathbf{v}_c \quad (2.26)$$

并且  $\mathbf{L}_e = \mathbf{L}_s$ 。将  $\dot{\mathbf{e}}$  作为输入量, 如果雅阁比矩阵已知, 就可以得到输出  $\mathbf{v}_c$ , 即摄像头的速度、角速度值, 所以关键问题在于求出雅阁比矩阵。

### 2.3.2 雅阁比矩阵

首先设置坐标系, 在摄像头坐标系下的 3-D 点坐标为  $(X, Y, Z)$ , 对应 2-D 像平面上的坐标为  $\mathbf{p} = (x, y)$ , 单位是 mm, 所以

$$\begin{cases} x = \frac{X}{Z} = (u - c_u) / f\alpha \\ y = \frac{Y}{Z} = (v - c_v) / f \end{cases} \quad (2.27)$$

$\mathbf{m}(u, v)$  是对应点以像素为单位的坐标,  $\mathbf{a} = (c_u, c_v, f, \alpha)$  是一组相机内部参数, 这一部分详见视觉成像部分。

将 (2.27) 求导, 得

$$\begin{cases} \dot{x} = \frac{\dot{X}}{Z} - \frac{\dot{Z}X}{Z^2} = \frac{\dot{X} - x\dot{Z}}{Z} \\ \dot{y} = \frac{\dot{Y}}{Z} - \frac{\dot{Z}Y}{Z^2} = \frac{\dot{Y} - y\dot{Z}}{Z} \end{cases} \quad (2.28)$$

3-D 坐标的速度可以表示为

$$\dot{\mathbf{X}} = -\mathbf{v}_c - \boldsymbol{\omega}_c \times \mathbf{X} \Leftrightarrow \begin{cases} \dot{X} = -v_x - \omega_y Z + \omega_z Y \\ \dot{Y} = -v_y - \omega_z X + \omega_x Z \\ \dot{Z} = -v_z - \omega_x Y + \omega_y X \end{cases} \quad (2.29)$$

联立 (2.28) 和 (2.29) 可以得到

$$\begin{cases} \dot{x} = -\frac{v_x}{Z} + \frac{xv_z}{Z} + xy\omega_x - (1+x^2)\omega_y + y\omega_z \\ \dot{y} = -\frac{v_y}{Z} + \frac{yv_z}{Z} - xy\omega_y + (1+y^2)\omega_x - x\omega_z \end{cases} \quad (2.30)$$

也就是

$$\dot{\mathbf{x}} = \mathbf{L}_x \mathbf{v}_c \quad (2.31)$$

其中  $\mathbf{L}_x$  可以表达为

$$\mathbf{L}_x = \begin{bmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & -\frac{1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix} \quad (2.32)$$

被称为交互矩阵，或雅阁比矩阵

视觉伺服的目的在于减小目标当前图像坐标与期望目标图像坐标之间的差，视觉伺服的意义在于通过雅阁比矩阵，建立了三维空间中的相机坐标系与二维图像坐标系之间的关联，为伺服控制打下数学基础。

## 2.4 本章小结

本章是整个系统使用视觉伺服算法控制的数学基础，第 4 章的基于视觉的追踪打击控制算法公式设计就是建立在本章的模型基础之上。本章中首先建立多旋翼非线性的飞行控制刚体模型，作为后文多旋翼控制的基础；然后以针孔模型为例，介绍了视觉成像的数学表达；最后介绍视觉伺服的概念和公式表达，并推导出雅阁比矩阵，建立三维相机坐标系和二维图像坐标系之间的关联。

### 3 基于视觉的目标追踪算法分析

在前文中已经说明的是，本文的重点设计内容在于视觉伺服控制算法部分，而不在于基于视觉的目标追踪算法的推导，所以在本章中我将主要选取一些成熟的可以在多旋翼平台在线使用的基于视觉的目标追踪算法，进行试验和比较，得出效果最优算法进行使用。

#### 3.1 基于视觉的目标追踪算法比较与选取

视觉跟踪是具有许多应用的经典计算机视觉问题。在一般跟踪任务中，仅在给定其初始位置的情况下估计图像序列中的目标的轨迹。这个问题特别具有挑战性，跟踪器必须将目标外观从非常有限的一组训练样本实现一个稳定鲁棒的滤波器，抵抗快速运动和变形。

这些经典相关滤波方法需要从一组训练样本中学习得相关滤波器。为了获得更多的训练样本，一般将一个基础的正样本进行循环移位，得到若干虚拟的负样本，如图 3.1，来训练一个分类器，这对应于对这些样本的周期性扩展<sup>[23]</sup>。基于循环样本周期性假设，通过利用快速傅立叶变换（FFT）可以实现有效的训练和检测。

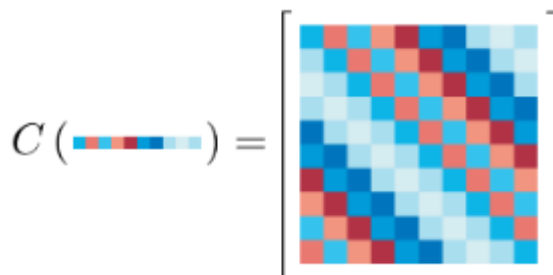

$$C(\text{vector}) = \begin{bmatrix} \text{grid} \end{bmatrix}$$

图 3.1 一维向量的循环移位<sup>[11]</sup>

本节比较的是 Boosting, MIL, KCF, TLD, MedianFlow, MOSSE 六种算法的追踪效果。对多组已经标定好的图像序列运行以上追踪算法，用方框框出追踪的目标，追踪效果如图 3.2 所示



图 3.2 追踪效果图

将每组运行结果与标定比较，测量计算每种方法的运行 FPS 和偏差值均值，获得表格 3.1。目标追踪结果和标定都是用方框将目标框出，方框的位置用  $[x,y,w,h]$  表示， $[x,y]$  表示方框左上角点坐标， $[w,h]$  分别代表方框的宽度和高度。

使用方框左上角坐标的偏差衡量追踪准确度，表中 x 偏差和 y 偏差均指方框的左上角坐标。当追踪算法在追踪过程中丢失时，计算偏差是没有任何实际意义的，所以标注“丢失”。

表格 3.1 追踪算法效果比较

方法名称	FPS	x 偏差均值(px)	y 偏差均值(px)
Boosting	55	丢失	丢失
MIL	10	丢失	丢失
KCF	110	-7.26	-1.10
TLD	10	丢失	丢失
MedianFlow	100	2.29	3.35
MOSSE	150	丢失	丢失

在目标运动比较快速的情况下,可以看到六种方法中的四种表现都完全不能投入实际应用,出现了突然丢失或者严重的漂移,对于 KCF 方法和 MedianFlow 方法,两者都可以进行使用,实验中不存在显著差异,所以选用比较熟悉常用的算法 KCF。

### 3.2 KCF 算法原理和优势

在线跟踪任务是许多计算机视觉系统的关键组成部分。在初始帧框出包含目标的初始图像块,为了训练学习出一个可以区分目标和周围环境地分类器,这个分类器可以在许多位置进行精细评估,这样可以在后续帧中检测。当然,每帧检测后都提供了更新了一个新的图像块并更新模型。

符合直觉的是专注于表征感兴趣的目标对象-分类器的正样本,但是一般认为负样本在判别方法中起到更加核心的作用,需要给与同样或者更多的相关环境-即负样本。最常用的负样本是来自不同位置和尺度的图像块。

问题的关键因素在于如何从图像中获得巨量的负样本,由于跟踪的时间敏感性,之前跟踪器在尽可能多地合并样本和降低运算掌握平衡,所以通常的做法是每帧随机选择几个样本。虽然这样做的原因是可以理解的,但文章认为负样本的欠采样是阻碍跟踪效果的主要因素。下面介绍 KCF 方法是如何解决以上问题并提高运算速度的。

#### 3.2.1 算法原理

##### 1、循环移位

文章中提出的循环移位方法可以说是及其有效的产生负样本方法,也是 KCF 方法基础。首先讨论一维单通道信号,也就是一维向量的循环移位。文章将  $n$  维列向量  $\mathbf{x}$  作为基础样本,也即是正样本,目的是使用一个正样本和通过循环移位后获得的几个负样本训练一个分类器。可以通过循环移位算子  $\mathbf{P}$  来建模向量  $\mathbf{x}$  的转换

$$\mathbf{P} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 1 \\ 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \quad (3.1)$$

那么  $\mathbf{P}\mathbf{x} = [x_n \ x_1 \ x_2 \ \cdots \ x_{n-1}]^T$  表示向量  $\mathbf{x}$  元素向右平移一个单位。将几个平移结果进行合并得到增加样本数后的循环矩阵  $\mathbf{X}$

$$\mathbf{X} = (\mathbf{P}^0 \mathbf{x} \quad \mathbf{P}^1 \mathbf{x} \quad \mathbf{P}^2 \mathbf{x} \quad \cdots \quad \mathbf{P}^{n-1} \mathbf{x})^T \quad (3.2)$$

$$\mathbf{X} = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_n & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix} \quad (3.3)$$

循环矩阵具有确定性的，完全由初始向量  $\mathbf{x}$  决定，并且无论初始向量  $\mathbf{x}$  如何，循环矩阵都可以通过离散傅里叶变换（DFT）进行对角化， $\mathbf{X}^H$  表示复共轭转置矩阵

$$\mathbf{X} = \mathbf{F} \text{diag}(\hat{\mathbf{x}}) \mathbf{F}^H \quad (3.4)$$

其中  $\hat{x}$  表示  $x$  的离散傅里叶变换结果。

## 2、一维岭回归

接下来以一维岭回归为例，介绍 KCF 方法中的核心思想，在频域中进行信号计算，并使用对角矩阵简化矩阵求逆的庞大运算量。

设训练样本集  $(\mathbf{x}_i, \mathbf{y}_i)$ ，回归函数  $f(\mathbf{x}_i) = \mathbf{w}^T \mathbf{x}_i$ ，其中  $\mathbf{w}$  是列向量的权重系数，也就是我们本次回归训练希望能得到的训练结果，通过最小二乘法求解

$$\min_{\mathbf{w}} \sum_i (f(\mathbf{x}_i) - \mathbf{y}_i)^2 + \lambda \|\mathbf{w}\|^2 \quad (3.5)$$

写成矩阵形式

$$\min_{\mathbf{w}} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|^2 + \lambda \|\mathbf{w}\|^2 \quad (3.6)$$

其中  $\mathbf{X} = [\mathbf{x}_1 \quad \mathbf{x}_2 \quad \cdots \quad \mathbf{x}_n]^T$  每一行表示一个向量， $\mathbf{x}_i, \mathbf{y}$  是列向量，求导等于零可以得到下式：

$$\mathbf{w} = (\mathbf{X}^T \mathbf{X} + \lambda \mathbf{I})^{-1} \mathbf{X}^T \mathbf{y} \quad (3.7)$$

可以看到式（3.7）中涉及矩阵求逆这种消耗计算资源的复杂操作，使计算复杂度过高。

文章中的核心观点在于如何利用循环矩阵的频域对角化特性降低求逆的运算复杂度。在前文中已经说明，所有循环矩阵都能够在频域使用离散傅里叶变换进行对角化，如式（3.4）。

使用  $\hat{\mathbf{x}}$  表示  $\mathbf{x}$  傅里叶变换后的值， $\mathbf{F}$  表示离散傅里叶矩阵， $w = e^{-j\frac{2\pi}{n}}$  有

$$\mathbf{F} = \frac{1}{\sqrt{n}} \begin{bmatrix} 1 & 1 & \cdots & 1 & 1 \\ 1 & w & \cdots & w^{n-2} & w^{n-1} \\ 1 & w^2 & \cdots & w^{2(n-2)} & w^{2(n-1)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & w^{n-1} & \cdots & w^{(n-1)(n-2)} & w^{(n-1)^2} \end{bmatrix} \quad (3.8)$$

将式 (3.4) 对角化公式带入岭回归公式得到

$$\begin{aligned} \mathbf{w} &= \left( \mathbf{F} \text{diag}(\hat{\mathbf{x}}^*) \mathbf{F}^H \mathbf{F} \text{diag}(\hat{\mathbf{x}}) \mathbf{F}^H + \lambda \mathbf{F}^H \right)^{-1} \mathbf{F} \text{diag}(\hat{\mathbf{x}}^*) \mathbf{F}^H \mathbf{y} \\ &= \left( \mathbf{F} \text{diag}(\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda) \mathbf{F}^H \right)^{-1} \mathbf{F} \text{diag}(\hat{\mathbf{x}}^*) \mathbf{F}^H \mathbf{y} \\ &= \mathbf{F} \text{diag} \left( \frac{\hat{\mathbf{x}}^*}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda} \right) \mathbf{F}^H \mathbf{y} \end{aligned} \quad (3.9)$$

根据对角化矩阵在频域中的性质, 对两边同时求傅里叶变换, 用  $\Gamma(\mathbf{x})$  表示傅里叶变换

$$\Gamma(\mathbf{w}) = \Gamma^* \left( \Gamma^{-1} \left( \frac{\hat{\mathbf{x}}^*}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda} \right) \right) \odot \Gamma(\mathbf{y}) \quad (3.10)$$

于是

$$\hat{\mathbf{w}} = \frac{\hat{\mathbf{x}} \odot \hat{\mathbf{y}}}{\hat{\mathbf{x}}^* \odot \hat{\mathbf{x}} + \lambda} \quad (3.11)$$

$\hat{\mathbf{w}}$  只是  $\mathbf{w}$  的傅里叶变换结果, 所以

$$\mathbf{w} = \Gamma^{-1}(\hat{\mathbf{w}}) \quad (3.12)$$

在本式中可以看到矩阵求逆转变成了向量的点积运算, 大大降低了运算复杂度, 提高计算速度, 这也是 KCF 方法能够快速计算的核心思想。

### 3.2.2 算法优势

在 KCF 方法中, 我认为它最大的贡献就是开发了一种通过相对平移有效地产生大量负样本的工具-循环移位, 并利用了循环矩阵在频域内良好的可对角化特性极大地简化了核运算的运算复杂度。算法优缺点如表 3.1





表格 3.1 KCF 方法优缺点总结

KCF 方法优点	KCF 方法缺点
1. 对目标区域循环移位，在不同的相对平移中产生大量的负样本，并通过循环移位构建出循环矩阵，具有两汉的频域特性	1. KCF 因为在跟踪过程当中目标框是已经设定好的，从始至终大小不发生变化
2. 在频域中分析问题，利用循环矩阵的傅里叶域中可对角化特点巧妙地降低基于 Kernel 的岭回归跟踪器运算复杂度，构成了快速跟踪的基础	2. KCF 对在跟踪过程当中当目标被遮挡时候的处理问题没有进行很好的解决，不过这也是跟踪界普遍存在难题

3.2 本章小结

本章中主要选取一些成熟的可以在多旋翼平台在线使用的基于视觉的目标追踪算法，进行试验并比较算法运算速度和精度。选取的六种算法中只有两种在目标移动较快时可以追踪到目标，最终选用了 KCF 方法作为本文的基于视觉的目标追踪方法，并简单介绍 KCF 方法核心思想和主要优势。

## 4 基于视觉的追踪打击控制

在第二章中, 本文已经通过研究多旋翼飞行器的运动学模型和动力学模型, 建立多旋翼飞行控制刚体模型, 并介绍视觉伺服模型概念和三维空间中的推导。为了能够便捷地设计基于视觉的追踪打击控制器, 可以进一步简化多旋翼刚体模型和视觉伺服模型到横侧向通道和纵向通道, 在两个通道中分别设计多旋翼基于视觉的追踪打击控制算法。

本章内容旨在设计多旋翼基于视觉的追踪打击控制算法, 可以在多旋翼追踪目标的过程中不脱离视域内并使目标向图像中心收敛。本章将在 4.1 简化和线性化多旋翼刚体模型到纵向和横侧向通道, 4.2 中解耦横侧向和纵向通道的视觉伺服模型, 4.3 中建立多旋翼通道模型和视觉伺服模型的关系, 并在后两个小节中分别设计横侧向和纵向通道的基于视觉的追踪打击控制算法。因为本文已经在第三章中介绍使用的基于视觉的图像追踪算法, 得到每一帧的目标图像坐标。至此已经有充分的理论基础和前置条件完成全部的算法设计任务。

### 4.1 多旋翼简化的线性模型

在第二章中建立的多旋翼飞行控制刚体模型是非线性模型, 为了便于控制器控制, 需要对非线性模型进行适当的简化和线性化处理<sup>[20]</sup>。基于第二章介绍的多旋翼非线性模型 (2.15), 忽略  ${}^b\boldsymbol{\omega} \times (\mathbf{J} \cdot {}^b\boldsymbol{\omega}) + \mathbf{G}_a$ , 得到如下简化结果。

$${}^c\dot{\mathbf{p}} = {}^c\mathbf{v} \quad (4.1)$$

$${}^c\dot{\mathbf{v}} = g\mathbf{e}_3 - \frac{f}{m}\mathbf{R}_b^c\mathbf{e}_3 \quad (4.2)$$

$$\boldsymbol{\Theta} = \mathbf{W} \cdot {}^b\boldsymbol{\omega} \quad (4.3)$$

$$\mathbf{J} \cdot {}^b\boldsymbol{\omega} = \boldsymbol{\tau} \quad (4.4)$$

位置方程可以进一步写成为

$$\begin{cases} \ddot{p}_{x_e} = -\frac{f}{m}(\sin \psi \sin \theta + \cos \psi \sin \theta \cos \phi) \\ \ddot{p}_{y_e} = -\frac{f}{m}(-\cos \psi \sin \phi + \cos \phi \sin \theta \cos \psi) \\ \ddot{p}_{z_e} = g - \frac{f}{m}(\cos \theta \cos \phi) \end{cases} \quad (4.5)$$

下文为了表达简便,将忽略上标、下标,即默认  $\mathbf{p} = {}^c\mathbf{p}$ ,  $\mathbf{v} = {}^c\mathbf{v}$ ,  $\boldsymbol{\omega} = {}^b\boldsymbol{\omega}$ ,  $\mathbf{R} = \mathbf{R}_b^c$ 。

从公式 (4.1) ~ (4.4) 可见,多旋翼飞行控制系统是一个典型的非线性系统,这使得多旋翼的分析和控制器设计十分复杂。此外,多旋翼还具有欠驱动、强耦合、阶数高的特点。因此需要更具多旋翼飞行特点对非线性模型进行化简,然后基于化简的系统模型设计控制器。假设多旋翼的飞行特点是俯仰角和滚转角都非常小,总拉力约等于多旋翼的重力。这些假设进一步可以写为

$$\sin \phi \approx \phi, \cos \phi \approx 1, \sin \theta \approx \theta, \cos \theta \approx 1, f \approx mg \quad (4.6)$$

此时,式 (4.3) 中的矩阵  $\mathbf{W}$  似为单位矩阵  $\mathbf{I}_3$  式 (4.2) 中的  $\mathbf{Re}_3$  变为

$$\mathbf{Re}_3 \approx \begin{bmatrix} \theta \cos \psi + \phi \sin \psi \\ \theta \sin \psi - \phi \cos \psi \\ 1 \end{bmatrix} \quad (4.7)$$

最终,原始模型 (4.1) ~ (4.4) 解耦为三个线性模型,即水平位置通道模型,高度通道模型和姿态模型。

### 1、水平位置通道模型

根据小角度假设,式 (4.5) 前两个式子化简为

$$\begin{cases} \dot{\mathbf{p}}_h = \mathbf{v}_h \\ \dot{\mathbf{v}}_h = -g\mathbf{A}_\psi \boldsymbol{\Theta}_h \end{cases} \quad (4.8)$$

其中,

$$\mathbf{p}_h = \begin{bmatrix} p_x \\ p_y \end{bmatrix}, \mathbf{R}_\psi = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix}, \mathbf{A}_\psi = \mathbf{R}_\psi \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}, \boldsymbol{\Theta}_h = \begin{bmatrix} \phi \\ \theta \end{bmatrix}$$

在水平通道模型 (4.8) 中,因为  $-g\mathbf{A}_\psi$  已知,  $-g\mathbf{A}_\psi$  可以看作输入,  $\mathbf{p}_h$  看成输出,水平位置通道模型 (4.8) 实质上是一个线性模型。

## 2、高度通道模型

根据小角度假设，式（4.5）第三个式子化简为

$$\begin{cases} \dot{p}_h = v_z \\ \dot{v}_z = g - \frac{f}{m} \end{cases} \quad (4.9)$$

与水平通道模型化简不同， $g - \frac{f}{m}$ 不是高阶无穷小，因此不能忽略，否则高度通道没有输入了。显然，高度通道模型也是一个线性模型。

## 3、姿态模型

联立（4.3）和（4.4）可得到如下姿态模型：

$$\begin{cases} \dot{\Theta} = \omega \\ \mathbf{J}\dot{\omega} = \tau \end{cases} \quad (4.10)$$

这也是一个线性模型。

将多旋翼模型从非线性模型线性化后，得到水平通道模型、高度通道模型、姿态模型三个通道模型，可以非常直观地得到期望多旋翼速度、角速度和控制向量  $[\theta \ \phi \ \psi \ f]$  之间的线性方程关系，为后文多旋翼通道控制打下基础。

## 4.2 视觉伺服模型解耦

在上一节中，已经将多旋翼非线性的刚体模型经过简化和线性化，推导成解耦地水平通道和高度通道模型。本节中将对视觉伺服公式纵向通道和横侧向通道解耦，为后文中对纵向通道和横侧向通道分别进行伺服控制算法设计打下数学基础。

$\mathbf{L}_x$  雅阁比矩阵的三维表达，与  $[v_x, v_y, v_z, \omega_x, \omega_y, \omega_z]$  全部相关，因为多旋翼是欠驱动，六个变量不互相独立，仍然有四个变量未知，为了计算方便并且消掉一些影响不大的变量，还可以对三维空间进行水平竖直分解，将分解后的线性方程组组成新的雅阁比矩阵。

水平分解时，在 x-z 平面上，三个自由度  $v_x, v_z, \omega_y$

$$\dot{x} = \frac{\dot{X}}{Z} - \frac{\dot{Z}X}{Z^2} = \frac{\dot{X} - x\dot{Z}}{Z} \quad (4.11)$$

由（2.29）得

$$\dot{X} = -v_x - \omega_y Z \quad (4.12)$$

$$\dot{Z} = -v_z - \omega_y X \quad (4.13)$$

联立得

$$\dot{x} = -\frac{v_x}{Z} + \frac{xv_z}{Z} - (1+x^2)\omega_y \quad (4.14)$$

竖直分解时，在 y-z 平面上，三个自由度  $v_y, v_z, \omega_x$

$$\dot{y} = \frac{\dot{Y}}{Z} - \frac{\dot{Z}Y}{Z^2} = \frac{\dot{Y} - y\dot{Z}}{Z} \quad (4.15)$$

由 (2.29) 得

$$\dot{Y} = -v_y + \omega_x Z \quad (4.16)$$

$$\dot{Z} = -v_z - \omega_x Y \quad (4.17)$$

联立得

$$\dot{y} = -\frac{v_y}{Z} + \frac{yv_z}{Z} + (1+y^2)\omega_x \quad (4.18)$$

由 (4.14) 和 (4.18)，可以联立出分解后  $\mathbf{L}$

$$\mathbf{L} = \begin{bmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} & 0 & -(1+x^2) \\ 0 & -\frac{1}{Z} & \frac{y}{Z} & 1+y^2 & 0 \end{bmatrix} \quad (4.19)$$

由式  $\dot{\mathbf{e}} = \mathbf{L}_e \mathbf{v}_e$ ，并且  $\mathbf{L}_e = \mathbf{L}$ 。

虽然在本小节中使用的是相机坐标系下的三维坐标，但是考虑到相机固连在机架上，可以合理地近似认为，二者重心重合，位置姿态、速度、角速度一致。这样就建立了多旋翼位姿与目标在图像中坐标的关系，可以实现控制闭环。

### 4.3 多旋翼的视觉伺服模型

前两个小节中已经将多旋翼刚体模型和三维视觉伺服模型分别简化到横侧向通道和纵向通道，本节考虑建立多旋翼通道模型与视觉伺服通道模型的关系，以便于控制器设

计。

在纵向通道, 图像坐标系中控制期望为  $y_d = 0$ , 以此保证目标在图像中像中心收敛, 防止目标脱离视线范围。因此可以得到  $e_y = y - y_d$ , 由视觉伺服模型可知

$$\dot{e}_y = \mathbf{L}_y \mathbf{v}_c \quad (4.20)$$

由式 (4.18) 可以得知竖直分解后展开式如下, 其中速度在相机系下

$$\dot{e}_y = -\frac{v_{yd}}{Z} + \frac{e_y v_z}{Z} + (1 + e_y^2) \omega_x \quad (4.21)$$

根据式 (4.21) 和式 (4.9) 可以整理出

$$\begin{cases} \dot{v}_z = -\frac{f}{m} \sin \theta \\ \dot{v}_y = g - \frac{f}{m} \cos \theta \\ \dot{\theta} = \omega_x \\ \dot{e}_y = -\frac{v_y}{Z} + \frac{e_y v_z}{Z} + (1 + e_y^2) \omega_x \end{cases} \quad (4.22)$$

进一步地, 设  $\mathbf{x} = [e_y \quad v_y \quad v_z \quad \theta]^T$ ,  $\mathbf{u} = [f \quad \omega_x]^T$

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{\Gamma}[\mathbf{x}(t), t] + \mathbf{B}[\mathbf{x}(t), t] \mathbf{u}(t) + \mathbf{c} \\ y(t) &= [1 \quad 0 \quad 0 \quad 0] \mathbf{x}(t) \end{aligned} \quad (4.23)$$

在横侧向通道上, 将图像坐标系中  $x_d = 0$  作为控制期望, 以此保证目标在图像中像中心收敛, 防止目标脱离视线范围。因此可以得到  $e_x = x - x_d$ , 由视觉伺服模型可知

$$\dot{e}_x = \mathbf{L}_x \mathbf{v}_c \quad (4.24)$$

由式 (4.14) 可以得知水平分解后展开式, 其中速度在相机系下

$$\dot{e}_x = -\frac{v_x}{Z} + \frac{e_x v_z}{Z} + (1 + e_x^2) \omega_y \quad (4.25)$$

因为式 (4.8) 中速度在地球坐标系下, 而本节及以后所有推导都统一建立在相机坐标系下, 所以式 (4.8) 速度乘以转换矩阵

$$\mathbf{R}_e^b = \begin{bmatrix} \cos \theta \cos \psi & \cos \theta \sin \psi & -\sin \theta \\ \sin \theta \cos \psi \sin \phi - \sin \psi \cos \phi & \sin \theta \sin \psi \sin \phi + \cos \psi \cos \phi & \cos \theta \sin \phi \\ \sin \theta \cos \psi \cos \phi + \sin \psi \sin \phi & \sin \theta \sin \psi \cos \phi - \cos \psi \sin \phi & \cos \theta \cos \phi \end{bmatrix} \quad (4.26)$$

因为前文已经说明，相机固连在机架上，所以可以合理近似认为相机和机体的位姿、速度、角速度一致，所以可以容易地再转换到相机系中，结合式（4.25）和可以整理出下式

$$\begin{cases} \dot{v}_z = -g \cos \theta \cos \psi (\phi \sin \psi + \theta \cos \psi) + \cos \theta \sin \psi g (\phi \cos \psi - \theta \sin \psi) \\ \dot{v}_x = -g (\sin \theta \cos \psi \sin \phi - \sin \psi \cos \phi) (\phi \sin \psi + \theta \cos \psi) \\ \quad + (\sin \theta \sin \psi \sin \phi + \cos \psi \cos \phi) (g \phi \cos \psi - g \theta \sin \psi) \\ \dot{\psi} = \omega_y \\ \dot{e}_x = -\frac{v_x}{Z} + \frac{e_x v_z}{Z} - (1 + e_x^2) \omega_y \end{cases} \quad (4.27)$$

进一步地，设  $\mathbf{x} = [e_x \quad v_x \quad v_z \quad \psi]^T$ ， $\mathbf{u} = [\omega_y \quad \omega_z]^T$

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{\Gamma}[\mathbf{x}(t), t] + \mathbf{B}[\mathbf{x}(t), t] \mathbf{u}(t) + \mathbf{c} \\ y(t) &= [1 \quad 0 \quad 0 \quad 0] \mathbf{x}(t) \end{aligned} \quad (4.28)$$

本文设计的基于视觉的追踪打击控制算法目的在于，控制输入  $[f \quad \omega_x \quad \omega_y \quad \omega_z]$ ，使得输出量  $y = [e_y \quad e_x]^T$  趋近于零，控制效果上表现为在多旋翼追踪打击目标过程中保持目标在相机视域内并向中心收敛，最终可以打击到目标。

考虑到已经在 4.1 节中将多旋翼刚体模型简化和线性化为横侧向和纵向通道模型，在 4.2 节中，已将视觉伺服模型简化为横侧向和纵向通道模型，可以分别在两个通道上进行伺服控制算法设计。

#### 4.4 纵向通道控制器设计

本实验挑选  $[f, \omega_x]$  为控制向量，即电机拉力和力矩，这二者因为是控制中的底层环，可以直接控制，速度快并且没有中间过程，符合实验要求。

在俯仰平面上，仍将图像坐标系中  $y_d = 0$  作为控制期望， $e_y = y - y_d$  为图像偏差，由式（4.20）和视觉伺服模型可知下式，其中速度在相机系下

$$\dot{e}_y = -\frac{v_{yd}}{Z} + \frac{e_y v_z}{Z} + (1 + e_y^2) \omega_x \quad (4.29)$$

在俯仰平面中,  $e_y, v_y, \theta$  是可测量值,  $v_{yd}$  是可计算值, 我们需要用可测量值、计算值和一些已知量来表示控制向量  $[f, \omega_x]$ , 从而完成控制闭环。

对  $y$  方向速度和俯仰角速度进行比例控制, 即

$$v_{yd} = k_1 e_y \quad (4.30)$$

$$\omega_x = -k_2 e_y \quad (4.31)$$

将式 (4.30) (4.31) 代入式 (4.29) 可化简得

$$\dot{e}_y = -\left(\frac{k_1}{Z} - \frac{v_z}{Z} + k_2(1 + e_y^2)\right) e_y \quad (4.32)$$

即  $\dot{e}_y = -\lambda e_y$  形式, 只需保证  $\lambda = \left(\frac{k_1}{Z} - \frac{v_z}{Z} + k_2(1 + e_y^2)\right) > 0$  就能确定  $e_y$  向图像中心收敛,

即只需保证  $k_1 > v_{z\max}$  就可以保证  $e_y$  向图像中心收敛,  $v_{z\max}$  可以在实验中得到。

对期望速度和期望角度的控制并非直接控制, 而是有中间过程的, 所以对达到期望角度和速度的过程进行比例控制, 即

$$\dot{v}_y = -k_4 (v_y - v_{yd}) \quad (4.33)$$

$$\omega_x = -k_3 (\theta - \theta_d) \quad (4.34)$$

将式 (4.31) 与式 (4.34) 合并可得到  $\omega_x$  的表达式

$$\omega_x = -k_2 e_y - k_3 (\theta - \theta_d) \quad (4.35)$$

由式 (4.9) 得到的高度通道模型与式 (4.22) 联立可以得到  $f$  的表达式:

$$f = \frac{m}{\cos \theta} (k_4 (v_y - k_1 e_y) + g) \quad (4.36)$$

将式 (4.35) (4.36) 作为控制向量  $[f, \omega_x]$  的输入表达式输入多旋翼飞行器模型, 即在俯仰平面上实现基于视觉伺服的多旋翼追踪打击控制。



#### 4.5 横侧向通道控制器设计

同样本实验在水平面上挑选角速度 $[\omega_z, \omega_y]$ 为控制向量,即力矩,因为是控制中的底层环,可以直接控制,速度快并且没有中间过程,符合实验要求。

在水平面上,仍将图像坐标系中 $x_d = 0$ 作为控制期望, $e_x = x - x_d$ 作为图像偏差,由式(4.24)可以得知水平分解后展开式,其中速度在相机系下

$$\dot{e}_x = -\frac{v_x}{Z} + \frac{e_x v_z}{Z} - (1 + e_x^2) \omega_y \quad (4.37)$$

在实验中, $e_x, \phi$ 是可测量值,我们需要用可测量值、计算值和一些已知量来表示控制向量 $[\omega_z, \omega_y]$ ,从而完成控制闭环。

首先考虑水平方向的收敛性。因为希望能将滚转角 $\phi$ 作为以后避障实验中用到的控制量,所以在本次基于视觉伺服的导航实验中希望滚转角 $\phi$ 动态为零。在横向上对多旋翼进行受力分解,可以得到

$$\dot{v}_x = \frac{f}{m} \sin \phi \quad (4.38)$$

之前假设多旋翼飞行特点是滚转角和俯仰角很小(小于10度),总拉力约等于多旋翼的重力,所以可以进一步简化得到 $\dot{v}_x = g\phi$ ,当滚转角 $\phi$ 动态为零时, $\dot{v}_x$ 动态为零,所以 $v_x$ 很小可以忽略。

对于偏航角进行比例控制,即

$$\omega_y = k_5 e_x \quad (4.39)$$

所以水平分解后的视觉伺服公式可以化简成

$$\dot{e}_x = -\left(-\frac{v_z}{Z} + k_5(1 + e_x^2)\right) e_x \quad (4.40)$$

即 $\dot{e}_x = -\lambda e_x$ 形式,只需保证 $\lambda = -\frac{v_z}{Z} + k_5(1 + e_x^2) > 0$ 就能确定 $e_x$ 向图像中心收敛,即只需

保证 $k_5$ 足够大,并且距离不过小。因为在视觉实验中距离过大和过小都会严重影响图像

算法的识别精确度,本次视觉伺服实验并不应用于距离过小的情况,距离足够小时可以使用运动预测等方法打击,所以上式在应用区域内收敛。

最后为了保持滚转角 $\phi$ 动态为零,对其进行比例控制,设 $\phi_d=0$

$$\omega_z = k_6 (\phi - \phi_d) \quad (4.41)$$

所以由式(4.39)和(4.41)得到控制向量 $[\omega_z, \omega_y]$ 。

#### 4.6 仿真测试和性能分析

在 MATLAB 上进行示意仿真,利用第二章中的针孔模型创建一个 Camera 类作为虚拟的相机,在虚拟相机中有一个虚拟焦平面,设质点目标在焦平面上的投影为目标在图像坐标系下的坐标,从而满足目标点图像坐标已知的控制算法输入条件,开始视觉伺服的示意仿真测试。

对俯仰平面的视觉伺服控制过程进行示意仿真,侧视图如图 4.1, Y 轴, Z 轴在相机系。

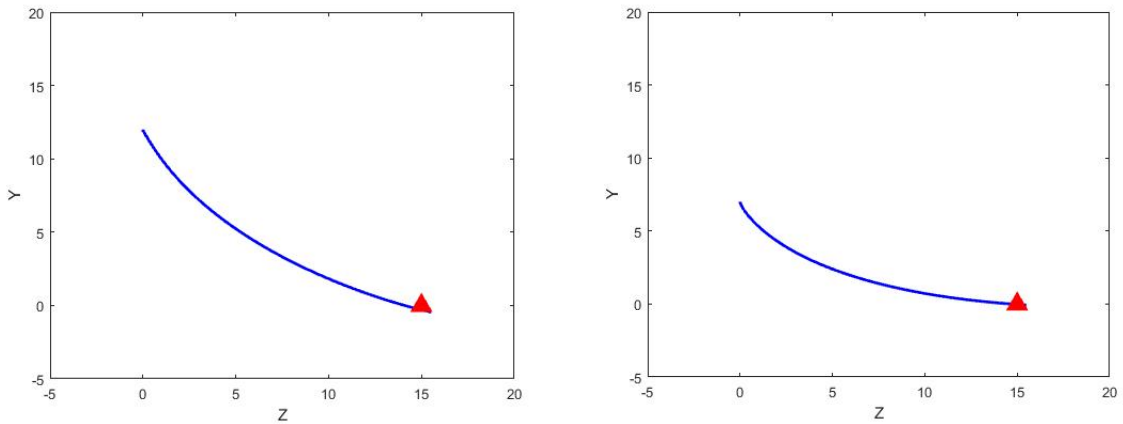


图 4.1 不同起点的俯仰平面多旋翼追踪打击示意仿真轨迹

对水平面的视觉伺服控制过程进行示意仿真,俯视图如图 4.2, X 轴, Z 轴在相机系。

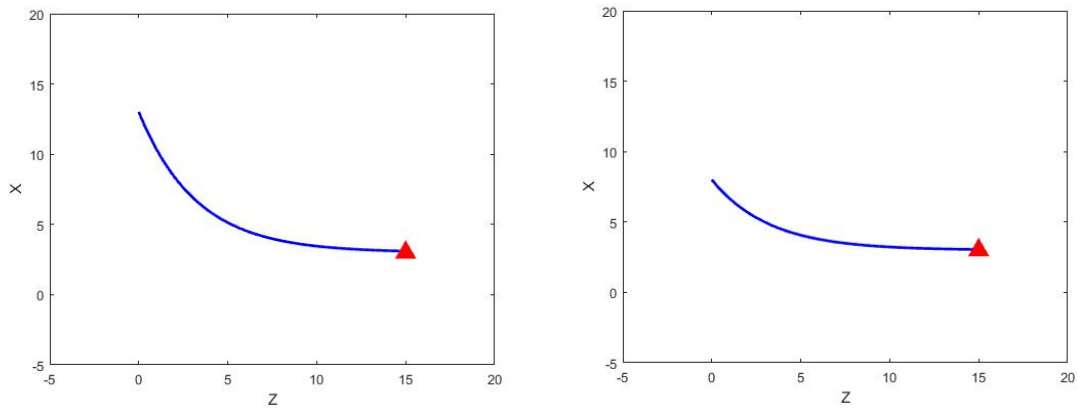
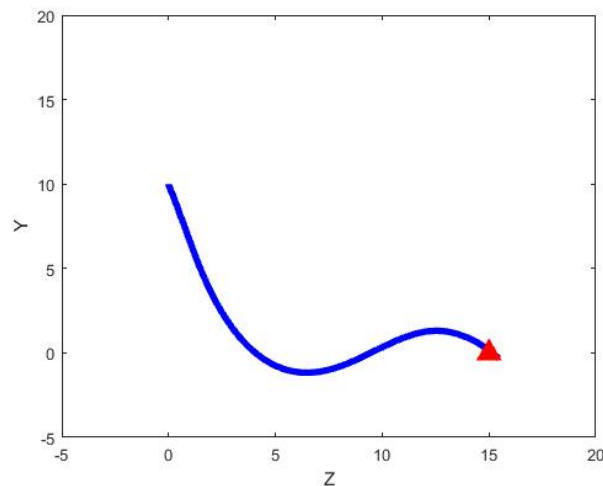


图 4.2 不同起点的水平面多旋翼追踪打击示意仿真轨迹

实验中需要设定的参数有  $k_1, k_2, k_3, k_4, \theta_d, k_5$ ，通过设定这些参数可以影响系统的稳定性、轨迹的曲线和逼近目标速度的大小，其中  $k_1, k_2, k_3, k_4, \theta_d$  在俯仰平面上影响控制效果， $k_5$  在水平面上影响控制效果，本文将对这些参数的影响进行分别研究，并显示当参数值过大时轨迹曲线的变化。

参数  $k_1$  是对  $y$  方向图像坐标系偏差的比例控制，前文中已经说明，为了保证  $e_y$  稳定收敛于零点，需要  $k_1 > v_{z\max}$ ， $k_1$  因为是对速度的控制可以保持一个较大的变化范围，但是当  $k_1$  过大时，速度比例控制为主，容易出现  $y$  方向震荡，如图 4.3， $Y$  轴， $Z$  轴在相机系。

图 4.3  $k_1$  过大时多旋翼追踪轨迹

参数  $k_2$  是对俯仰方向图像坐标系偏差的比例控制，因为在式 (4.4) 中  $k_2$  是对角速度的直接控制，对视角影响很大，所以稳定变化范围较小，当  $k_2$  过大时曲线变化剧烈并且不容易稳定，如图 4.4，Y 轴，Z 轴在相机系。

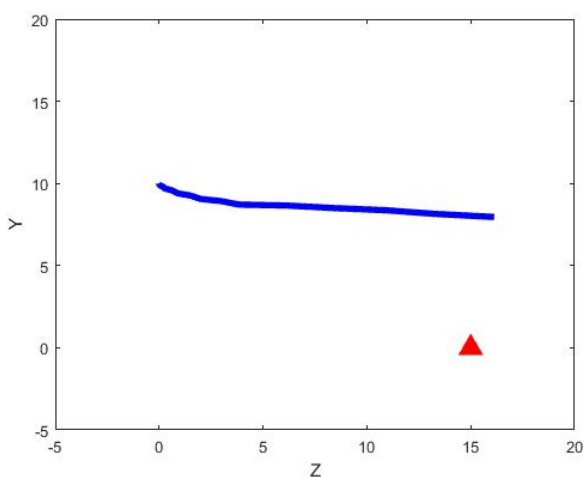


图 4.4  $k_2$  过大时多旋翼追踪轨迹

参数  $k_3$  是对实际角度与期望角度  $\theta_d$  偏差的比例控制，见式 (4.7)，目的是控制实际角度到达期望角度的中间过程，因为是对角速度的直接控制，对视角影响较大，变化范围较小，当  $k_3$  过大时，曲线变化剧烈并且不容易稳定，如图 4.5，Y 轴，Z 轴在相机系。

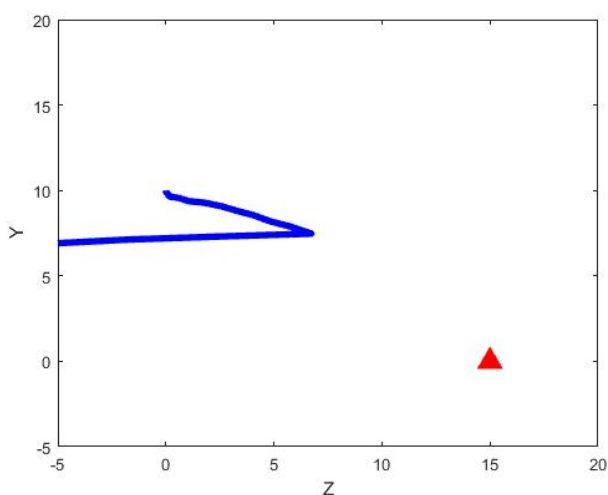


图 4.5  $k_3$  过大时多旋翼追踪轨迹

参数  $k_4$  是对 y 方向速度与速度期望偏差的比例控制，目的是控制实际 y 方向速度到

达期望速度的中间过程，可变化范围较大。需要注意的是根据式 (4.9) 计算拉力  $f$  的公式，当  $k_4$  过大时，可能出现  $f$  饱和，计算值为负值的情况，这时需要采用额外的角度变化来补偿拉力  $f$  的不足，即当  $f$  饱和时，增大  $k_2$  设计值。如图 4.6 为  $k_4$  过大时追踪轨迹，Y 轴，Z 轴在相机系。

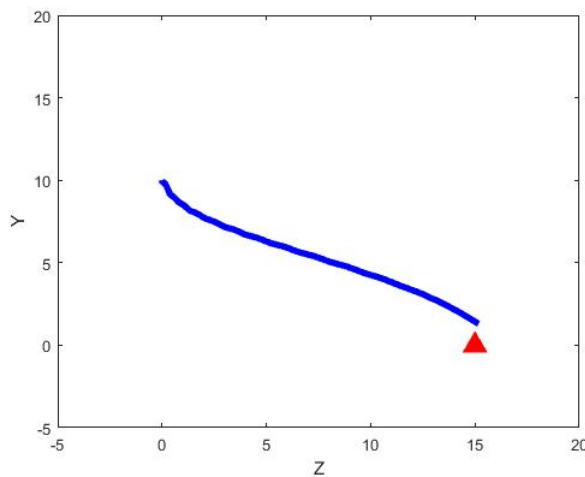
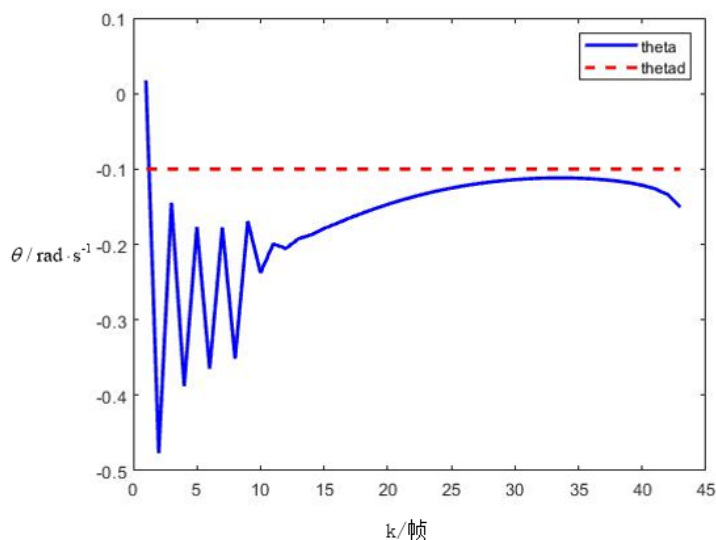
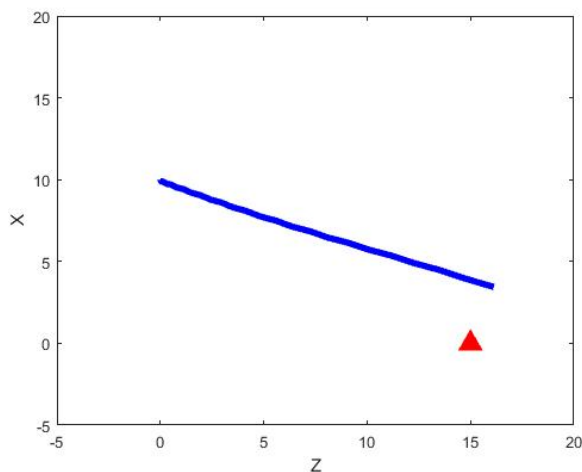


图 4.6  $k_4$  过大时多旋翼追踪轨迹

参数  $\theta_d$  用于保持多旋翼飞行器始终向目标方向飞行，即向 Z 正方向飞行， $\theta_d$  越小， $\dot{v}_z$  越大，飞行器越快到达目标。因为在式 (4.5) 中  $\omega_x$  由两项之和构成，所以  $\theta$  随时间向  $\theta_d$  变化，但是没有达到  $\theta_d$ 。图 4.7 可见，当  $\theta_d = -0.1$  时，去除首尾几帧震荡比较剧烈，中间过程  $\theta$  始终向 -0.1 收敛。

图 4.7 追踪过程  $\theta$  时域图

在水平面上进行对参数  $k_5$  进行影响分析, 示意仿真为俯视图, 相机系下坐标轴 X 轴, Z 轴如图 4.8 所示, 为了去除与水平面控制无关的因素影响, 设多旋翼以恒定速度向机头方向飞行。 $k_5$  是对水平方向偏差的比例控制, 因为是对视角的控制, 不能过大, 当  $k_5$  过大时曲线不稳定。

图 4.8  $k_5$  过大时多旋翼追踪轨迹

## 4.7 本章小结

本章是本文的核心章节, 将多旋翼的刚体模型和视觉伺服模型简化到横侧向和纵向通道, 详细描述了多旋翼飞行器在目标图像坐标已知的情况下进行视觉伺服的过程, 并



分别在两通道中进行核心算法公式推导、设计和验证。其中 4.1 节将多旋翼刚体模型简化和线性化为横侧向和纵向通道模型，在 4.2 节中，将视觉伺服模型简化为横侧向和纵向通道模型，在 4.3 节中建立多旋翼通道模型和视觉伺服通道模型的关系，4.4 节和 4.5 分别在纵向通道和横侧向通道设计基于视觉的目标追踪打击控制算法，明确了已知量、可测量量和输出控制向量，将设计后的算法带入 4.2 节视觉伺服模型证明了算法的收敛性，即证明算法理论上可行。最后在 4.6 节中进行示意仿真，探讨设置的每个参数对控制有何影响、影响大小和参数本身的大致区间，为后文仿真中的参数整定打下基础。

至此，本文涉及的算法设计部分全部完成，并证明了其理论上的可行性，但是还需要在仿真平台上进行进一步检验，对误差进一步分析，这一部分将在下一章中介绍。



## 5 仿真平台和综合实验

在第 2,3,4 章,已经通过将建立数学模型,简化模型,分别在纵向通道和横侧向通道设计基于视觉的多旋翼追踪打击控制算法,并已经证明算法的收敛性。本章将基于视景仿真通过实验进一步验证算法可行性,并通过改变目标运动方式和运动速度做多组实验,探讨参数变化对控制效果的影响。

首先在 5.1 中介绍仿真环境搭建工具和场景,然后将基于视觉的伺服控制打击系统分成功能独立四个模块,分别介绍模块的结构、功能和输入输出,最后控制变量分别改变目标运动速度和运动方式,计算、评价控制过程的精度,以此探讨改变目标运动状态对于控制效果的影响。

### 5.1 仿真环境搭建

V-Realm Builder 是一个强大的三维物体结构工具包,在 MATLAB 自带的安装库中,它提供了最小化文件存储容量的工具,它生成的三维物体和虚拟世界可以用浏览器的 VRML 插件进行观察,而且使复杂物体建模简单化。

V-Realm Builder 采用树状图对场景中的三维物体进行多元化描述,树状图中每个节点对应对象的某一个特征,可以直接对界面中的每个节点信息编辑,相对应的场景信息立即更改;也可以很容易地对节点进行增加、编辑、修改和删除。节点可以设置多种属性,比如尺寸、中心点位置、轴角表示的旋转运动和平移运动等,可以通过修改这些属性来改变对象的初始位姿;并且在节点中可以很容易地添加不同属性的子类,这些子类既可以是和父类具有耦合的运动关系的小部件和视角,也可以是描述父类某种特性的属性声明,这一切取决于创建子类时定义的子类类型。为了更好观察仿真全过程,同时可以建立多个观察点获取不同角度的图像信息,通过设置从属关系,既可以让观察点和一些物体一起运动,也可以静止在某处观察,并且支持设置观察点的位置、角度、从属等信息,全面展现仿真过程中的各个细节。除此之外,为了环境更加逼真,物体更加写实,V-realm Builder 中还支持添加纹理、涂装、色彩,添加不同角度的环境灯,使搭建的场景具有各种物理特征。

如图 5.1,为本文实验中在 V-realm Builder 内搭建的场景、目标标志桶和多旋翼飞行器。



在前文中已经阐明，本文的研究重点在于视觉伺服控制算法的研究，在于当目标在图像坐标系中的坐标已知时，使用控制令多旋翼飞行器准确打击到目标，并且过程中始终保持目标在视域之内，所以利用图像处理和图像跟踪算法获取每一帧目标的坐标并非重点。所以在建立场景时应该使场景尽量简单明了，不出现对跟踪算法和图像处理的干扰信息，干扰对控制算法结果误差的测评。本文中搭建的场景只有地板、目标和飞行器，正是为了减小获取目标图像坐标的难度，专注于控制算法仿真实验。

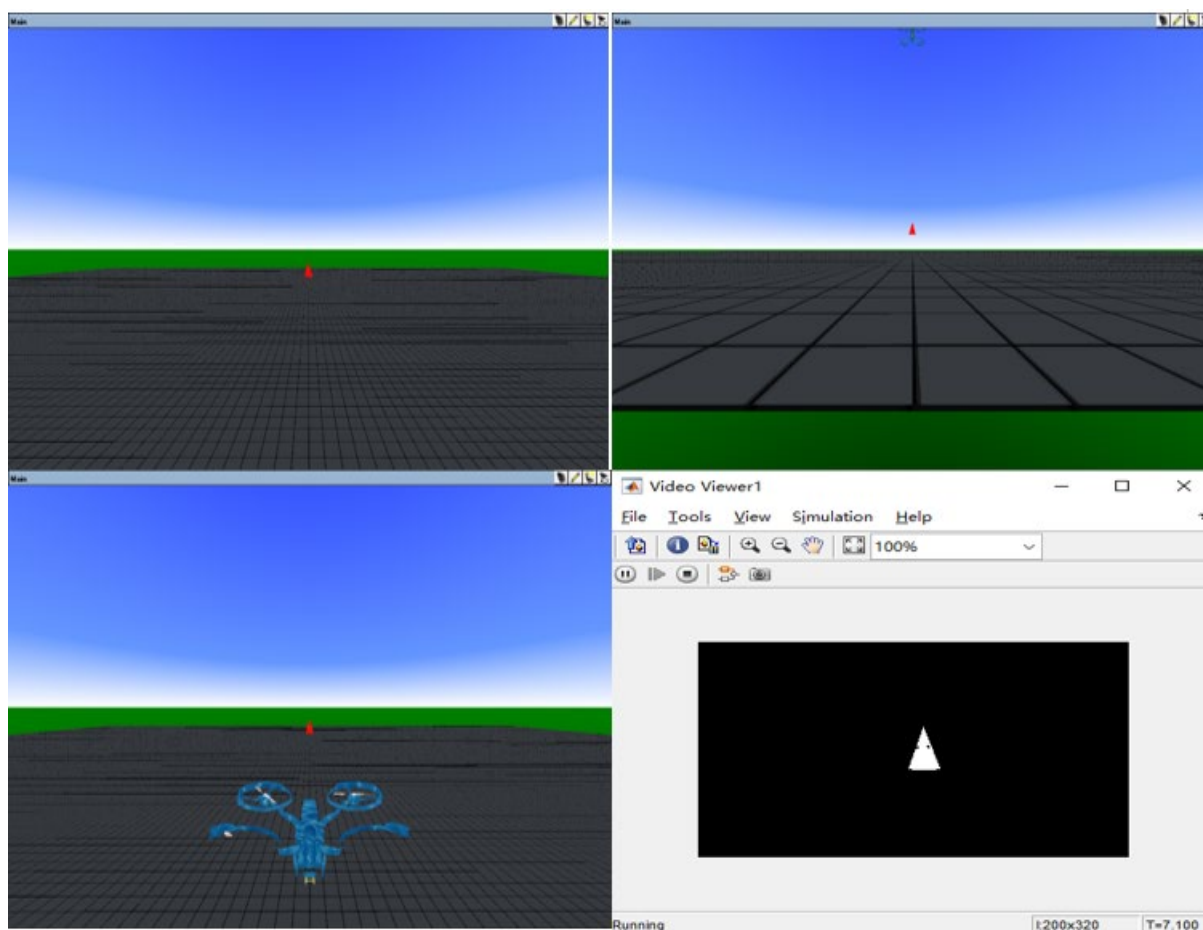


图 5.1 仿真环境视角图

在 V-realm Builder 中的多旋翼模型只是具有位姿、桨转速等输入参数的显示模型，并不具有物理引擎支撑，所以需要在 Simulink 中 VR 模块的外部搭建多旋翼飞行器模型，获取多旋翼的位姿输入和桨转速等数据参数。下文将具体介绍多旋翼飞行器和 Animation 3D 中的 VR 模块搭建连接情况。

## 5.2 综合控制系统框架

整个视觉伺服仿真系统主要分为四大部分,分别为 Simulink 多旋翼模型、VR 模型、视觉跟踪算法模块和视觉伺服控制模块,由这五个部分经过一些坐标变换等辅助运算就构成了一个完整的仿真系统。首先介绍一些各模块的搭建情况。

### 5.2.1 子模块构成

#### 1、多旋翼模块

多旋翼模块是本文实验中的控制对象,控制算法中的输入角速度和力可以根据多旋翼模型反解出多旋翼的期望速度,本文使用的多旋翼模型也正是使用期望速度  $v_x, v_y, v_z$  和期望偏航角速度  $\omega_z$  作为输入控制量进行控制的。多旋翼模型的输出为 12 维列向量,分别输出的是多旋翼的世界坐标系下坐标、速度、滚转角、俯仰角、偏航角和滚转角速度、俯仰角速度和偏航角速度,可以供之后的控制和 VR 显示模块使用。下面将具体地介绍多旋翼模型内部结构。

模块的 Simulink 模型如图 5.2 所示,由外部的控制部分 (Control System) 和多旋翼的运动模型 (Quadcopter Dynamic) 组成,本文实现对多旋翼控制主要研究对控制系统的改写和使用。

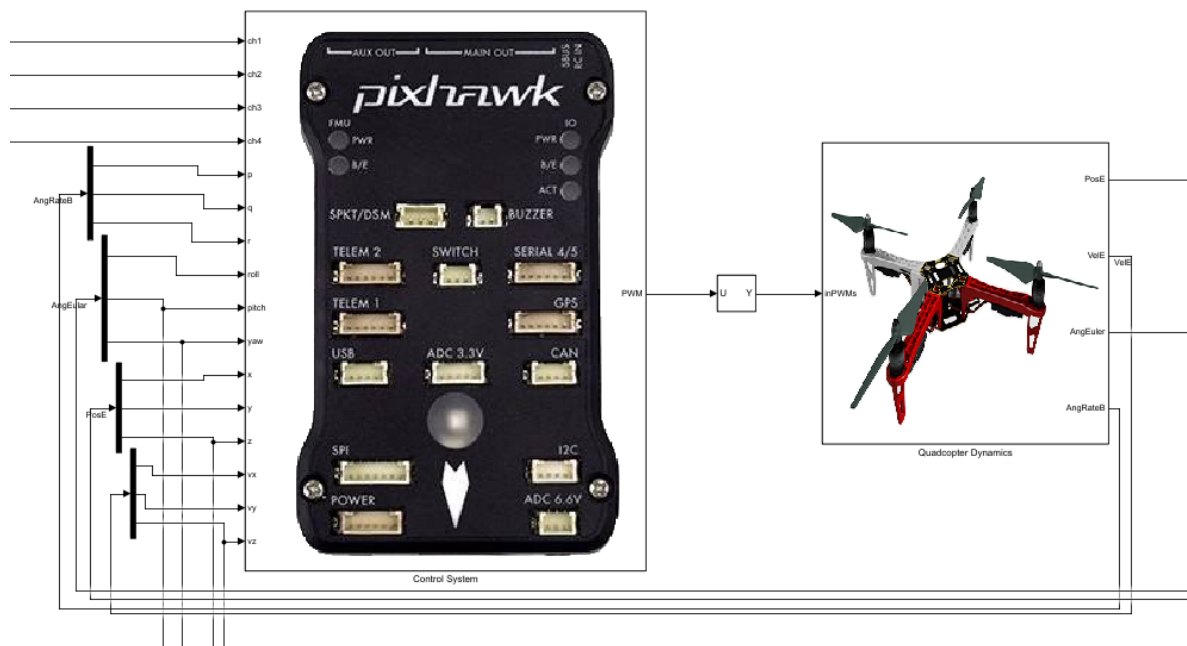


图 5.2 多旋翼 Simulink 模型

多旋翼控制系统中输入量为三个世界坐标系下的速度和偏航角速度，本实验模型为了贴合使用遥控器控制，控制量输入量级接近信道信号输入，并且为了限制输入信号要求不超过多旋翼自身物理限制，对输入信号添加饱和进行限幅。此外，为了使模型接近实际使用，还对输入量添加了死区，对过小的输入信号不响应。

输入量  $u = [v_{xd}, v_{yd}, v_{zd}, \omega_{zd}]^T$  是对多旋翼位姿的期望值，而多旋翼本身会输出速度和角速度的实际值，速度上用期望值减去实际值的偏差乘上一个比例系数就是多旋翼的期望加速度。在第二章的多旋翼数学模型建立中，已经建立一些假设，假设多旋翼飞行特点是滚转角和俯仰角都非常小，总拉力约等于多旋翼的重力。输入  $z$  方向期望加速度，根据第二章给出的多旋翼高度通道模型，可以得到期望拉力；输入多旋翼偏航角  $\psi$ ，期望加速度  $a_x, a_y$  可以根据多旋翼模型得到期望俯仰角和滚转角

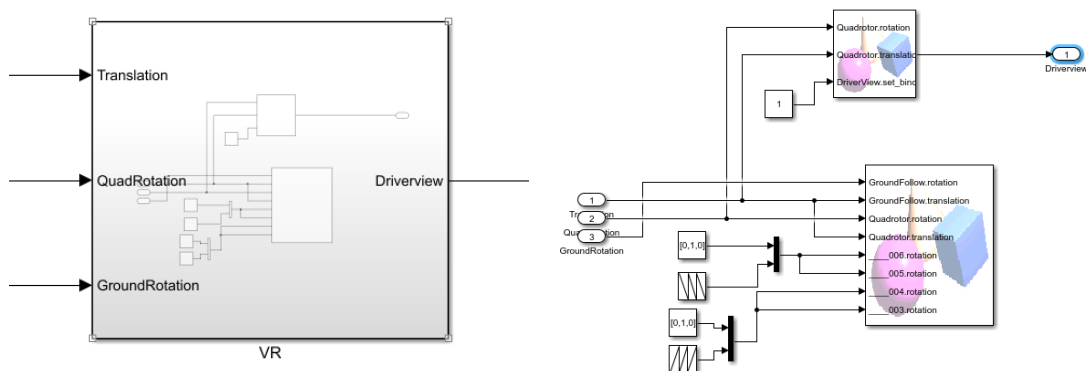
$$\begin{aligned}\phi &= (-a_x \sin(\psi) + a_y \cos(\psi)) / g \\ \theta &= (-a_x \cos(\psi) - a_y \sin(\psi)) / g\end{aligned}\tag{5.1}$$

通过期望俯仰角  $\theta$  和期望滚转角  $\phi$  与实际输出值做差，乘以比例系数后可以得到期望滚转角速度和期望俯仰角速度，因为本模型中输入还有期望偏航角速度，所以可以将这三个期望角速度直接减去对应的实际角速度  $\omega_x, \omega_y, \omega_z$ ，因为力矩与角速度成正比，所以得到三个角方向的力矩。

根据三个角方向的力矩和计算出的期望拉力，可以对多旋翼四个桨进行转速分配，分配的结果即整个多旋翼控制系统的最终输出，结合多旋翼的动力模型就可以得到在四桨转速如此分配时的 12 维输出向量。至此，多旋翼控制系统构建完成。

## 2、VR 模块

VR 模块本身不具有物理引擎，主要作用有两种，一种是根据期望坐标、姿态输入在虚拟场景中显示仿真过程，可视化验证算法；第二种是通过设置视角获取摄像机视域图像。本实验中 VR 模块分为两块，一块输入目标期望的移动方式，做多种目标移动方式的仿真；另一块连接多旋翼模块，显示仿真过程并获取摄像头图像。主要部分是多旋翼的显示仿真，为了显示仿真过程，如图 5.3 左所示，需要输入多旋翼平移和轴角旋转向量，输出的是固连在多旋翼前方的摄像机获得的 DriverView 图像，作为下一步图像跟踪算法的输入图像。



为了在 Simulink 中实现视景仿真，选用 Simulink 中虚拟现实工具箱 Simulink 3D Animation 中的 VR Sink 和 VR To Video 模块。它们可以很方便地将在 V-Realm Builder 中构建的虚拟场景导入，并且导入也可以直接在 Editor 中添加、编辑、删除场景的所有信息，可以高效地及时修改虚拟场景。使用 VR Sink 和 VR To Video 需要

在图 5.3 右中展示了 VR 模块的具体搭建方式，为了输入多旋翼输出的平移坐标和旋转轴角矩阵，选择 VR Sink 模块作为 VR 输入，并且将桨的转速参数分别接入桨转速的输入。选用 VR To Video 模块，同样接入多旋翼的平移坐标和旋转轴角矩阵，因为基于视觉的目标追踪算法需要的是多旋翼摄像头第一视角图像，并且摄像头固连在多旋翼机架前方，所以在 VR Builder 中设置 DriverView 为多旋翼下子类，并且位于机架正前方，以达到视角要求。为了视角保持 DriverView 不跳变到其他观察视角，外接置 1 信号到 DriverView 的设置开关，因此 VR 模块的输出就可以保证是固连在多旋翼机架前方的摄像头第一视角图像，达到基于视觉的目标追踪需求。

### 3、基于视觉的目标追踪算法模块

基于视觉的目标追踪算法模块输入 VR 模块中输出的 DriverView 图像, 经过选取算法追踪目标, 得到其在图像中坐标位置, 减去中心坐标, 就得到了目标与图像中心偏差, 作为视觉伺服模块的输入, 控制目标在图像中向中心收敛。模块输入输出如图 5.4 所示

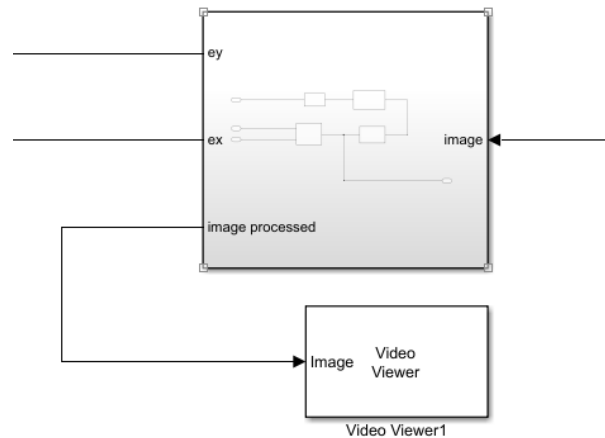


图 5.4 基于视觉的目标追踪算法输入输出图

#### 4、视觉伺服算法模块

视觉伺服算法模块是本次仿真的检测目标所在，根据在第二章中建立的视觉伺服模型、多旋翼通道模型和第四章中设计的纵向和横侧向通道视觉伺服算法，编写控制器程序。前文介绍多旋翼模块时提及多旋翼输入控制向量为  $u = [v_{xd}, v_{yd}, v_{zd}, \omega_{zd}]^T$ ，而视觉伺服算法设计输出是  $u' = [\omega_{xd}, \omega_{yd}, \omega_{zd}, f]$ ，可以通过多旋翼通道模型反解出多旋翼控制向量，从而实现多旋翼控制。模块输入输出如图 5.5 所示

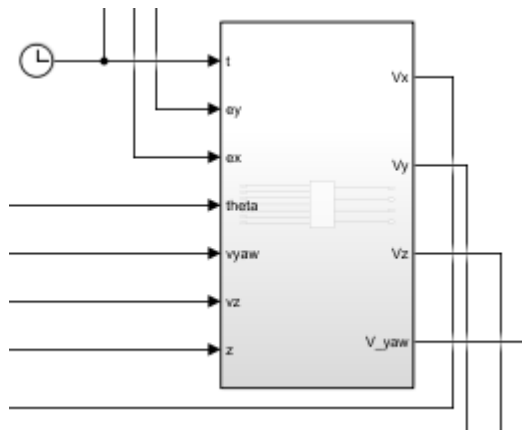


图 5.5 视觉伺服算法输入输出图

#### 5.2.2 仿真框架

首先说明仿真中四部分模块的连接方式，如图 5.6。多旋翼模块输出三维位置与姿态信息，经过相应的坐标转换后得到视景坐标系下多旋翼的平移与旋转轴角向量；VR 模块中通过固连在机架正前方的视角模拟摄像头获取多旋翼在运动中的 RGB 图像，经过

VR To Video 输出到基于视觉的目标追踪模块，经过基于视觉的目标追踪算法后获取每一帧目标在图像中坐标位置，从而得到目标与中心点偏差  $e_x, e_y$ ；然后根据 4.2 和 4.3 中推导的纵向和横侧向通道视觉伺服算法，由图像偏差  $e_x, e_y$  得到多旋翼的控制向量，包含拉力、俯仰角速度，偏航角速度和滚转角速度，经过坐标变换输入到多旋翼，引导多旋翼正确打击目标，完成一个控制闭环。

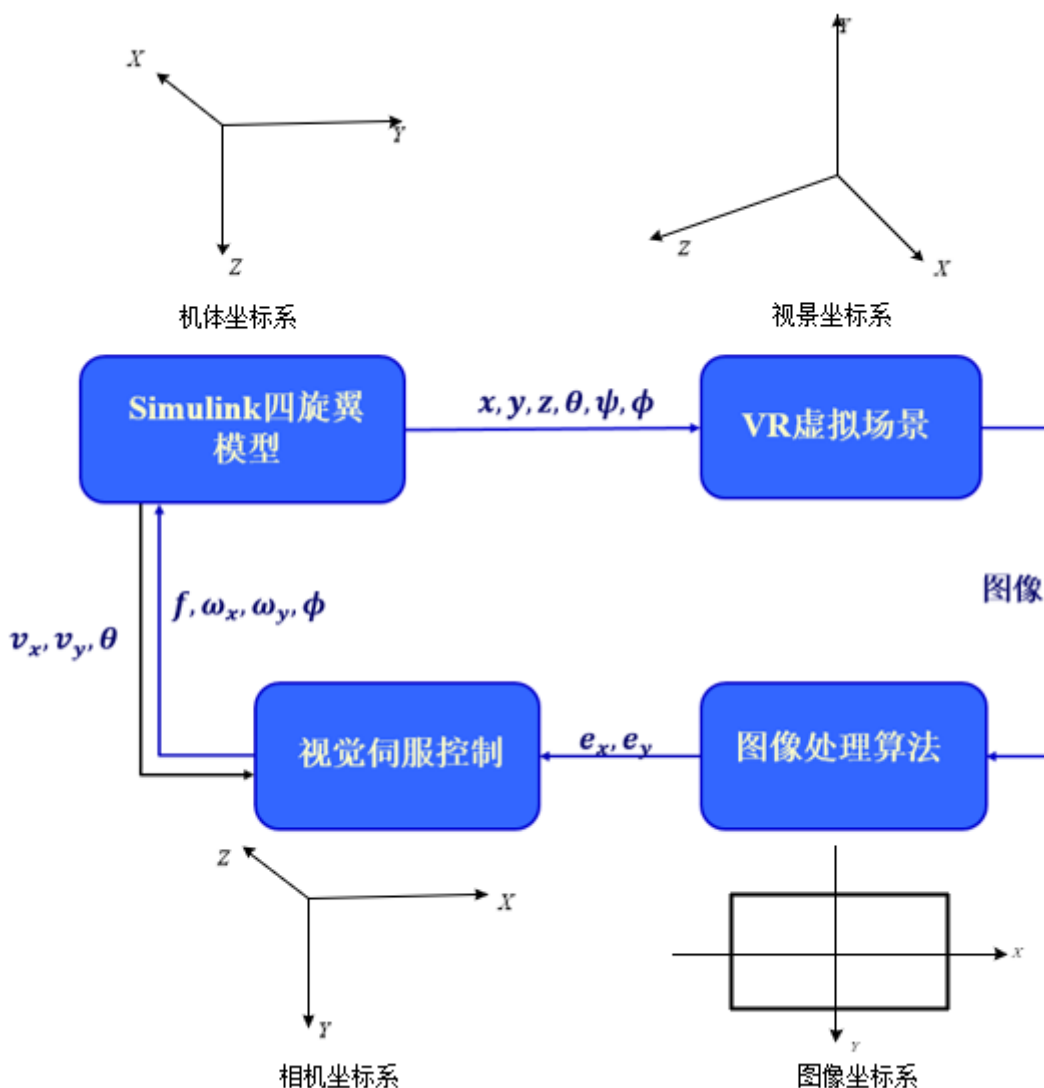


图 5.6 仿真系统框图和坐标系变换

仿真过程中每个模块都有不同的坐标系，所以连接模块的输入输出时进行了多次坐标变换，如图 5.6 所示，多旋翼模型中使用 NED 机体坐标系，（x 轴指北，y 轴指东，z 轴指地）输出的多旋翼位姿信息需要变换到 VR 模块的视景坐标系；在第四章设计过程

中提到视觉伺服算法本身使用的是相机坐标系，因为相机固连在机架上，所以近似地可以认为相机的运动状态和多旋翼飞行器的运动状态是一致的，所以运动位姿信息大小不变，只需要从相机坐标系变换到机体坐标系。

根据以上的仿真模块、仿真框架和模块间坐标系转换连接，在 MATLAB/Simulink 中搭建仿真整体，最终如图 5.7 所示

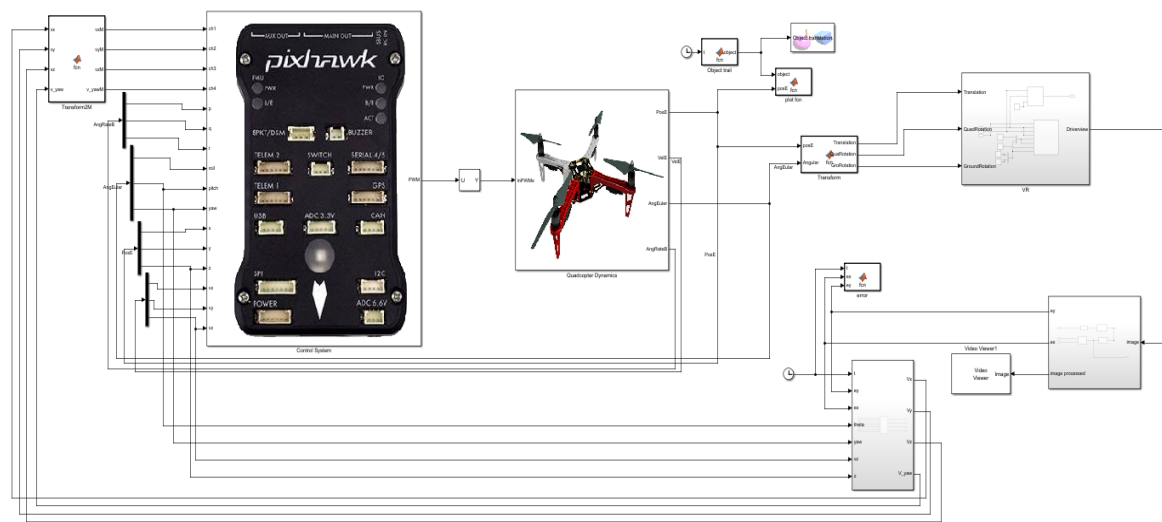


图 5.7 视觉伺服仿真系统 Simulink 模型

### 5.3 仿真测试和性能分析

本文的控制器性能分析部分旨在通过改变目标物体运动轨迹、目标运动速度、物体运动速度等，观察视觉伺服控制多旋翼飞行器伺服轨迹，比较图像坐标系下与中心点偏差  $e_x, e_y$  大小，计算  $e_x, e_y$  偏差的时间均值和标准差，分析多旋翼速度、目标速度和目标轨迹形状对伺服控制器效果的影响。下文将就这几个方面进行多维度的展开讨论。

#### (1) 目标摇摆式上升

令目标横侧向坐标成  $2 * \sin(2t)$  运动，纵向匀速  $2\text{m/s}$  上升，多旋翼处于较高速运动中，速度为  $6\text{m/s}$ ，目标与多旋翼飞行器在地面的投影距离为  $15\text{m}$ ，多旋翼视域内初始时目标处于一个与中心较大偏差的位置，具有一个阶跃的初始输入。



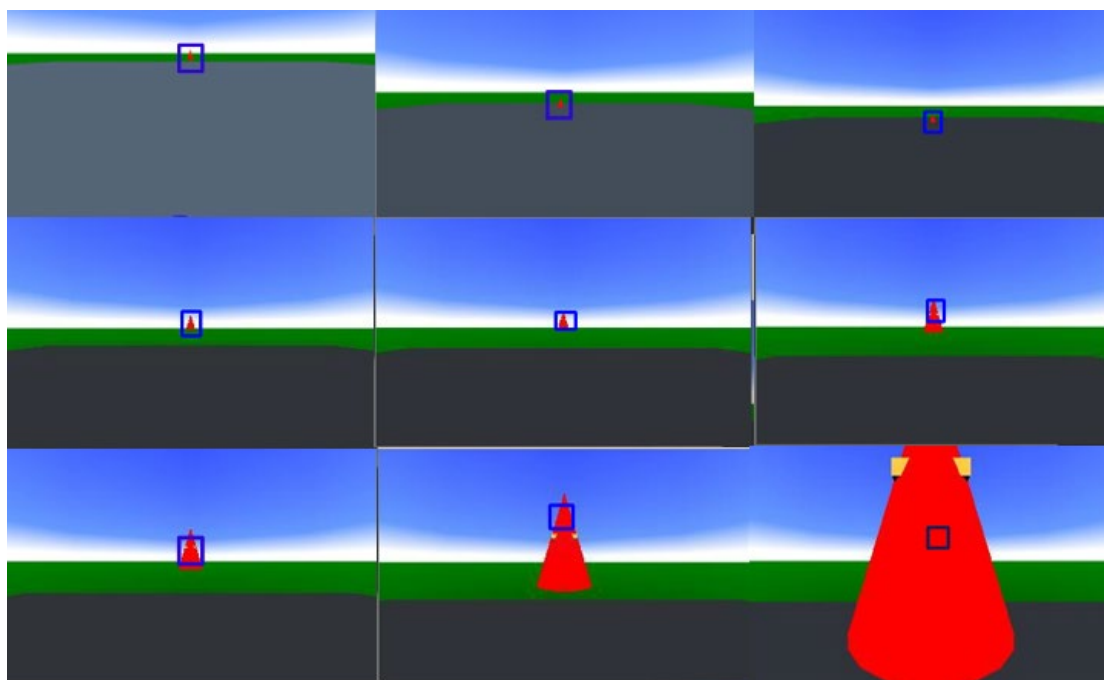


图 5.8 DriverView 视景仿真目标追踪

在视景仿真中，固连在机架前方的摄像头获取的 DriverView 角度图片如图 5.9 所示。用矩形框在第一帧框取目标，随后基于视觉的目标追踪算法标记出每一帧目标的位置，将框中心视为目标中心，计算图像像素坐标系下目标距离图像中心的偏差，作为视觉伺服算法的输入量。

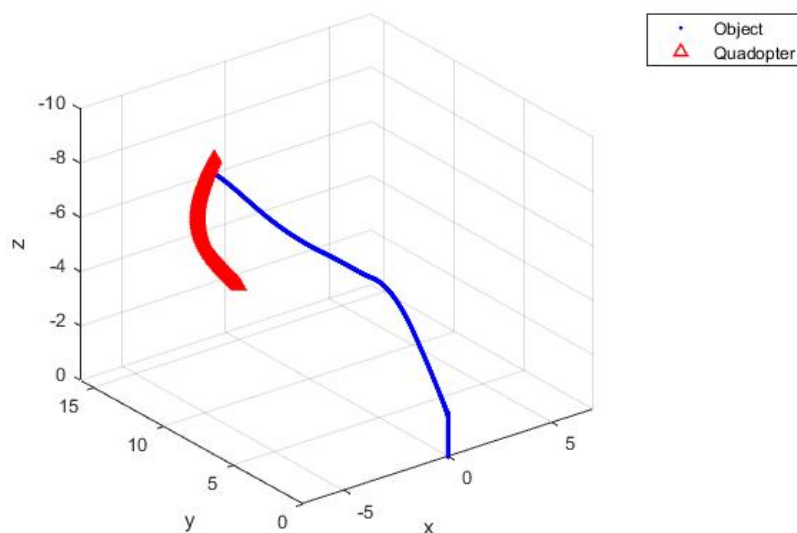


图 5.9 目标摇摆式上升多旋翼视觉伺服打击轨迹

因为 DriverView 视角的视频不易在论文中展示，绘制多旋翼视觉伺服控制打击目标



的轨迹曲线,如图 5.8 所示,红色是目标轨迹,蓝色是多旋翼轨迹,可见多旋翼完成了打击任务,最终与目标相遇。曲线中最初的一段平稳上升是没有伺服控制的,目的是多旋翼起飞并保持平稳,到达伺服控制起点;然后开始伺服控制时,根据第四章中公式推导,因为多旋翼飞行器需要达到目标俯仰角 $\theta_d$ ,所以竖直方向上初始时以俯仰角速度控制为主,之后以高度方向速度控制为主。为了评价本次多旋翼伺服控制打击的控制效果,绘制图像坐标系下目标坐标与中心偏差的时域图,如图 5.10 所示

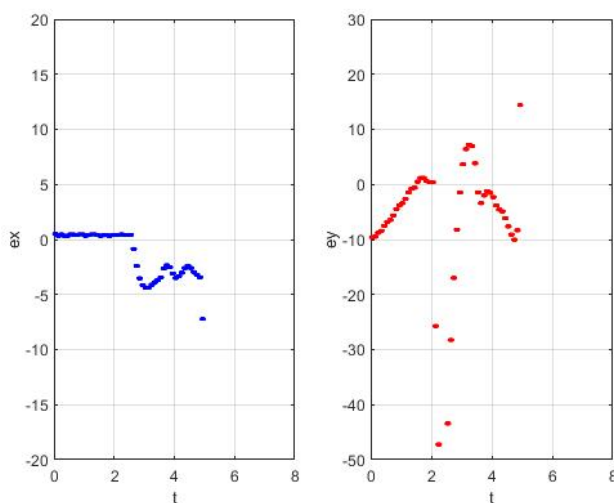


图 5.10  $e_x, e_y$  时域图

图像中在第 2.2 秒之前是多旋翼上升和加速的过程,所以可以看到 $e_x, e_y$ 偏差增大。之后开始视觉伺服控制,具有非常明显的向中心收敛的趋势,并在中心附近震荡。最终并没有收敛到零位,而是收敛到零位附近有两方面原因,一是目标在不断运动,目标点坐标持续变化,每帧之间具有偏差;二是为了更好仿真实场景中的多旋翼模型,加入了控制死区,对于非常小的输入没有控制响应,所以理论上也不可能收敛到零位,只能在附近震荡。

对图像偏差进行定量计算,分别计算 x 方向图像偏差的均值和标准差, y 方向图像偏差的均值和标准差,单位是像素。多旋翼速度 6m/s 不变,多旋翼与目标在地面投影距离 15m 不变,改变目标的水平速度大小,调节目标实际速度,进行多次测量和计算,仿真测量结果如表格 5.1 所示:



表格 5.1 仿真图像坐标偏差结果

目标最大速度(m/s)	2	3	4
$e_x$ 平均值(px)	-3.0523	-1.8102	1.4920
$e_x$ 视域占比 (%)	0.94	0.56	0.47
$e_y$ 平均值(px)	-1.4504	-9.2546	-1.1091
$e_y$ 视域占比 (%)	0.72	4.62	0.55
$e_x$ 标准差(px)	1.8528	2.2902	5.3570
$e_y$ 标准差(px)	5.8346	6.2965	6.5727

可以根据  $e_x, e_y$  平均值占视域的比例可以评价视觉伺服算法控制效果。表格中  $e_x, e_y$  单位是像素 (pixel)，可以根据视域被像素宽度计算目标偏差所占视域比例，在 x 方向上视域内像素宽度 320px，在 y 方向上视域内像素宽度 200px。

经过表格对比，发现目标在这种运动方式下，当目标速度增大时， $e_x, e_y$  的绝对值和所占视域比例并没有相关特征，但是  $e_x, e_y$  的标准差随着目标速度增大逐渐增大，也就是说当目标速度增大时，目标仍在图像中心附近震荡，但是离散程度增大。

(2) 目标左斜上升

令目标横侧向坐标成  $2t$  运动，纵向匀速 2m/s 上升，多旋翼速度为 4m/s，目标与多旋翼飞行器在地面的投影距离为 15m，多旋翼视域内初始时目标处于一个与中心较大偏差的位置，具有一个阶跃初始输入。

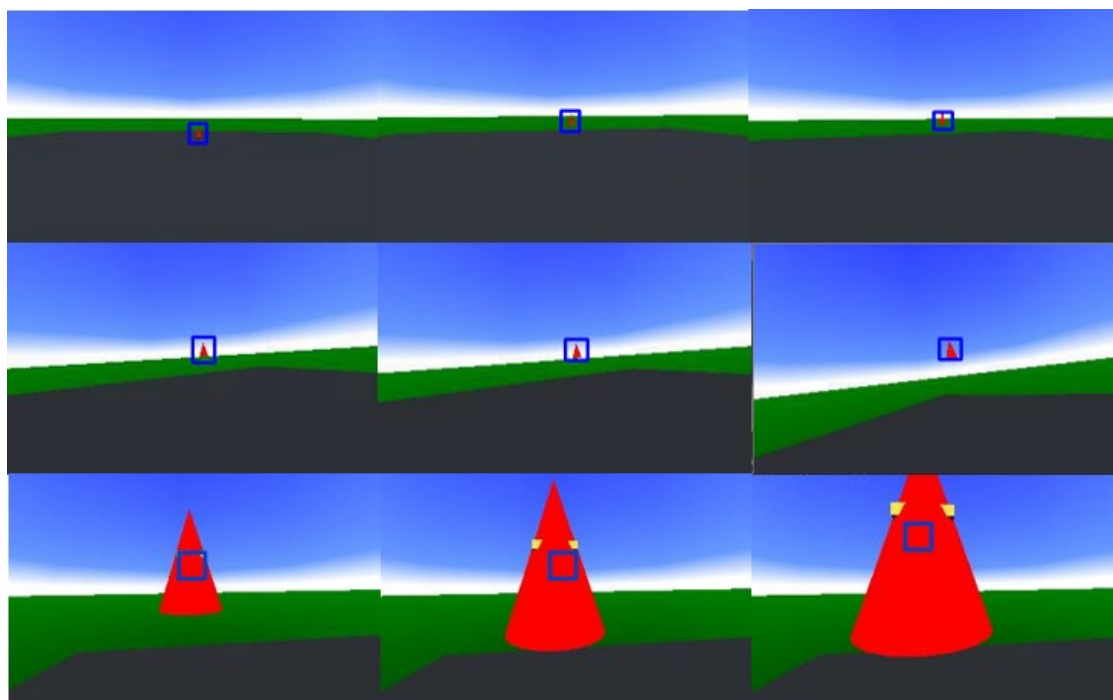


图 5.11 DriverView 视景仿真目标追踪

与之前运动方式相似地，在视景仿真中，固连在机架前方的摄像头获取的 DriverView 角度图片如图 5.11 所示。用矩形框在第一帧框取目标，随后基于视觉的目标追踪算法标记出每一帧目标的所在位置，将框中心视为目标中心，计算图像像素坐标系下目标距离图像中心的偏差，作为视觉伺服算法的输入量。

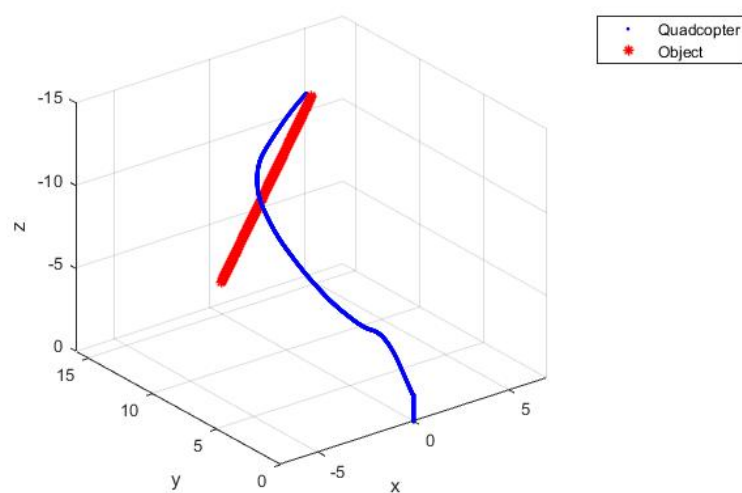


图 5.12 目标左斜上升多旋翼视觉伺服打击轨迹

因为 DriverView 视角的视频不易在论文中展示，绘制多旋翼视觉伺服控制打击目标

的轨迹曲线,如图 5.12 所示,红色是目标轨迹,蓝色是多旋翼轨迹,可见多旋翼完成了打击任务,最终与目标相遇。曲线中最初的一段平稳上升是没有伺服控制的,目的是多旋翼起飞并保持平稳,到达伺服控制起点。为了评价本次多旋翼伺服控制打击的控制效果,绘制图像坐标系下目标坐标与中心偏差的时域图,如图 5.13 所示

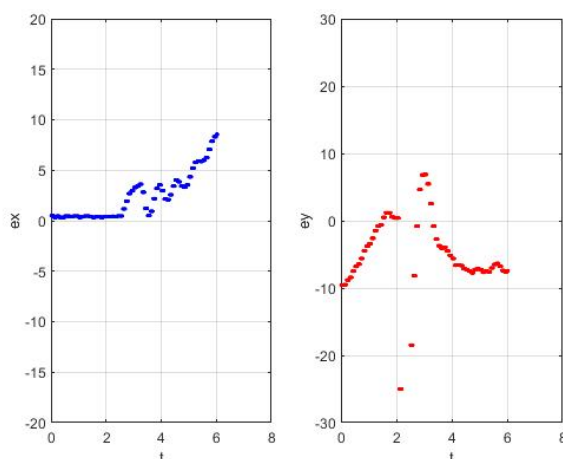


图 5.13  $e_x, e_y$  时域图

同样的,图像中在第 2.2 秒之前是多旋翼上升和加速的过程,所以可以看到  $e_x, e_y$  偏差增大。之后开始视觉伺服控制,具有非常明显的向中心收敛的趋势,并在中心附近震荡。最终并没有收敛到零位,而是收敛到零位附近有两方面原因,一是目标在不断运动,目标点坐标持续变化,每帧之间具有偏差;二是为了更好仿真实场景中的多旋翼模型,加入了控制死区,对于非常小的输入没有控制响应,所以理论上也不可能收敛到零位,只能在附近震荡。

对图像偏差进行定量计算,分别计算 x 方向图像偏差的均值和标准差, y 方向图像偏差的均值和标准差,单位是像素。多旋翼速度 4m/s 不变,多旋翼与目标在地面投影距离 15m 不变,改变目标的实际速度大小,进行多次测量和计算,仿真测量结果如表格 5.2 所示:



表格 5.2 仿真图像坐标偏差结果

目标速度(m/s)	2.3	2.8	3.5
$e_x$ 平均值(px)	4.0611	4.1666	4.1081
$e_x$ 视域占比 (%)	1.3	1.3	1.3
$e_y$ 平均值(px)	-4.7605	-5.1571	-8.6771
$e_y$ 视域占比 (%)	2.3	2.6	4.3
$e_x$ 标准差(px)	2.0321	1.5803	5.3570
$e_y$ 标准差(px)	2.6518	5.3646	6.5727

可以根据  $e_x, e_y$  平均值占视域的比例可以评价视觉伺服算法控制效果。表格中  $e_x, e_y$  单位是像素 (pixel)，可以根据视域被像素宽度计算目标偏差所占视域比例，在 x 方向上视域内像素宽度 320px，在 y 方向上视域内像素宽度 200px。

经过表格对比，发现目标在这种运动方式下，当目标速度增大时，  $e_x, e_y$  的绝对值和所占视域呢比例大致呈增大趋势，并且  $e_x, e_y$  的标准差随着目标速度增大逐渐增大，也就是说当目标速度增大时，目标仍在图像中心附近震荡，但是偏离程度和离散程度增大。

5.4 本章小结

本章是对前几章理论推导的仿真实验验证部分，并对视觉伺服算法在不同参数情况下的控制效果进行评价，最后得出了一些定性的结论。在 5.1 节中介绍了仿真环境搭建过程、三维建模工具 V-Realm Builder。5.2 节主要介绍了构成仿真系统的四个主要模块和仿真框架，并将每个模块的坐标系和连接每个模块的坐标变换进行了详细绘图说明。5.3 节是对控制效果的测试评价环节，通过令目标进行两种不同轨迹的运动，改变目标速度和多旋翼运动速度，比较控制过程图像偏差平均值和标准差，得出定性的结论。



## 总结和展望

### 总结

多旋翼飞行器因其体积小、重量轻、造价低,具有良好的机动性能和操纵性能而才成为一种被广泛使用的现代工具,在搜索、航拍、监视、军事等领域表现出出色的特征优势。现代战争中已经开始将多旋翼投入到狭窄空间和街头楼巷的作战任务中,但是主流作战方式需要经验丰富的操纵手控制或者使用 GPS 实现多旋翼对目标的精确打击。这两种方法各自有其弊端,第一种有高风险性、低易用性,第二种因为信号问题地域局限很大,所以本文希望提出一种可靠的基于图像的伺服控制方法,能够使多旋翼自主飞行打击目标。

本文的主要目标是设计一种基于图像的伺服控制方法,可以在近距离自主打击目标并且打击过程中目标收敛在图像视域内。视觉伺服系统全过程为首先从固连在机架前方的摄像头获取实时的图像信息,利用基于视觉的目标追踪算法获取每一帧中目标在图像中的坐标,利用视觉伺服算法得到多旋翼控制向量,从而改变多旋翼位姿,也就改变了摄像头位姿,从而完成一个控制闭环。

本文在 MATLAB 上利用虚拟现实工具箱对视觉伺服算法进行视景仿真验证。采用 V-Realm Builder 建立三维场景模型,在 Simulink 中搭建完整视觉伺服系统所需的多旋翼模型、VR 视景、基于视觉的目标追踪模块和视觉伺服控制器。仿真结果表明,本文设计的方法可以实现多旋翼飞行器的基于图像的近距离追踪打击控制,最后探讨了目标不同运动轨迹、不同运动速度对控制效果的影响。

### 展望

本文完成了算法的视景仿真,下一步工作还可以从如下方向进行:

(1) 本文的所有仿真模型和控制器都是在 Simulink 平台上完成的,可以将控制器和基于视觉的目标追踪算法迁移到 Pixhawk 上进行半物理仿真,进一步验证算法。

(2) 本算法研究的是多旋翼与目标距离不过长也不过短,即图像中必须能够识别出目标这段区间的控制算法。距离过远时可以使用 GPS 导航到目标附近再使用视觉伺服,对距离过短的情况可以使用运动预测,还需要进一步探讨。



## 致谢

半年弹指一挥间，毕设选题开题仿佛才过去不久，今天我已经写完毕业论文，距离正式毕业，离开实验室还有不过几天时间了。在这半年中，我充分体验到实验室大家庭中互帮互助温暖，体会到科研中做一件小事的不易。也感谢全老师、郭正龙师兄、杨坤师兄和邓恒师兄在这半年时间里对我的教诲和帮助，因为老师和师兄们的帮助，我在毕设期间得以克服重重困难，完成毕设任务。

在完成毕业设计过程中，全老师和实验室的师兄师姐们都在我心中留下了深刻鲜明的精神形象。全老师治学严谨，学识广博，对学生的学业十分认真负责，时常给与我思路上的导引，让我明白怎样用正确的思维方式分析科研问题；杨坤师兄、邓恒师兄、正龙师兄我接触得最久，在学习态度，学习方法上都给我很大启发，和这些优秀的师兄们接触，我认识到我自己如此多方面的不足，给我自己以激励；在实验室中我深受互帮互助氛围的濡染，高岩学长、王帅学长这些我并不太熟悉的学长，在我有问题不能理解时，都非常热情地给我指点，让我励志以后也要做一个乐于分享知识的人。

在实验室中我不仅对毕设中的知识、应用到的工具有了自己的理解，也从全老师和师兄师姐们身上学到了以后解决问题的思路和做事做人的品德，在此再次感谢全老师和实验室的所有师兄师姐，愿以后情意长存。



## 参考文献

- [1] P. Doherty, P. Rudol. A UAV search and rescue scenario with human body detection and geolocalization [C]. Proceedings of the 20th Australian Joint Conference on Advances in Artificial Intelligence. Berlin: Springer-Verlag, 2007: 1-13.
- [2] 霖深笙.叙利亚政府军开始研制干扰器 只为应对战场上的“大疆”无人机[EB/OL]. [http://www.elecfans.com/application/Military\\_avionics/2017/0705/530088.html](http://www.elecfans.com/application/Military_avionics/2017/0705/530088.html), 2017-07-05.
- [3] S. N. Zenk, A. J. Schulz, S. A. Matthews, et al, Activity space environment and dietary and physical activity behaviors: a pilot study[J]. Health &Place, 2011, 17(5): 1150-1161.
- [4] 陈旭潮, 曹志强, 于莹莹, 周超.基于视觉伺服的四旋翼飞行器悬停控制[J].华中科技大学学报(自然科学版), 2015, 43(S1):6-9.
- [5] T. Hamel and R. Mahony. Visual servoing of an under-actuated dynamic rigid-body system: an image-based approach[C]. Robotics and Automation, IEEE Transactions on, 18(2):187-198, Apr 2002.
- [6] S. Saripalli, J.F. Montgomery, and G. Sukhatme. Vision-based autonomous landing of an unmanned aerial vehicle[C]. Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on, volume 3, pages 2799-2804, 2002.
- [7] E. Frew, T. McGee, Z. W. Kim, X. Xiao, S. Jackson, M. Morimoto, S. Rathinam, J. Padial, and R. Sengupta. Vision-based road-following using a small autonomous aircraft[C]. Aerospace Conference, 2004. Proceedings. 2004 IEEE, volume 5, pages 3006-3015 Vol.5, March 2004.
- [8] D.A. Ross, J. Lim, R. Lin, M. Yang. Incremental learning for robust visual tracking[J]. Int. J. Comput. Vis. 77 (1-3) (2008) 125-141.
- [9] B. Babenko, M. Yang, S. J. Belongie. Robust object tracking with online multiple instance learning[C]. IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1619-1632.
- [10] C. Bao, Y. Wu, H. Ling, H. Ji. Real time robust L1 tracker using accelerated proximal gradient approach[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1830-1837.





- [11] Y. Wu, J. Lim, M. Yang. Online object tracking: a benchmark[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2013, pp. 2411-2418.
- [12] X. Jia, H. Lu, M. Yang. Visual tracking via coarse and fine structural local sparse appearance models[C]. IEEE Trans. Image Process. 25 (10) (2016) 4555-4564.
- [13] H. Nam, B. Han. Learning multi-domain convolutional neural networks for visual tracking[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 4293-4302.
- [14] L. Wang, W. Ouyang, X. Wang, H. Lu. STCT: sequentially training convolutional networks for visual tracking[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1373-1381.
- [15] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, P.H.S. Torr. Fully-convolutional Siamese networks for object tracking[C]. Proceedings of the European Conference on Computer Vision Workshops, 2016, pp. 850-865.
- [16] F. Chaumette, S. Hutchinson. Visual servo control. I. basic approaches[J]. Robotics & Automation Magazine, 2006, 13(4): 82-90.
- [17] 吕强, 马建业, 王国胜, 林辉灿, 梁冰. 基于视觉伺服的小型四旋翼无人机自主飞行控制研究进展[J]. 科技导报, 2016, 34(24): 68-73.
- [18] M. Xin, S. Balakrishnan, E. Ohlmeyer. Guidance law design for missiles with Reduced seeker field-of-view[C]. AIAA Guidance, Navigation, and Control Conference and Exhibit. 2006: 6085.
- [19] D. Sang, C. K. Ryoo, K. R. Song, et al. A guidance law with a switching logic for maintaining seeker's lock-on for stationary targets[C]. AIAA Guidance, Navigation and Control Conference and Exhibit. 2008: 6497.
- [20] B. G. Park, T. H. Kim, M. J. Tahk. Optimal impact angle control guidance law considering the seeker's field-of-view limits[J]. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 2013, 227(8): 1347-1364.
- [21] 全权著. 多旋翼飞行器设计与控制[M]. 杜光勋, 赵峙尧, 戴训华, 任锦瑞, 邓恒译. 电子工业出版社: 北京, 2018:227.



- 
- [22] F. Chaumette, S. Hutchinson. Visual servo control. I. basic approaches[J]. Robotics & Automation Magazine, 2006, 13(4): 82-90.
- [23] P. Li, D. Wang, L. Wang, and H. Lu. Deep visual tracking: Review and experimental comparison[J]. Pattern Recognition, 76:323-338, 2018.