

摘要

在自然界中，广泛存在着各种大规模的壮观的群体运动。这些现象吸引了多领域的科学家们，他们提出了不同的理论模型来解释和模拟这些大规模群体行为。

近年来，高速高清数码相机的发展和计算机性能的大幅提升，使三维运动轨迹的跟踪和测量成为可能，但大规模群体运动的跟踪仍然极具挑战性，主要因为群体数量庞大，往往有成百上千只个体，数据量非常大、这些个体彼此相似，无法依靠外形特征分辨、群体的数量及密度很大，个体间的遮蔽问题频繁出现，使得正确、准确地跟踪非常困难。

基于上述困难，本文提出了大规模果蝇群体的三维跟踪算法。在目标检测时，本文采用减去背景图像的方法分割出果蝇个体，然后计算果蝇的二维图像坐标。在目标跟踪、立体匹配、三维轨迹重连接步骤中，分别将这三个问题建模为最大后验概率问题，然后分别转化为三个线性分配问题，通过建立二分图模型，使用二分图最佳匹配算法求解。

在果蝇群体实测中，对 200 余只果蝇进行了同步拍摄，本文选取了其中 200 帧图像进行果蝇轨迹跟踪，最后得到了 794 条完整的三维轨迹。可以进一步用于果蝇的群体行为分析。因此，本文的跟踪算法可以自动、准确、高效地进行大规模果蝇群体三维运动跟踪。

关键词： 计算机视觉, 动物群体行为, 三维运动测量, 运动轨迹重构

Abstract

Collective motion of large animal groups is one of the most common but spectacular scenes in the nature, which has attracted great attention of scientists from many research areas, and they have developed different theoretical models to explain and simulate these large scale collective motion.

With the rapid development of high-speed camera and high-performance computers in recent years, it is now possible to track and measure the 3D trajectories of animals. But, it is still a very challenging work to track large scale of animal groups due to: Large scale of collectively moving animals usually involve thousands of individuals, the data set will be too large to process. And the individuals have similar features and thus it is impossible to distinguish them using their appearance features. What's more, the group is often very dense, and therefore, occlusions of the objects themselves will be very frequent in image sequence, which makes tracking each individual correctly a very difficult work.

To solve the above problems, a 3D tracking algorithm was developed in this paper. In object detection step, a background subtraction method was used to segment the image, then, the 2D image coordinates of the drosophila were calculated. In object tracking, stereo matching and 3D trajectory reconnection steps, the three problem were modeled as three maximum a posterior problems, and then, these three problems were exchanged to three linear assignment problems. Then, three bipartite graph models were built, the three linear assignment problems can be efficiently solved using bipartite graph best matching algorithm(Kuhn-Munkras Algorithm).

In real experiment, a flying group of more than 200 drosophila were captured synchronously. 200 frames of photos were selected to track the trajectories of the drosophila. At last 794 complete trajectories were achieved, which can be used to analyze the collective motion of drosophila. In summary, the 3D tracking algorithm developed in this paper can track large scale of drosophila group flying in 3D space automatically and correctly.

Keywords: Computer Vision, Collective Motion, 3D Motion Measurement, Trajectory Reconstruction

目 录

1 绪论

1.1 研究背景

人类早在两千年前就开始记录、研究动物群体的运动行为。在此之后的两千多年中，多领域、多学科的科学家们都在尝试探索群体运动背后的规律。但受限于当时的科学技术条件，所以并不能对动物群体行为进行准确地定性分析。对群体运动的较为准确地定量研究始于上世纪六七十年代，且到目前为止已有多种不同的研究方法。记录群体运动的三维空间坐标也有多种方式，可以使用 GPS 全球定位系统、也可以使用多相机定标后进行摄影测量，然后进行三维重构。

1.1.1 三维运动轨迹测量

世界是物质的，物质是运动的，运动是有规律的。掌握物质运动的规律对于认识自然和改造自然具有重要意义，而测量运动是研究运动规律所不可或缺的关键技术。大规模群体的三维运动是科学研究与工程技术中经常遇到的对象。如图??所示，我们常看到的大规模动物群体运动有：在天空中飞翔的鸟群^[2]，鸟的数量可以从几十只到上千只，它们可以呈现出复杂而多变的形态；在水中游动的鱼群^[2]，一个包含成千上万条个体的鱼群可以呈现如漩涡一样的美丽图案；在田野里运动的大型蝗虫群^[2]，它们所到之处，庄稼就会化为乌有；在陆地上辛勤劳作的蚂蚁群^[2]；除此之外，还有草原上奔跑的羊群、野牛群和斑马群等。另外一个更为鲜明的例子则是运动的人群^[2]形成的奇妙图案，比如在大型场合（如车站、集市等），成千上万的人同样会在不知不觉中形成特殊的图案。

在我们人眼无法观察到的微观世界中，也有运动群体呈现的丰富多彩的图案，并且十分常见。如图??所示，大量的细菌可以在运动时形成十分吸引人的形状，如大肠杆菌^[2]、枯草杆菌^[2]等。

要发现这些群体运动背后的根本规律、要检测与评判群体运动的行为特性、要验证相关理论的正确性，就需要知道群体中的每只个体是如何运动的，即知道每只个体的三维坐标是如何随时间而变化的，换言之，需要测量运动群体中每只个体的三维运动轨迹。

图 1-1 自然界中常见的大规模运动群体。(a)蝗虫群；(b)美洲红蚁群；(c)金鲱群；(d)漩涡状的鱼群；(e)天空中飞舞的棕鸟群；(f)斑马群；(g)人群；(h)羊群（图片来自^[2]）

图 1-2 微观世界中的大规模运动群体。(a)枯草杆菌群；(b)大肠杆菌群；(c)斑马鱼胚胎发育时的表面细胞群（图片来自^[2]）

图 1-3 匈牙利科学家 Vicsek 在大规模群体运动方面的一些研究成果，左图为鸽群层次结构的研究；右图为人群运动的研究（图片来自^[2]）

图 1-4 红眼果蝇（图片来自^[2]）

1.1.2 动物群体行为分析与建模

近年来，群体运动行为的研究吸引了众多信息科学、物理学、生物学领域的科研人员^{[2][3][4]}，相关交叉学科的研究成果不仅增进了人类对自然的认识，而且服务于社会经济发展，具有广阔的应用前景。

探索群体运动的机理将有利于人类社会诸多领域的发展和进步，如科学计算、军事、医学、交通、农业和动漫电影产业^[2]等。如：人类通过研究蚂蚁在寻找食物时的群体运动，提出了一种模拟进化算法——蚁群优化算法（ACO, Ant Colony Optimization）^[2]，用于在图中寻找特定的优化路径。可以解决如作业调度问题（JSP, Job Scheduling Problem）、旅行商问题（TSP, Travelling Salesman Problem）、最大独立集问题（MIS, Maximum Independent Set Problem）等，可以得到较好的优化效果。又如：科学家们研究大规模人群的交通流模型^[2]，从而研究怎样减少交通堵塞，更好地疏导交通。

果蝇、斑马鱼等模式生物的群体三维运动行为日益受到关注。2009 年，Nature Methods 刊登了一篇题为 No fruit fly an island? 的社论，指出研究果蝇群体飞行行为的重要意义："果蝇作为一种有明显群体行为的生物（如图??所示），研究其群体环境下的飞行行为将有利于揭示更多果蝇运动的神经机理，比如它们如何在群体情景下做出避碰的决策，其飞行行为如何受到性别比例、物种比例的影响？"要发现这些群体运动背后的根本规律、要检测和分析群体运动行为的特征、要验证相关理论的正确性，需要两个方面的关键理论与技术：（1）准确测量运动群体中每只个体的三维运动轨迹，得到每只个体的瞬时三维坐标；（2）检测和分析群体三维运动行为特性的理论与方法，群体行为的分析与建模。

1.2 群体三维运动研究方法综述

早期对大规模群体运动的研究主要采用肉眼观察和定性分析的方法，因为定量研究群体运动需要借助先进的传感器技术，所以受科学技术水平的制约，群体行为的定量研究上世纪六、七十年代才开始。主要测量群体中每只个体的运动轨迹，然后做相关分析。

早期的相关工作有：

1987 年，美国的计算机图形学专家 Reynolds 提出了一个模拟自然界的鸟群、羊群、鱼群等动物运动群体的基于分布行为的模型（BOIDS）^[2]，模型提出三方面的假设，即：

1) 群体中的每个个体都会尽量避免与周围个体的碰撞；2) 每个个体都会尽量将自己的运动方向调整到与周围个体一致的方向；3) 每个个体都会朝周围个体运动的目标位置运动；基于这些假设，该模型成功地模拟了自然界一些不同群体的运动，图??是他们对于鸟群的模拟结果。目前这些模拟群体运动的模型已越来越多地用在动画、游戏等商业领域。

图 1-5 Reynolds 的 BOIDS 模型模拟鸟群的结果（图片来自^[2]）

1995 年，匈牙利著名生物物理学家 Vicsek 从数学模型入手，对自然界的群体运动行为进行了研究与分析。提出自排斥粒子模型（Self-propelled particle model），将运动的群体看成一个个粒子组成的复杂系统，用物理的手段进行分析建模。当改变参数，使个体密度到达一定程度时，群体的运动呈现出有序而奇妙的状态，该研究成果发表于当年的物理学顶级期刊 *Physical Review Letters* 上^[2]。

随着各种传感器技术的不断发展，我们可以得到越来越精确地进行群体运动的实际测量。当然，这也意味着越来越大的数据量，对算法性能的要求也越来越高。

目前，群体运动的测量研究一般包含以下步骤：

- 寻找需要研究的大规模群体运动。
- 利用不同的传感仪器（如高速相机，GPS，雷达等等）进行跟踪拍摄。
- 通过各种不同的计算机算法分析传感器获得的群体运动数据，得到群体中每只个体的运动轨迹。
- 分析通过运行群体运动跟踪算法获得的运动轨迹数据，进一步获得群体或其中个体更多的运动信息，如速度、加速度、方向变化、同步性。
- 综合获得的所有数据，建立不同的数学、物理模型，从不同角度（生物学、物理学或数学等角度）分析动物的群体行为。

1.2.1 GPS 测量三维轨迹

GPS 可以提供瞬时的确定坐标，所以只要安装一个 GPS 接收及无线通信装置到被测个体上，就可以持续地测定该个体的瞬时坐标并发送给带有相应无线通信装置的计算机，这样就得到了该个体的运动轨迹^[2]。用 GPS 来测量个体的运动轨迹的优点是测量范围大、时间精度高，缺点是空间精度较低、另外该方法需要在被测个体上安装 GPS 装置。基于这些特点，GPS 适用于大范围运动而且方便安装 GPS 装置的个体，例如，运动的汽车、轮船、飞机、体型较大的动物。但不适用于小范围内运动的较小的个体，例如，蜜蜂、蝗虫、果蝇、斑马鱼等。对于这些群体运动，GPS 的空间精度不够，而且无法在这些小型个体上安装 GPS 装置。

2010 年，匈牙利 Etvos 大学和英国牛津大学的研究人员合作，将微型 GPS 模块绑在鸽子身上，如图??所示，他们测量了十只鸽子的飞行轨迹，基于信鸽方向选择的特征

延迟时间,通过研究数据中成对交互的头领角色,发现了鸽群飞行中群体成员间的层次结构,如图??。鸽群中存在"首领"与"追随者",首领通常位于鸽群前端,并且能更大程度上指引鸽群的飞行方向。头领与追随者除了在前后位置上有区别,另外还发现,它们在左右位置上也不同。而是否能成为鸟群的"首领"与鸟的导航能力有关,导航能力越强的鸟越有可能成为鸟群的“首领”。从进化的角度看,这种层次结构飞行时的组织比一个平均结构的飞行组织更有利。

图 1-6 身上绑有微型 GPS 模块的鸽子 (图片来自^[21])

图 1-7 鸽群飞行中的层次结构 (图片来自^[21])

1.2.2 多摄像机测量三维轨迹

该方法用多台高速相机,几何定标后同步拍摄运动物体^[21]。每台相机的成像平面上一只个体影像的坐标确定空间的一条射线,对应于同一个体的源自不同相机的多根射线的交点即为该个体的三维空间位置,再加上所记录的每帧图像的拍摄时间,即可得到该个体的运动轨迹。用多相机摄影测量个体运动轨迹的优点是:空间和时间的精度都比较高、不需要在个体上安装传感器装置、不需要接触个体、不影响个体的运动。缺点是测量范围较小。基于上述特点,该方法适用于小范围内运动、个体比较小、不适宜安装传感器装置的群体。上世纪六十年代起就有科研人员开始利用摄影测量研究群体的三维结构。早期的工作使用胶片相机/模拟视频摄像机拍摄,然后依靠人眼来定位和匹配照片上的个体,接着通过双目视差、影子等的几何关系计算出个体的三维坐标。代表性工作包括:

1965 年,英国牛津大学、美国自然历史博物馆、美国亚利桑那州传感系统实验室的 Cullen 等联合提出用一台胶片相机加上两个棱镜来产生两台相机的视差效果来测量鱼群中每条鱼的三维坐标的方法^[21],论文提出了通过鱼及其影子的几何关系来测量鱼的三维坐标的方法。

1978 年,加拿大西蒙菲莎大学的 Major 等用两台 35 毫米胶片摄影机,固定在金属杆上,相距约 5.5 米,平行地同步拍摄鸟群^[21],如图??所示。拍摄完毕后,他们人工定位和匹配胶片上飞鸟的影像,然后依据双目立体几何关系计算出飞鸟的空间坐标。

图 1-8 两台摄像机同步拍摄装置 (图片来自^[21])

1980 年,英国牛津大学、北爱尔兰新阿尔斯特大学、澳大利亚莫纳什大学的 Partridge 等联合用模拟视频摄像机拍摄鱼群,然后把视频复制到电影胶片上,接着人工

定位和匹配胶片上的鱼及其影子的影像，并通过鱼和影子的几何关系来计算鱼的三维坐标^[2]。

这些早期的方法依赖人眼来定位和匹配照片上的个体，这些方法的缺点是工作量大，而且容易出错。另一个局限性是这些方法仅测量个体的三维坐标，而不能测量个体的运动轨迹，因而无法用于研究三维群体运动行为。近年来，高速高清数码摄像机的发展，计算机的运算速度、储存容量、通信带宽的长足提高，使全自动三维跟踪测量群体中每只个体的运动成为可能。代表性工作包括：

2006 年，美国佐治亚理工学院 Khan 等利用基于马尔可夫链蒙特卡罗（MCMC, Markov chain Monte Carlo）的粒子滤波器跟踪数十只蚂蚁的二维运动^[2]，但该算法时间复杂度较大，不适宜用于跟踪大规模群体的运动，并且文章只跟踪了蚂蚁群的二维平面群体运动。

2007 年，复旦大学的陈雁秋等提出通过对运动个体在视频图像平面上的运动轨迹片段的匹配来实现双目视频图像中的个体影像的匹配^[2]，从而重构其在三维空间中的运动轨迹，该方法可以测量运动群体中每只个体的三维运动轨迹的部分片段。该方法综合和考虑了对极约束关系、图像上的点的运动速度关系和图像的纹理信息，提出了一个描述匹配程度的函数。将轨迹间的匹配问题转化为二分图最佳匹配问题，在较低的时间复杂度内求解。

2008 年，美国加州理工大学的 Maimon 等使用 5 台视频摄像机在三维空间中跟踪 20 只果蝇，实验装置如图??，他们分别研究了果蝇在自由飞行和在被拴住飞行时对不同尺寸的障碍物的反应情况^[2]，他们测试了无障碍物、短障碍物和长条型障碍物三种情况下果蝇的飞行行为，发现果蝇被较长的条状障碍物吸引，而躲避较小的障碍物。

图 1-9 加州理工大学的 Maimon 等使用的装置图及三种障碍物（图片来自^[2]）

2008 年，意大利 Centre for Statistical Mechanics and Complexity, Istituto dei Sistemi Complessi, 和 Universita' di Roma "La Sapienza" 的 Cavagna 等联合提出用 3 台相机测量大型鸟群中每只个体的三维坐标的方法^[2]，他们的研究发现受交互支配的动物群体行为取决于拓扑距离而不是度量距离，这样可以使鸟群有更高的凝聚力。而且在鸟群飞行时某一只鸟周围其他鸟的分布具有各向异性，在这只鸟飞行方向上的其他鸟数量较其他方向少，并且各向异性随着距离的增加而减弱。

2009 年，复旦大学的陈雁秋等提出全局最优匹配选择算法^[2]和三步线性分配算法^[2]，实现了对数百只同时飞行的果蝇的三维飞行轨迹的测量。其中，全局最优匹配选择算法构造了一个总的费用函数，该函数由三项组成，分别为：对极约束，动力学一致性和配置与观察的匹配程度。对于飞行轨迹整体使用基于 DP 思想的全局最优算法，使总的费用函数得到最小值，以此得到重构的三维轨迹。三步线性分配算法将果蝇三维运动轨迹的重构分成三个阶段考虑，将这三个阶段抽象为三个线性分配问题，即（跟踪两个视图中的二维轨迹；两个视图的二维轨迹的匹配；轨迹的三维重构），分别使用匈

牙利算法求解。以上两种算法均可以在较低的时间复杂度下解决轨迹匹配和重构问题,并且对因为测量误差和其他原因导致的跟踪丢失进行了单独处理,对中断的轨迹可以较好地进行重新连接。图??是实测的果蝇飞行轨迹。

图 1-10 数百只同时飞行的果蝇的三维飞行轨迹 (图片来自^[21])

2009 年,美国波士顿大学、美国北卡罗来纳大学的 Zheng Wu 等联合用三台红外摄像机重构测得了百余只蝙蝠的三维飞行轨迹^[21]。在进行轨迹匹配时提出一种迭代贪心随机自适应搜索 (IGRASP) 算法,能处理一部分遮蔽和轨迹交换问题。但由于算法使用迭代搜索,且需要大量枚举,算法结果优劣受迭代次数和临界值 τ 影响较大,且算法效率略有不足,在每帧 100 只蝙蝠的情况下每帧处理时间为 3 秒,因为实拍蝙蝠飞行时每秒拍摄 100 帧以上图片,若每帧处理时间为 3 秒,则该算法不能如文章中所说做到实时处理。

2009 年,美国加州理工学院、美国凯斯西储大学的 Kristin Branson 等联合提出一种用摄像机跟踪测量数十只在平面上运动的果蝇的二维运动轨迹的方法^[21],在对飞行轨迹进行分析时对果蝇飞行行为进行了分类。之后,用果蝇飞行行为预测其性别并对预测准确度进行了分析。

2010 年,美国加州理工学院的 Andrew D.Straw 等人提出一个采用 11 台摄像机的实时三维跟踪 3 只果蝇的系统^[21]。多目标跟踪算法使用扩展卡尔曼滤波和最近邻域数据关联算法。他们的系统的可以实时跟踪这三只果蝇的轨迹,帧率达 60fps。但该多目标跟踪算法具有局限性,不能用于处理更多果蝇飞行的轨迹。

2010 年,德国科学家 Schaller 等对肌动蛋白丝进行了二维轨迹跟踪^[21],个体数量达数百至上千个。进行轨迹分析后发现了肌动蛋白丝方向一致地运动时形成的旋涡状模式(如图??)。但该论文只得到这些肌动蛋白丝整体的运动速度场,没有对其中单独的个体进行跟踪。

图 1-11 肌动蛋白丝方向一致地进行运动时形成的旋涡状图像 (图片来自^[21])

同年,得克萨斯大学奥斯汀分校的 H.Zhang^[21] 等对密度较大的细菌群体进行了二维轨迹跟踪,在细菌群体非常密集的情况下依然能做准确地跟踪。通过分析细菌的群体运动他们发现,细菌群体会形成一个个动态的簇。当细菌的群体密度增加时,会大大增加它们形成大规模的簇的可能性。但是,关于细菌或其他微观世界的个体微粒的群体研究目前绝大部分仍停留在二维轨迹的跟踪与分析阶段。如果可以较准确地测量微观微粒的三维运动,将使科学家们能更深入地研究细菌等微观生物的群体行为,并且有可能帮助医学工作者防治和控制细菌传播造成的疾病。

1.3 课题目的及意义

近半世纪以来,随着大规模甚至超大规模的存储设备、高速高清数码摄像机的设计制造工艺及技术飞速发展,计算机视觉已成为测量研究群体运动的主要方法。科学家们已借助计算机视觉手段测量研究了多种动物的群体运动(见??总结描述)。这些研究发现了自然界中得多种群体行为的不同特点,但这些研究仍有一些难题未能完全解决。

- 自然界中的很多生物群体往往数量巨大且密度很高,拍摄时不同个体互相遮挡,使得从多张图片序列中得到其中每只个体准确的运动轨迹非常具有挑战性。
- 自然界的大规模群体中每只个体外表相似,体积大致相同,表面纹理很相似,利用以往模式识别中的检测、分类、特征提取、匹配算法很难确定每只个体准确的运动轨迹。
- 在处理跟踪大规模群体运动时,为了得到较准确的结果、尽量避免因遮挡造成的跟踪失败,通常需要多次迭代搜索一类的算法,造成整体算法复杂度高、内存占用大、且实时性不足。
- 一些算法只能做到半自动操作,在发生遮挡或者丢失跟踪时需要手工操作,以保证结果的准确性。

为了克服以往算法的上述不足,本文基于多台高速相机摄影测量,提出了进行该实验的实验方案、实验平台的搭建,并提出一种跟踪算法,用于跟踪果蝇的大规模群体运动。果蝇(*Drosophila*)是被广泛应用于基因研究、动物行为研究、动物神经学研究的一种模式生物。获得其群体飞行轨迹、研究其飞行行为与避碰策略,可以探索其神经系统的避碰机制,对生物科学领域有着重要意义。

本文所提出的实验平台利用两台高速相机,通过同步高速拍摄果蝇群体、准确重构果蝇飞行轨迹,从而跟踪在三维空间飞行的大规模果蝇群体,该方法主要有以下优点:

- 将相邻帧间、两个视角间的果蝇飞行轨迹的匹配转化为最优二分匹配问题。
- 当果蝇之间发生遮挡、跟踪丢失时,该跟踪算法能很好的处理。
- 该算法时间、空间复杂度较低,可以在较短的时间内就得到果蝇群体飞行的较准确的轨迹数据。

1.4 本文的组织结构

在本文的第??章中,介绍了群体三维运动测量与分析的相关知识背景、相关工作与相关方法。包括多相机测量平台、相机标定的过程、图象目标检测、跟踪、匹配及目标三维重建的方法。

在第??章中,详细介绍了本课题,即果蝇群体三维运动跟踪的算法及具体实现方法。

在第??章中,给出了果蝇群体运动实验的结果,包括相机标定、目标检测以及跟踪算法的运行结果。最后,本文对这些结果进行的详细的评估与总结。

最后，在第??章中，对全文所作工作进行了总结，并进行了展望，给出了未来改进的方向。

2 相关方法综述

计算机视觉领域的发展开始于 20 世纪 70 年代后期，随着计算机性能的飞速发展，使得拍摄、处理高清图像及大规模数据成为可能。利用计算机视觉相关技术解决大规模群体运动的跟踪问题，需要多方面知识的结合^[2]。其中最重要的是多相机测量平台的搭建、图像中的目标检测、多帧多视角间的目标匹配跟踪以及三维轨迹重建。这些步骤的相关关系如图??所示。本章将对这些工作进行介绍和分析。

图 2-1 三维群体运动跟踪操作步骤

2.1 多相机摄影测量平台

本课题研究采用多相机测量平台对果蝇群体的三维飞行轨迹进行测量。实验平台构造如图??所示^[2]。

图 2-2 多相机测量平台（图片来自^[2]）

图 2-3 放置果蝇的有机玻璃箱

果蝇被放置于一个透明的立方体有机玻璃箱中，箱的大小约为 40cm*40cm*40cm，厚度约为 3mm，如图??所示。用两台 (IO Industries) 高速相机从不同角度对其进行同步拍摄。一般一次实验放入 100-400 只果蝇，相机帧率设置为约 100fps，可以非常清楚地拍摄到每只果蝇的位置。图像分辨率 100 万像素 (1024*1024)，图片导出为 jpg 或 bmp 格式，后续操作非常方便。图??为实验室的真实情景。若拍摄时周围场景光线不足，则在有机玻璃箱后部放置两个 2000w 白色无频闪光源 (如图??) 进行照射，则可以得到清晰的图片，背景明亮干净、果蝇呈现为一个个黑（灰）色粒子状。后续步骤中可以通过简单的背景剪除操作去除图像背景。

2.2 相机标定

根据多视图几何的相关知识^[2]，为了对果蝇群体飞行轨迹进行三维重构，必须使用至少两个相机从不同的视角定标后进行同步拍摄，然后根据相机与场景的几何关系对场景进行三维重建。

图 2-4 多相机实验平台

图 2-5 白色无频闪光源

两个相机与图像平面的关系如图??，其中 c_0 和 c_1 代表两台高速相机的光心， $p \in \mathbb{R}^3$ 代表三维场景中的一点， $x_i \in \mathbb{R}^2$ 代表三维点 p 在第 i 个视角的图像平面坐标。 \mathbf{P}_i 代表第 i 个相机视角下三维空间到二维空间的投影矩阵，则存在关系：

$$x_0 = \mathbf{P}_1 p; x_1 = \mathbf{P}_2 p \quad (2-1)$$

所以，为了使用两台相机重建物体的三维坐标，需要得到每个视图对应的投影矩阵 \mathbf{P}_i ，建立空间中的三维点与图像平面的二维点之间的关系。这就需要进行相机标定（camera calibration）^[2]。用一大一小尺寸已知的物体作为标定物，得到标定物上的点与相机图像上的点之间的对应关系，利用相应的标定算法获得投影矩阵^[2]。而 \mathbf{P}_i 又可以分解为相机的内部参数（intrinsic parameter）和外部参数（extrinsic parameter）两部分，表达式如下：

$$\mathbf{P} = \mathbf{K} \mathbf{R} [\mathbf{I} \mid -\mathbf{C}] \quad (2-2)$$

式中， \mathbf{K} 为相机的内部参数，是相机内部所固有的参数，不随拍摄场景、时间等外部环境而改变，包括焦距 f 等参数。其余字母为相机的外部参数，代表相机与场景三维空间的相对位置关系， \mathbf{R} 是旋转矩阵，代表相机坐标系相对于世界坐标系的方向， \mathbf{C} 代表场景世界坐标系下相机中心的位置。

目前，相机标定算法已经较为成熟，已有的算法已经能比较精确地得到较精确的相机内外参数。被广泛使用的相机标定方法主要有两种：一是 Tsai 于 1987 年提出的使用表面有标识的专用三维标定物进行标定^[2]；二是张正友于 1999 年提出的使用带有标识的平面标定板，多视图、多角度进行拍摄，得到相机内外参数。目前，基于以上两种标定方法，已有较为成熟的 Matlab 工具箱^[2]，拍摄完标定图片之后可以方便地进行标定操作。

本课题中，采用张正友提出的标定方法，标定板为黑白棋盘格，如图??所示，每一小格的尺寸为 1.5cm*1.5cm。

2.3 目标检测

在得到了每个视角在一段时间内的连续若干帧图片之后，首先需要进行目标检测（object detection），即依次处理每幅图片，从图片中检测出每只运动的个体。目前，目标检测有很多种不同的算法，大致有背景剪除法（background subtraction）和图像分割法

图 2-6 两视图几何关系 (图片来自^[21])

图 2-7 黑白棋盘格标定板

(image segmentation) 两类^[2]。

背景剪除法的思路是将研究对象看做图像的前景，其余部分为图像的背景，如果前景与背景在颜色上有较大差异，则可以利用颜色性质来区分图像的前景和背景。假设 F 和 B 分别代表图像的前景与背景的颜色值（一般为灰度图像的灰度值），则它们满足下式关系：

$$\begin{cases} 1 & |F(x, y) - B(x, y)| > I_{thr} \\ 0 & otherwise \end{cases} \quad (2-3)$$

其中 I_{thr} 为区分前景与背景的阈值。此方法的关键是定义何为图像的背景以及图像背景与前景的阈值的选择。主要方法有：另外拍摄一幅没有物体的图片作为背景，此法非常方便，但不适用于在拍摄过程中背景发生变化的情况。或者使用时域中值滤波 (temporal median average)^[2]，选取连续 n 帧的背景灰度值的平均值作为整个拍摄图像序列的背景灰度值。此方法可以在一定程度上解决拍摄的图片背景发生变化的情况。但是不适用于较复杂或者不稳定的背景。

图像分割法主要利用相关技术直接从图像中分割出所关注的物体，得到其位置和形状信息。图像分割方法众多，适用于不同的图像，常见的图像分割方法有以下几种：

- 阈值分割法 (thresholding)^[2]；
- 均值漂移法 (mean shift)^[2]；
- 基于图论 (graph theory) 的分割法^[2]；
- 水平集法 (level set)^[2]；

阈值分割法需要确定一个阈值来分割前景物体与背景，与前面提到的背景剪除法类似，简单方便，但阈值的选择直接影响到分割的效果，所以如何选取合适的阈值是非常具有挑战性的工作。

均值漂移法基于概率密度分布思想，是一种无参的取样。在特征空间里建立三维窗口，特征空间里的一点对应一个五维向量，考虑邻近窗口对该窗口的贡献值，假设其会向更密集的区域转移，计算移动后的新坐标位置，迭代数次之后改坐标值趋于稳定。如此，先对图像进行平滑操作，利用平滑之后的图像建立区域邻接矩阵或者区域邻接链表，在特征空间比较接近的且在二维的图像平面也比较接近的像素算成一个区域，这样就对应一个区域的邻接链表，合并两个表面张力很接近的相邻区域。最后合并面积较小

的区域，对于待合并的较小的那个区域，寻找其所有的邻接区域，找到距离其最近的那个区域，合并之到那个区域。

常见的基于图论的图像分割方法有归一化切割（normalized cut），将图像中每个像素点看成一个无向图 G 中的一个结点，利用像素间的欧氏距离和灰度差异信息设立边权 w ，将图像分割问题转化为最小化一个归一化函数（如式??）的问题。最后图上所有的结点被分为两个集合， A 和 B 。

$$N_{cut} = \frac{cut(A,B)}{assoc(A,V)} + \frac{cut(A,B)}{assoc(B,V)} \quad (2-4)$$

$$cut(A,B) = \sum_{a \in A, b \in B} w(a,b) \quad assoc(A,V) = \sum_{a \in A, b \in V} w(a,b)$$

上式中 $w(a,b)$ 表示边权。但是，当物体与背景灰度区别不明显或是物体间有遮挡发生时，该方法分割效果无法让人满意。

利用水平集方法进行图像分割的原理是将 N 维闭合曲线（面）的演化问题转化为 $N+1$ 维空间中水平集函数曲面演化问题求解。可以用于处理变化的拓扑结构。

在本课题中，因为拍摄果蝇时背景基本不变，且拍摄的图片背景与图片上的果蝇个体灰度值具有明显差别，所以，本文采用简单方便的背景剪除方法对图像进行分割。在向实验箱中投放果蝇之前，先拍摄一张照片作为背景照片，然后进行拍摄。拍摄完成进行图像处理时，对于每张图片，将其转换为灰度图像，然后进行反色变换处理，之后，用该图片减去背景照片，即得到剪除背景之后的照片，效果良好。此时，果蝇为白（灰）色粒子状，背景接近纯黑色，二者的灰度值具有明显差别。然后，对这些图片进行二值化操作，使得背景成为黑色，果蝇个体为白色，方便后续处理。然后识别出图片中得每只果蝇个体，以每个白点区域的重心点代表一只果蝇的位置。

2.4 多目标跟踪

多目标跟踪问题的实质是一个随机过程（stochastic process）。包含两种随机变量，即传感器直接测量到的观察值（measurement，如个体的外形、纹理、二维坐标位置等等），以及个体隐含的运动状态（state，如三维空间坐标、速度、角速度、加速度等等）。对个体运动的跟踪就是一个概率推断问题，即利用传感器测量得到的观察值对个体的运动状态进行估计。

定义 t 时刻个体运动的状态变量为 \mathbf{S}_t ($t \in \mathbb{N}$, $\mathbf{S} \in \mathbb{R}^{n_s}$)， n_s 为状态变量空间的维数， t_i 时刻到 t_j 时刻的状态变量集合为 $\mathbf{S}_{t_i:t_j}$ ， \mathbf{S}_t^i 表示 t 时刻的第 i 个状态变量。定义 t 时刻个体运动的观察变量为 \mathbf{O}_t ($t \in \mathbb{N}$, $\mathbf{O} \in \mathbb{R}^{n_o}$)， n_o 为观察变量空间的维数， t_i 时刻到 t_j 时刻的观察变量集合为 $\mathbf{O}_{t_i:t_j}$ ， \mathbf{O}_t^i 表示 t 时刻的第 i 个观察变量。

求解该问题时，对该问题作一阶马尔可夫假设。即：

- 当前时刻个体的状态变量仅与前一时刻的状态变量相关，也即：

$$P(\mathbf{S}_t | \mathbf{S}_{1:t-1}, \mathbf{O}_{1:t-1}) = P(\mathbf{S}_t | \mathbf{S}_{t-1}) \quad (2-5)$$

- 当前时刻个体的观察变量仅与当前时刻的状态变量相关，也即：

$$P(\mathbf{O}_t | \mathbf{S}_{1:t}, \mathbf{O}_{1:t-1}) = P(\mathbf{O}_t | \mathbf{S}_t) \quad (2-6)$$

根据观察变量的时刻，即 t 与 τ 的大小，分为三种类型（ t 为状态变量时刻， τ 为观察变量时刻）：

- $t = \tau$ ，此时该操作为状态滤波
- $t < \tau$ ，此时该操作为状态平滑
- $t > \tau$ ，此时该操作为状态估计

而利用观察变量进行状态变量的推断的问题，即，最大化估计概率 $P(\mathbf{S}_t | \mathbf{O}_{1:\tau})$ 的问题。根据上述两条假设、贝叶斯规则（Bayesian Rule）以及马尔可夫链（Markov Chain）相关理论，可对 $P(\mathbf{S}_t | \mathbf{O}_{1:\tau})$ 做如下推导：

$$\begin{aligned} P(\mathbf{S}_t | \mathbf{O}_{1:t}) &= P(\mathbf{S}_t | \mathbf{O}_{1:t-1}, \mathbf{O}_t) \\ &\propto P(\mathbf{O}_t | \mathbf{S}_t, \mathbf{O}_{1:t-1}) P(\mathbf{S}_t | \mathbf{O}_{1:t-1}) \\ &= P(\mathbf{O}_t | \mathbf{S}_t) P(\mathbf{S}_t | \mathbf{O}_{1:t-1}) \\ &= P(\mathbf{O}_t | \mathbf{S}_t) \int P(\mathbf{S}_t | \mathbf{S}_{t-1}, \mathbf{O}_{1:t-1}) P(\mathbf{S}_{t-1} | \mathbf{O}_{1:t-1}) d\mathbf{S}_{t-1} \\ &= P(\mathbf{O}_t | \mathbf{S}_t) \int P(\mathbf{S}_t | \mathbf{S}_{t-1}) P(\mathbf{S}_{t-1} | \mathbf{O}_{1:t-1}) d\mathbf{S}_{t-1} \end{aligned} \quad (2-7)$$

上述递归估计概率的方法称为递归贝叶斯滤波器（recursive Bayes filter）。其中，第二步推导利用了贝叶斯规则，第三步推导利用了式??的假设，第四步推导利用了 Chapman-Kolmogorov 等式^[2]。

但是，式??一般不能得到解析解，所以，一般使用其他模型来估算概率密度 P 。目前，效果较好的、应用最广泛的方法是使用卡尔曼滤波器（Kalman filter）^[2]。该滤波器用线性高斯动态系统对目标的运动进行建模，状态变量和观察变量的计算更新方式如下：

$$\begin{aligned} \mathbf{S}_t &= \mathbf{F}\mathbf{S}_{t-1} + \boldsymbol{\omega}_t \\ \mathbf{O}_t &= \mathbf{G}\mathbf{S}_t + \mathbf{v}_t \end{aligned} \quad (2-8)$$

式中， \mathbf{F} 为状态转移矩阵， \mathbf{G} 为运动状态矩阵。 $\boldsymbol{\omega}$ 和 \mathbf{v} 分别表示状态变量和观察变量的噪声，且为零均值高斯白噪声，互相独立不相关。

传统的卡尔曼滤波器被用于跟踪线性运动系统，为使其能用于跟踪非线性运动的系统，^[2]中提出了扩展卡尔曼滤波器（Extended Kalman filter），使用两个非线性函数代替卡尔曼滤波器模型中的状态转移矩阵 \mathbf{F} 和运动状态矩阵 \mathbf{G} ，其更新方式如下：

$$\begin{aligned}\mathbf{S}_t &= f(\mathbf{S}_{t-1}, t-1) + \boldsymbol{\omega}_t \\ \mathbf{O}_t &= g(\mathbf{S}_t, t) + \mathbf{v}_t\end{aligned}\quad (2-9)$$

在实际跟踪个体运动时，还需要解决状态变量和观察变量的关联问题。即，在更新状态变量之前，将预测的状态变量同该时刻的观察变量相关联，在此基础上对每个个体的状态变量进行更新。得到数据关联问题的理论最优解的方法，即，记录所有可能的关联假设，但这将是一个 NP 问题，无法在多项式时间复杂度内计算出结果，所以不适用于跟踪大规模群体运动的情况。

本文使用基于连续两帧的数据关联。设 t 时刻时，个体数目为 N_t ，观察变量数目为 M_t ，相关联的观察变量与状态变量组成一个序列对，表示为： $\Lambda_{i,j}$ ，则第 i 个观察变量关联第 j 个状态变量表示为： $\Lambda_{i,j} = (i, j) \in \{0, \dots, M_t\} \{1, \dots, N_t\}$ ，所有关联序列对的集合为 Λ ， $\Lambda = \{\Lambda_{i,j}, i \in (0, \dots, M_t), j \in (1, \dots, N_t)\}$ 。其中包含零观察变量 \mathbf{O}_t^0 ，没有关联观察变量的状态变量将与之关联。这里使用基于联合数据关联 JPDA (Joint Probabilistic Data Association) 方法^[2]，估计观察变量 \mathbf{O}_t^i 关联于状态变量 \mathbf{S}_t^j 的后验概率，为：

$$\xi_{i,j} = P(\Lambda_{i,j} | \mathbf{O}_t^i) \quad (2-10)$$

利用贝叶斯规则，依据观察变量确定数据关联的计算推导如下：

$$\begin{aligned}P(\Lambda | \mathbf{O}_t) &= P(\Lambda | \mathbf{O}_t, \mathbf{S}_t) \\ &= \alpha P(\mathbf{O}_t | \Lambda, \mathbf{S}_t) P(\Lambda | \mathbf{S}_t) \\ &= \beta P(\mathbf{O}_t | \Lambda, \mathbf{S}_t)\end{aligned}\quad (2-11)$$

最后一步推导成立的原因是 $P(\Lambda | \mathbf{S}_t)$ 代表已知状态变量时，数据关联集合为 Λ 的概率，一般为一常数。 $P(\mathbf{O} | \Lambda, \mathbf{S}_t)$ 的意义是已知状态变量和数据关联集合时，观察变量集合为 \mathbf{O} 的概率。假设关联变量集合中，有数据关联的观察变量数量为 M'_t ，设在任意时刻，所有观察变量互相独立，另设一个观察变量没有数据关联的概率为 γ ，则成立：

$$P(\mathbf{O}_t | \Lambda, \mathbf{S}_t) = \gamma^{M_t - M'_t} \prod_{(i,j) \in \Lambda} P(\mathbf{O}_t^i | \mathbf{S}_t^j) \quad (2-12)$$

因此，

$$P(\Lambda | \mathbf{O}_t) = \beta \gamma^{M_t - M'_t} \prod_{(i,j) \in \Lambda} P(\mathbf{O}_t^i | \mathbf{S}_t^j) \quad (2-13)$$

假设相机视野的总体积为 V ，则 $\gamma = \frac{1}{V}$ ，状态变量 \mathbf{S}_t^j 的更新计算方法为：

$$P(\mathbf{S}_t^j | \mathbf{O}_{1:t}) = \beta \sum_{i=0}^{M_t} \xi_{i,j} P(\mathbf{O}_t^i | \mathbf{S}_t^j) P(\mathbf{S}_t^j | \mathbf{O}_{1:t-1}) \quad (2-14)$$

2.5 立体匹配

在进行相机标定与目标检测之后，就得到了相机的内外参数和每张图片上的所有果蝇的图像坐标以及两个相机视图中跟踪到的所有果蝇的二维飞行轨迹。下一步就是要对不同相机视角中的二维运动轨迹进行匹配，即立体匹配（stereo matching）。立体匹配主要利用的是多视图几何（multi-view geometry）中极线几何（epipolar geometry）的相关原理。

极线几何表征了三维场景中的点与不同相机视角内的投影点的对应关系，以及不同相机视图间的对应关系^[2]。以图??为例，图中，点 p ， c_0 ， c_1 ， x_0 ， x_1 在同一平面上，该平面称为极平面（epipolar plane），相机光心 c_0 ， c_1 的连线分别与两个相机图像平面相交于 e_0 ， e_1 两点，这两点称为极点（epipole）。在相机视图 1 中投影在极平面上点 x_0 的场景三维点都位于相机光心 c_0 与点 x_0 的连线上，这条直线在相机视图 2 中的投影为直线 l_1 ，称为极线（epipolar line），同理，相机视图 1 中的直线 l_0 亦为极线。因此，两个相机视图中的点满足以下极线约束（epipolar constraint），即，相机视图 2 中与视图 1 中的点 x_0 所匹配的点必然落在极线 l_1 上。因此， x_0 与 x_1 的几何关系可以表述为：

$$x_1 \mathbf{F} x_0 = 0 \quad (2-15)$$

l_1 与 x_0 的关系可以表述为：

$$l_1 = \mathbf{F} x_0 \quad (2-16)$$

其中，矩阵 \mathbf{F} 称为基础矩阵（fundamental matrix），大小为 3×3 ，自由度为 7。所以，只要找到两个相机视图中的至少 7 个已知的匹配点对就能确定基础矩阵 \mathbf{F} ，也可以通过以求出的相机内外参数来求解。

在进行匹配之前，一般需要再进行图像纠正（image rectification）操作^[2]，图像纠正的目的是将两个相机视角纠正至同一平面上，使得纠正后的两个相机视图的图像上的点只存在水平位移，这样能大大简化立体匹配操作的复杂度。图像纠正的过程如图??所示^[2]。

图 2-8 图像纠正的原理示意图（图片来自^[2]）

然后，进行立体匹配操作。立体匹配通常有两种思路，一种是局部窗口搜索算法^[2]，考虑待匹配个体间灰度的相似性。另一种是全局最优搜索算法^[2]，除了考虑灰度相似性约束，还加入了唯一性约束，第一个相机视图中的一个特征点只能对应匹配第二个相机视图中的一个特征点。全局最优搜索算法的具体实现方式有很多种，如动态规划算法^[2]、网络最大流算法^[2]等等。

本文采用第二种思路，即全局最优搜索算法，第一个相机视图中的一个特征点只能匹配第二个相机视图中的最多一个特征点，反之亦如此。本文用该方法可以得到全局最

优解。在具体实现时，建立二分图模型，采用二分图最佳匹配算法，关于该算法的原理及详细实现将在第??章中将详细讨论。

3 果蝇群体三维轨迹跟踪测量

3.1 引言

对于大规模动物群体如鱼群、鸟群、昆虫群等的社会行为的研究与探索,多年来一直吸引着多领域的科学家们。因为这些动物群体的运动不但构成了自然界一个个美丽的景象,同时,它们的群体运动反映了这些动物群体的社会行为机制。例如,哪些动物群体有群体行为?哪些动物群体有同步运动机制?个体在运动时如何与运动的群体中的其他个体交互?个体与个体在运动时有无避碰策略?等等。群体运动机制的相关研究不但能增进人类对自然界的认识,而且能造福人类社会,促进多领域、多学科如仿生学等的发展,应用前景广阔。

因此,就群体运动问题,多领域的科学家都从不同角度进行了深入研究。生物学家往往从生物特性,生物行为方面展开实验和分析^[2];物理学家往往建立数学或者物理模型来对群体运动行为进行模拟^[2];计算机科学家往往通过实验拍摄,利用各种算法进行分析^[2]。为了定量地精确地分析这些群体运动,必须建立一套自动、准确、高效的实验方法与算法。

但是,要做到群体运动测量的自动、准确、高效是非常具有挑战性的。因为大规模的动物群体往往有成百上千个个体,而且其在三维空间运动,在图片上手工测量标注工作量巨大,而且不能保证测量的准确性,所以必须借助高效、准确的测量仪器和相应高效、准确的跟踪算法。用一些先进的传感器技术如 GPS 全球定位系统可以对较大型、个体数量不多的群体进行跟踪测量^[2]。但是如果是对大规模群体进行测量,则对其中每个个体都负载 GPS 传感器将是基本无法完成的任务。另外,如果测量小型的运动个体,则根本无法将 GPS 传感器附在其身上,无法完成测量。

所以,用多台高速相机从多角度对动物运动场景进行拍摄,通过计算机视觉相关知识和技术利用这些图片序列重构动物运动轨迹是一种可行并且效果很好的解决方案。使用这种方法有以下几个技术难点:

- 运动个体尺寸非常小,测量时的较小误差都有可能很大地影响最后三维轨迹的重建效果。
- 每个运动个体外貌特征非常相似,很难通过外表特征对它们进行分辨。
- 由于运动个体数量、密度较大,在拍摄过程中经常发生遮挡现象,极大地影响每个个体轨迹跟踪的准确性。

因此,依靠外表特征(颜色、纹理等等)进行立体匹配的算法不适用于当前情况。另外,因为要对成百上千只运动个体进行跟踪,计算量非常大,所以对算法的时间、空间复杂度的要求也很高。

本文的算法基于两台高速相机的同步测量平台,可以较好地解决以上技术难点。整个跟踪过程分为四个步骤。第一步为目标检测,第二步为两维单视图的目标跟踪,第三步为两视图间的立体匹配,第四步为三维轨迹重建。这四个步骤会在下一小节中详细介绍。

3.2 果蝇三维轨迹跟踪的实现

3.2.1 目标检测

在本课题的实测实验中，实测拍摄时玻璃箱两侧使用大功率白色无频闪光源照明，所以图片背景与图片上的果蝇个体灰度值具有明显差别，如图??。因为拍摄果蝇时背景不变，因此，本文采用简单方便的背景剪除方法对图像进行分割。在向实验箱中投放果蝇之前，先拍摄一张照片作为背景照片，然后进行拍摄。拍摄完成进行图像处理时，对于每张图片，转换为灰度图像，进行反色变换之后，用之减去背景照片（背景照片也先转为灰度图像并做反色变换处理），即得到剪除背景之后的照片，效果良好，如图??所示。然后，对这些图片进行二值化操作，使得背景成为黑色，果蝇个体为白色，方便后续处理。然后识别出图片中得每只果蝇个体，以其重心点代表其位置，用红色点标记，如图??所示：

图 3-1 其中一台高速相机拍摄的其中一帧的图片

图 3-2 背景剪除并二值化图像后的结果

图 3-3 目标检测结果，每个红点代表其中一只果蝇的位置

该方法用 matlab 编程实现非常方便，读入图像后，彩色 RGB 图像在 matlab 中存储格式为一个大小为 $3 \times n \times m$ 的矩阵， n ， m 为图像的长宽像素值，矩阵中的数值大小为 $[0, 255]$ 。然后将图像转换为灰度图像，只需 `rgb2gray()` 函数即可，灰度图像的存储形式为 $n \times m$ 的矩阵，矩阵中得数值大小为 $[0, 255]$ 。然后对这些灰度图像进行反色变换，调用 `imcomplement()` 函数。之后，两幅图像相减只需调用 `imsubtract()` 函数。最后对相减之后得到的图像进行二值化处理，只需设置一个阈值，使像素灰度值高于这一阈值的像素点变为白点，值为 1，其余的变为黑点，值为 0，二值化操作只需调用 `im2bw()` 即可。

这里用此种方法另一好处是，可以去除一些恒定存在的图像噪声。如右下角导出图像时自带的图像标记戳、有机玻璃实验箱的棱角、表面的一些污渍、实验箱顶部倒果蝇的洞口等。

3.2.2 两维单视图目标跟踪

本文提出的两维单视图目标跟踪的模型在第??章中已作初步介绍。这里将每一时刻果蝇的二维图像坐标位置看做观察变量，将果蝇个体具体的运动信息（速度、角速度、

加速度等) 作为状态变量。而对果蝇个体的跟踪即为利用已知的观察变量估计当前的状态变量 (隐含) 的过程。

假设每个飞行的果蝇个体相互独立, 果蝇 i 在第 t 时刻的状态变量为 \mathbf{S}_t^i 。记录预测的果蝇二维坐标及速度信息, 即 $\mathbf{S}_t^i = (x_t^i, y_t^i, v_{t_x}^i, v_{t_y}^i) \in \mathbb{R}^4$ 。观察变量记录果蝇的二维坐标, 即 $\mathbf{O}_t^i = (x_t^i, y_t^i) \in \mathbb{R}^2$ 。根据式??, 则有:

$$P(\mathbf{S}_t^i | \mathbf{O}_{1:t}) = \alpha P(\mathbf{O}_t | \mathbf{S}_t^i) \int P(\mathbf{S}_t^i | \mathbf{S}_{t-1}^i) P(\mathbf{S}_{t-1}^i | \mathbf{O}_{1:t-1}) d\mathbf{S}_{t-1}^i \quad (3-1)$$

式中 α 为一常数, $\mathbf{O}_{t_i:t_j}$ 为时刻 t_i 到时刻 t_j 的观察变量集合。

式??的实际实现分为以下两个步骤:

- 状态预测 (state prediction) 由先前的观察变量预测当前的状态变量 (即用时刻 1 到 $t-1$ 的观察变量预测时刻 t 的状态变量), 如下式:

$$P(\mathbf{S}_t^i | \mathbf{O}_{1:t-1}) = \int P(\mathbf{S}_t^i | \mathbf{S}_{t-1}^i) P(\mathbf{S}_{t-1}^i | \mathbf{O}_{1:t-1}) d\mathbf{S}_{t-1}^i \quad (3-2)$$

- 状态更新 (state update) 由状态预测和当前的观察变量更新当前的状态变量:

$$P(\mathbf{S}_t^j | \mathbf{O}_{1:t}) = \alpha P(\mathbf{O}_t^j | \mathbf{S}_t^j) P(\mathbf{S}_t^j | \mathbf{O}_{1:t-1}) \quad (3-3)$$

对于运动状态的更新, 本文采用一阶马尔可夫假设, 用线性高斯动态系统 (Linear Gaussian Dynamic System) 来为每个果蝇个体的飞行运动建模, 其状态变量和观察变量的更新方式如下:

$$\begin{aligned} \mathbf{S}_t &= \mathbf{F}\mathbf{S}_{t-1} + \boldsymbol{\omega}_t \\ \mathbf{O}_t &= \mathbf{G}\mathbf{S}_t + \mathbf{v}_t \end{aligned} \quad (3-4)$$

在这里, \mathbf{F} 表示所跟踪的一只果蝇的状态转移矩阵 (从时刻 $t-1$ 到时刻 t), 反映了该只果蝇的运动状态信息。 \mathbf{G} 表示该只果蝇的观察矩阵。两个变量最后分别加上了零均值的高斯白噪声 $\boldsymbol{\omega}$ 和 \mathbf{v} , 它们互相独立且不相干。

假设时刻 t 的状态变量的估计值为 $\hat{\mathbf{S}}_t$, 为使状态变量估计值与实际值的均方差的期望 ($\mathbb{E}[\|\mathbf{S}(t) - \hat{\mathbf{S}}(t)\|^2]$) 最小, 这里使用卡尔曼滤波器 (Kalman filter) 模型^[2]。

由高速相机拍摄观察得知, 果蝇在飞行中基本处于匀速运动状态。因此, 这里使用匀速运动模型, 根据之前的状态变量预测当前的状态变量。即, 状态转移矩阵 \mathbf{F} 为:

$$\mathbf{F} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3-5)$$

其中 Δt 为连续两帧帧间采样时间间隔。则, 式??简化为:

$$\hat{\mathbf{S}}_t^i = \mathbf{F} \mathbf{S}_{t-1}^i \quad (3-6)$$

然后进行数据关联操作，将假设状态变量 \mathbf{S}_t^i 关联的观察变量为 \mathbf{O}_t^j ，则状态变量利用下式更新：

$$\mathbf{S}_t^i = \hat{\mathbf{S}}_t^i + \mathbf{Q} [\mathbf{O}_t^j - \mathbf{H} \hat{\mathbf{S}}_t^i] \quad (3-7)$$

其中 \mathbf{Q} 称为卡尔曼增益 (Kalman gain)^[2]，为简化状态更新操作，减小程序运行的时间开销，这里使用简化的卡尔曼滤波器，即 $\alpha - \beta$ 滤波器模型^[2]，增益函数 \mathbf{Q} 定义如下

$$\mathbf{Q} = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \\ \frac{\beta}{\Delta t} & 0 \\ 0 & \frac{\beta}{\Delta t} \end{bmatrix}, \quad 0 \leq \alpha, \beta \leq 1, \quad \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad (3-8)$$

$\alpha - \beta$ 滤波器模型中， α 和 β 参数为固定常数，代表测量变量的测量误差在状态变量的位置和速度值的更新中所占比例。这里使用 $\alpha - \beta$ 有以下优点：

- 增益参数为确定常数，计算开销很小，可以用于跟踪大规模群体的运动。
- 可以根据实际的观察变量的误差情况手动修改增益参数 α 和 β ，更加灵活。当噪声较大时，选择较小的增益参数，反之，选择较大的增益参数。

在进行状态更新操作前，最重要的是要将该时刻的观察变量与预测的状态变量进行数据关联操作。相关解决方法中最简便最常用的是最近邻算法，最近邻算法基于贪心 (greedy) 思想，简单地将每个预测的状态变量关联给距其最近的观察变量。虽然这种方法非常简便易懂，但由于跟踪大规模果蝇运动群体时，遮挡现象频繁发生，所以直接使用最近邻算法会导致较高的错误率。这里，本文将这一问题建模为一个线性分配 (linear assignment) 问题，可以得到全局最优解。

首先，定义观察变量与预测的状态变量的关联矩阵如下：

$$\mathbf{R}(i, j) = \begin{cases} 1 & \mathbf{O}_t^i \text{ matches } \hat{\mathbf{S}}_t^j \\ 0 & \text{otherwise} \end{cases} \quad (3-9)$$

关联矩阵 \mathbf{R} 的求解可转化为求解以下最大后验概率问题：

$$\mathbf{R} = \arg \max_{\mathbf{R}} P(\mathbf{R} | \mathbf{O}_t) = \arg \max_{\mathbf{R}} P(\mathbf{R} | \mathbf{O}_t, \hat{\mathbf{S}}_t) \quad (3-10)$$

根据贝叶斯规则，有：

$$P(\mathbf{R} | \mathbf{O}_t, \hat{\mathbf{S}}_t) = \alpha P(\mathbf{O}_t | \mathbf{R}, \hat{\mathbf{S}}_t) P(\mathbf{R} | \hat{\mathbf{S}}_t) \quad (3-11)$$

其中, 假设 $P(\mathbf{R} | \hat{\mathbf{S}}_t)$ 为一常数, 可将式??转化为:

$$\mathbf{R} = \arg \max_{\mathbf{R}} P(\mathbf{O}_t | \mathbf{R}, \hat{\mathbf{S}}_t) = \arg \max_{\mathbf{R}} \prod_{\mathbf{R}(i,j)=1} P(\mathbf{O}_t^i | \hat{\mathbf{S}}_t^j) \quad (3-12)$$

此外, 数据关联满足唯一性, 即, 每个观察变量至多只能关联一个预测的状态变量, 每个预测的状态变量也至多只能关联一个观察变量。定义 t 时刻观察变量和预测的状态变量的数目分别为 M_t 和 N_t , 则上述唯一性约束可写为:

$$\sum_{i=1}^{M_t} \mathbf{R}(i, j) = 1, \quad \sum_{j=1}^{N_t} \mathbf{R}(i, j) = 1 \quad (3-13)$$

于是, 求解关联矩阵 \mathbf{R} 的问题转化为以下最大后验概率问题:

$$\begin{aligned} \mathbf{R} = \arg \max_{\mathbf{R}} \quad & \prod_{\mathbf{R}(i,j)=1} P(\mathbf{O}_t^i | \hat{\mathbf{S}}_t^j) \\ \text{s.t.} \quad & \sum_{i=1}^{M_t} \mathbf{R}(i, j) = 1, \quad \sum_{j=1}^{N_t} \mathbf{R}(i, j) = 1 \\ & i = 1, 2, \dots, M_t; \quad j = 1, 2, \dots, N_t \end{aligned} \quad (3-14)$$

定义关联概率 $P(\mathbf{O}_t^i | \hat{\mathbf{S}}_t^j)$ 如下:

$$P(\mathbf{O}_t^i | \hat{\mathbf{S}}_t^j) = h(\|\mathbf{O}_t^i - \hat{\mathbf{S}}_t^j\|, \sigma^2) \quad (3-15)$$

函数 h 定义如下:

$$h(x, \sigma) = \begin{cases} \exp\left(-\frac{x^2}{\sigma^2}\right) & \text{if } x^2 < \sigma^2 \\ 0 & \text{otherwise} \end{cases} \quad (3-16)$$

σ 为一个阈值, 含义为: 若观察变量与预测的状态变量的空间距离大于该阈值 σ , 则这二者不相关联, 该阈值的选取一般通过考察具体的个体运动信息而定。

然后对关联概率函数取负对数, 可得以下函数 \mathbf{W} :

$$\mathbf{W}(i, j) = -\log P(\mathbf{O}_t^i | \hat{\mathbf{S}}_t^j) \quad (3-17)$$

该函数可看做关于数据关联的代价函数, 表示将观察变量 \mathbf{O}_t^i 与预测的状态变量 $\hat{\mathbf{S}}_t^j$ 关联的代价, 并且该值越小说明关联性越强, 解越优。此时, 式??描述的最大后验概率问题转化为一线性分配问题, 定义如下:

$$\begin{aligned}
\mathbf{R} = \arg \min_{\mathbf{R}} & \sum_{i=1}^{M_t} \sum_{j=1}^{N_t} \mathbf{W}(i, j) \mathbf{R}(i, j) \\
s.t. & \sum_{i=1}^{M_t} \mathbf{R}(i, j) = 1, \quad \sum_{j=1}^{N_t} \mathbf{R}(i, j) = 1 \\
& i = 1, 2, \dots, M_t; \quad j = 1, 2, \dots, N_t
\end{aligned} \tag{3-18}$$

通过 KM 算法，可以在 $O(n^4)$ 的时间复杂度内得到线性分配的全局最优解。关于 KM 算法的原理及实践，将在第??节中详细介绍。

在实际的果蝇运动跟踪时，有以下几种特殊情况需要考虑：

- 图像中的果蝇个体间发生遮挡现象
- 果蝇个体飞离相机视野或停歇在有机玻璃箱壁上，轨迹中止
- 新起飞的果蝇，新轨迹的出现

对应的解决方案如下：

- 该时刻，当观察变量找到关联的预测状态变量时，直接按式??进行状态变量更新。并将该更新后的状态变量设置为有效且有关联状态
- 该时刻，若某预测的状态变量没有找到关联的观察变量，则将该预测变量关联于一虚拟观察变量，更新方式如下：

$$\mathbf{S}_t = \hat{\mathbf{S}}_t \tag{3-19}$$

该种情况的出现通常因为果蝇个体间的遮挡引起，此时将该状态标记为有效但无关联状态。若某持续有效的轨迹连续 T_ϵ 帧为无关联状态，即连续 T_ϵ 帧未成功关联到对应时刻的观察变量，则视作该果蝇个体已飞离相机视野或者已经停止飞行，将此轨迹标记为已结束。

- 该时刻，若某预测的观察变量没有找到关联的预测状态变量，则新增一条新轨迹并进行相关的轨迹初始化操作，将此状态设置为有效且有关联状态。此种情况一般因新飞进相机视野或新起飞的果蝇个体引起，有时也可能是因图像噪声而引起的错误的观察变量引起。如果是因错误的观察变量而导致新轨迹的出现，则一般不会持续很久，所以继续使用上述阈值 T_ϵ ，若最后某条轨迹长度小于 T_ϵ ，则去掉这条轨迹。
- 每帧的数据关联处理时，只考虑该帧观察变量与状态为有效的状态变量的关联。

综上所述，两维单视图目标跟踪的总体流程如下图所示：

图 3-4 两维单视图目标跟踪的总体流程

3.2.3 立体匹配

当通过上一步得到了两个相机视图的所有果蝇的二维轨迹之后, 需要利用多视图几何相关知识, 在两个视图中寻找匹配的轨迹^[2]。在本课题中, 研究对象为飞行的果蝇群体, 其中每只果蝇个体很小, 只占不到 10 个像素点, 外形相似, 颜色灰度值相似, 所以不能用个体外形或者颜色特征来进行立体匹配。因此, 考虑使用个体的运动信息来帮助完成立体匹配。

设相机视图 1 中得到的轨迹集合为 T_0 , 相机视图 2 中得到的轨迹集合为 T_1 , 且 $T_i = \{T_i^1, \dots, T_i^{N_i}\}$, N_i 表示第 i 个相机视图中得到的轨迹个数, $N_i = |T_i|$, 其中一条轨迹 T_i^j 上的元素表示为 $(x_{t,i}^j, y_{t,i}^j, t)$, $(x_{t,i}^j, y_{t,i}^j)$ 为第 i 个相机视图中第 j 条轨迹在第 t 帧的二维图像坐标。这里的立体匹配问题定义为: 对于相机视图 1 中的每一条轨迹, 在相机视图 2 中找到与之匹配的轨迹, 定义 0-1 邻接矩阵 \mathbf{R} , 表示两个相机视图中的轨迹匹配情况, 有:

$$\mathbf{R}(i, j) = \begin{cases} 1 & T_0^i \text{ matches } T_1^j \\ 0 & \text{otherwise} \end{cases} \quad (3-20)$$

两个相机视图的二维轨迹的立体匹配可以转化为一最大后验概率问题^[2]。其中 \mathbf{F} 为在进行相机标定时, 已求得的相机基础矩阵。假设匹配关系 \mathbf{R} 的先验概率 $P(\mathbf{R} | T_1, \mathbf{F})$ 为常数, 则有:

$$\begin{aligned} P(\mathbf{R} | T_0, T_1) &= P(\mathbf{R} | T_0, T_1, \mathbf{F}) \\ &= \alpha P(T_0 | \mathbf{R}, T_1, \mathbf{F}) P(\mathbf{R} | T_1, \mathbf{F}) \\ &= \beta P(T_0 | \mathbf{R}, T_1, \mathbf{F}) \\ &= \beta \prod_{\mathbf{R}(i,j)=1} P(T_0^i | T_1^j, \mathbf{F}) \end{aligned} \quad (3-21)$$

两个相机视图间的二维轨迹匹配的最优化模型如下 (最大化后验概率 $P(T_0^i | T_1^j, \mathbf{F})$):

$$\begin{aligned} \mathbf{R} &= \arg \max_{\mathbf{R}} \prod_{\mathbf{R}(i,j)=1} P(T_0^i | T_1^j, \mathbf{F}) \\ s.t. \quad &\sum_{i=1}^{|T_0|} \mathbf{R}(i, j) = 1, \quad \sum_{j=1}^{|T_1|} \mathbf{R}(i, j) = 1 \\ &i = 1, 2, \dots, |T_0|; \quad j = 1, 2, \dots, |T_1| \end{aligned} \quad (3-22)$$

式中的约束条件表示轨迹匹配的唯一性, 即, 其中一个相机视图中任一条轨迹只能与另一视图中的一条轨迹匹配。

下面讨论如何求解轨迹的匹配概率 $P(T_0^i | T_1^j, \mathbf{F})$, 定义轨迹 T_k^i 的各帧图像索引为 τ_k^i 。 $\tau_k^i(bg)$ 和 $\tau_k^i(ed)$ 表示该段轨迹的第一帧和最后一帧。对于已进行图像纠正之后得到的图像, 所有匹配的轨迹中的点对满足下式:

$$\Phi_{T_0^i, T_1^j} = \{t \mid \|y_{t,0}^i - y_{t,1}^j\| \leq \epsilon, \quad t \in \{\tau_0^i, \tau_1^j\}\} \quad (3-23)$$

式中 ϵ 表示匹配误差阈值, $\{\tau_0^i, \tau_1^j\}$ 表示两条轨迹在时间轴上的交集。当这一时间轴交集集中的点都能匹配, Φ 中的元素为连续的时间序列。但是, 由于跟踪错误或者个体间发生遮挡, 会导致 Φ 断裂为若干个时间片段, 构成 Φ 的若干子集, 每个子集中为一个匹配片段的时间序列, 这些子集表示如下:

$$\Phi^p = \{\Phi^p(bg), \Phi^p(bg) + 1, \dots, \Phi^p(ed)\} \quad (3-24)$$

$$\Phi = (\Phi^1, \Phi^2, \dots, \Phi^{N_\Phi}) \quad (3-25)$$

$$\Phi^p \cap \Phi^q = \emptyset, \quad \forall p, q, \quad p \neq q \quad (3-26)$$

$$\Phi^{p+1}(bg) - \Phi^p(ed) \geq 2 \quad (3-27)$$

其中 N_Φ 表示 Φ 的子集的个数, $??$ 表示 M 的子集中至少匹配上了 2 帧。定义共极线长度 (epipolar co-motion length) $|\Phi^p|$ 为其中一个子集中的匹配上的帧数。则最大共极线长度可以表示为:

$$MECL(T_0^i, T_1^j) = \max \{|\Phi^p|\}, \quad p = \{1, \dots, N_\Phi\} \quad (3-28)$$

将 MECL 归一化如下:

$$MECL'(T_0^i, T_1^j) = \max \{|\Phi^p|\} \cdot \left(\frac{1}{\tau_0^i} + \frac{1}{\tau_1^j} \right) \quad p = \{1, \dots, N_\Phi\} \quad (3-29)$$

有 $0 \leq MECL' \leq 2$, 则轨迹匹配的概率可以根据 MECL' 计算如下:

$$P(T_0^i \mid \Gamma_1^j, \mathbf{F}) = \exp \left(-\frac{1}{MECL'(T_0^i, T_1^j)} \right) \quad (3-30)$$

对于两个相机视图中的轨迹, 匹配部分所占长度比例越大, 两个轨迹匹配的概率也越大。

对轨迹匹配概率取负对数, 可将式??转化为一个线性分配问题。

但是, 跟踪时发生的错误会导致两个轨迹只有部分匹配上, 因此, 本文采用迭代求解, 尽量使更多的轨迹得到匹配。设求解出的匹配上的帧序列为 $\tilde{\Phi}$, 则一条轨迹被分成如下三个部分:

$$T_k^a = T_k^i \left(\tau_k^i(bg) : \tilde{\Phi}(bg) - 1 \right) \quad (3-31)$$

$$T_k^b = T_k^i \left(\tilde{\Phi} \right) \quad (3-32)$$

$$T_k^c = T_k^i \left(\tilde{\Phi}(ed) + 1 : \tilde{\tau}_k^i(ed) \right) \quad (3-33)$$

$$k \in \{1, 2\}, \quad \tau \in \mathbb{Z}, \quad \tau \geq 0 \quad (3-34)$$

其中 T_k^b 为已匹配上的轨迹片段, T_k^a 和 T_k^c 为未匹配上的轨迹片段。分为两条待匹配轨迹存储于待匹配轨迹队列中, 继续迭代求解。

使用 MECL 进行匹配的优点有:

- 使用个体的运动信息和两个相机视图间的几何关系来实现匹配, 包含的用于匹配的信息量较大, 匹配的准确度较高。
- 因为使用迭代求解, 所以对于处理因跟踪错误或个体发生遮挡而致的匹配失败可以很好的处理。

3.2.4 三维轨迹重建与重连接

在上一节中, 已完成两个视图的二维轨迹间的立体匹配并得到三维轨迹, 但是由于两维单视图跟踪的错误或果蝇个体间的遮挡现象时有发生, 所以原本完整的运动轨迹被分为若干片段, 使得得到的轨迹数大于实际的轨迹数。所以, 在最后, 需要添加一步操作, 将因为上述原因而断开的轨迹进行重新连接, 从而得到完整的果蝇群体三维运动轨迹。

假设上一节中得到的轨迹集合为 T , 其中部分轨迹是因上述原因在前面步骤的计算中被中断成的若干轨迹片段, 即 $T = \{T^1, T^2, \dots, T^{N_T}\}$, 其中一个轨迹片段表示为: $T_t^i = \{x_t^i, y_t^i, z_t^i, t\}$ 。假设其中 n 条轨迹片段因连接在一起构成一条完整的轨迹 \hat{T}^i , 另假设轨迹的连接操作作用符号 $*$ 表示, 则 $\hat{T}^i = \{T^a * T^{a+1} \dots * T^{a+n}\}$ 。 \hat{T}^i 为一条完整轨迹的概率等价于其中所有相邻两个轨迹片段连接的概率的乘积, 表示为:

$$P(T^i) = P(\{T^a * T^{a+1} \dots * T^{a+n}\}) = \prod_{j=1}^{n-1} P(T^{a+j} * T^{a+j+1}) \quad (3-35)$$

同时, 用一个 0-1 关系矩阵 \mathbf{R} 表示两个轨迹片段的连接关系, 定义如下:

$$\mathbf{R}(i, j) = \begin{cases} 1 & \text{trajectory } T^i \text{ connects with } T^j \\ 0 & \text{otherwise} \end{cases} \quad (3-36)$$

因此, 中断轨迹的重连接可建模为以下最大化概率问题:

$$\begin{aligned}
\mathbf{R} &= \arg \max_{\mathbf{R}} \prod_{\mathbf{R}(i,j)=1} P(T^i * T^j) \\
s.t. \quad & \sum_{i=1}^{|T|} \mathbf{R}(i,j) = 1, \quad \sum_{j=1}^{|T|} \mathbf{R}(i,j) = 1 \\
& i, j = 1, 2, \dots, |T|
\end{aligned} \tag{3-37}$$

下面，讨论轨迹片段连接概率的计算。这里，考虑轨迹在时间与空间两方面的信息，只有这两方面的条件都符合约束条件时，才将这两条轨迹连接。

设有两条轨迹 T^i 和 T^j ，开始和结束时间分别标记为 bg 和 ed ，讨论以下两种情况：

- $1 \leq \tau^j(bg) - \tau^i(ed) \leq \delta$ ，表示轨迹 T^j 在轨迹 T^i 结束后 δ 帧之内开始，此时，认为轨迹 T^i 和 T^j 可能连接。设轨迹 T^k 在 t 时刻的真实三维空间坐标和预测的三维空间坐标分别为 $\mathbf{X}_t^k = \{x_t^k, y_t^k, z_t^k\}$ 和 $\hat{\mathbf{X}}_t^k = \{\hat{x}_t^k, \hat{y}_t^k, \hat{z}_t^k\}$ 。假设果蝇做匀速运动，将轨迹 T^i 向后预测 $\tau^j(bg) - \tau^i(ed)$ 帧，将轨迹 T^j 向前预测 $\tau^j(bg) - \tau^i(ed)$ 帧。定义这两条轨迹的距离为：

$$dis(i, j) = \frac{\sum_{t=\tau^i(ed)}^{\tau^j(bg)} \|\hat{\mathbf{X}}_t^i - \hat{\mathbf{X}}_t^j\|}{\tau^j(bg) - \tau^i(ed) + 1} \tag{3-38}$$

则同式??，可将轨迹 T^i 和 T^j 相连接的概率定义为：

$$P_{noverlap}(T^i * T^j) = h(dis(i, j), \sigma^2) \tag{3-39}$$

式中 σ 为一阈值，若两条轨迹的距离超过该阈值，则它们不能被连接。函数 h 的定义同式??。

- $1 \leq \tau^i(ed) - \tau^j(bg) \leq \delta$ ，表示轨迹 T^j 在轨迹 T^i 结束前 δ 帧之内开始，此时，认为轨迹 T^i 和 T^j 亦可能连接。这种情况出现的原因是在两维单视图轨迹的跟踪时，为避免可能的误差，保留有轨迹最后几帧，导致两个本应相连的轨迹可能在头和尾有若干帧重叠，此时，利用重叠的几帧的信息定义连接概率为：

$$P_{overlap}(T^i * T^j) = h(dis(i, j), \sigma^2) \tag{3-40}$$

阈值 σ 与函数 h 的定义同式??。但，轨迹距离的计算式变为：

$$dis(i, j) = \frac{\sum_{t=\tau^j(bg)}^{\tau^i(ed)} \|\hat{\mathbf{X}}_t^i - \hat{\mathbf{X}}_t^j\|}{\tau^i(ed) - \tau^j(bg) + 1} \quad (3-41)$$

综上所述，两个轨迹片段的连接概率为：

$$P(T^i * T^j) = \begin{cases} P_{noverlap}(T^i * T^j) & 1 \leq \tau^j(bg) - \tau^i(ed) \leq \delta \\ P_{overlap}(T^i * T^j) & 1 \leq \tau^i(ed) - \tau^j(bg) \leq \delta \\ 0 & otherwise \end{cases} \quad (3-42)$$

将轨迹片段连接的概率取福对数，如式??，有：

$$\mathbf{W}(i, j) = -\log P(T^i * T^j) \quad (3-43)$$

将此函数作为轨迹片段连接的代价函数，该函数值越大，说明这两条轨迹连接的概率越大。轨迹的重连接问题由此建模为一线性分配问题，使用 KM 算法求解。关于该算法的原理及实现，将在第??节中详细介绍。

3.3 线性分配问题的算法实现

本课题所研究的大规模果蝇群体的三维运动跟踪与轨迹重建，大致可分为七个步骤，即，相机标定、相机拍摄、图像预处理、目标检测、目标跟踪、立体匹配、三维轨迹的重建与重连接。在最后两个步骤中，将以下三个问题转化为线性分配问题，分别是：

- 在进行二维单视图多目标跟踪时，每帧的观察变量和该帧预测的状态变量之间的数据关联问题；
- 进行立体匹配时两个相机视图中得到的二维轨迹间的匹配问题；
- 轨迹重连接步骤中，中断的三维轨迹间的匹配问题；

所以，能否很好地解决线性分配问题对于本文所建模型的算法效率和最终结果的准确性有很大影响，是本课题中需要解决的重要问题之一。本文使用二分图模型，将上述三个问题转化为二分图最佳匹配（Bipartite Graph Best Matching）问题，利用相关算法（Kuhn-Munkras Algorithm）解决之。经实际测试，算法时间、空间复杂度较低，效率较高，准确度较高，很好地解决了问题。下面详细介绍该模型以及 Kuhn-Munkras 算法的原理及实现。

二分图（Bipartite graph）是图论中的一种常用模型^[2]。定义如下：设 $G = (V, E)$ 是一个无向图，其中 V 是其顶点集合， E 是其边集合，若顶点集 V 可分为两个互不相交的子集 A 与 B ，且满足边集合中的每一条边所邻接的两个顶点分别属于这两个不同的顶点集，则图 G 为一个二分图，如图??所示^[2]。

图 3-5 二分图示例

设 M 为图 G 的边集 E 的一个子集，若 M 中的任意两条边都不与相同顶点邻接，则 M 是图 G 的一个匹配。二分图的极大匹配 (Bipartite Graph Maximal Matching) 是指：对于该匹配 X ，在图中找不到可以增加的匹配边。二分图的最大匹配 (Bipartite Graph Maximum Matching) 则是二分图的各极大匹配中匹配边数最多的一个匹配。二分图的完备匹配 (Bipartite Graph Perfect Matching) 是指：在该匹配中，对于图中的每一个顶点，都存在边与其邻接。若该二分图为加权二分图，即该图的每条边都有一权值，则此二分图匹配边权值和最大的完备匹配称为该图的最佳匹配。

二分图的增广路径的定义为：若 P 是图 G 中与两个未匹配的顶点相邻接的一条路径，并且，属于匹配 M 的已匹配边和不属于 M 的未匹配边在 P 上交替出现，则称 P 是相对于匹配 M 的一条增广路径，如图??所画出的整条轨迹为一条增广路，其中画粗线的边表示已匹配的边。

图 3-6 增广路示例

求解二分图最大匹配问题的经典算法是匈牙利算法 (Hungarian Algorithm)^[2]。匈牙利算法的核心思想是，在二分图中不断寻找增广路径，直至找不到为止。时间复杂度为 $O(n^3)$ ，空间复杂度为 $O(n^2)$ 。

求解二分图最佳匹配问题通常使用 KM 算法 (Kuhn-Munkras Algorithm)^[2]。KM 算法的核心思想是，每个顶点设置一个标记，并将之转化为一个求二分图完备匹配的问题。假设集合 A 中顶点 i 的标记为 $X[i]$ ，集合 B 中顶点 j 的标记为 $Y[j]$ ，连接 i, j 的边权为 $w[i, j]$ 。且对任意 i, j ，总有： $X[i] + Y[j] \geq w[i, j]$ 。

若二分图内所有满足式： $X[i] + Y[j] = w[i, j]$ 的边 (i, j) 构成的图 G 的子图存在完备匹配，则该完备匹配即二分图的最佳匹配。

KM 的一种实现方法如下：

- 初始化时，将 $X[i]$ 置为与顶点 x 邻接的边的最大边权值， $Y[j]$ 置为 0。循环以下操作，直至找到其相等子图的完备匹配。
- 利用匈牙利算法寻找完备匹配。
- 若找到完备匹配，则结束；否则，修改顶点标记的值，修改方法如下：

此时，没有找到相等子图的完备匹配，即从某一顶点 x ，没有找到一条由它出发的增广路，将找增广路时经过的 X 集合内的顶点标记值都减小 d ，经过的 Y 集合内的顶点标记值都增加 d ，这些经过的点集构成一棵“交错树”。此时，考虑以下四种情况：

- (i) 边 (i, j) 的两个邻接顶点都在交错树中, 则 $X[i] + Y[j]$ 的值不变。所以, 原来属于相等子图的边现在仍属于相等子图。
- (ii) 边 (i, j) 的两个邻接顶点都不在交错树中, 则 $X[i] + Y[j]$ 的值不变。所以, 原来不属于相等子图的边现在仍不属于相等子图。
- (iii) i 在交错树中, j 不在, 则 $X[i] + Y[j]$ 的值增大 d , 该边原来不属于相等子图, 现在仍不属于相等子图。
- (iv) i 不在交错树中, j 在交错树中, 则 $X[i] + Y[j]$ 的值减小 d , 该边原来不属于相等子图, 现在有可能属于相等子图。

下面讨论 d 的求法, 已知式 $X[i] + Y[j] \geq w[i, j]$ 始终成立, 为保证至少有一条边能被加入相等子图, d 取:

$$d = \min \{X[i] + Y[j] - w[i, j] \mid i \text{ 在交错树中且 } j \text{ 不在}\} \quad (3-44)$$

用以上朴素的算法实现, 时间复杂度为 $O(n^4)$, 空间复杂度为 $O(n^2)$ 。因为需要进行 $O(n)$ 次找增广路操作, 每次找增广路最多可能需要修改 $O(n)$ 次顶点标记。求标记改动值 d 时需要枚举所有边, 求 $X[i] + Y[j] - w[i, j]$ 的最小值, 该步骤的时间复杂度为 $O(n^2)$, 所以总时间复杂度为 $O(n^4)$ 。

这里的一个算法效率瓶颈是标记改动值 d 的求解, 所以, 考虑对朴素的 KM 算法进行。改进的算法中不用枚举所有边来求 d 。方法是: 对每个 B 集合中的顶点, 设置一个 $slack$ 值 (松弛变量)。在每次开始找增广路之前, 将该松弛变量数组的值初始化为正无穷。在找增广路, 检查边 (i, j) 时, 若其不在图 G 的相等子图中, 则 $slack[j]$ 的值更新为其本身与 $X[i] + Y[j] - w[i, j]$ 之中的较小值。在找完增广路, 没有得到完备匹配而需要修改顶点标记时, d 的值即为所有 $slack$ 值的最小值。这样的实现方式下, 时间复杂度可降至 $O(n^3)$, 空间复杂度仍为 $O(n^2)$ 。

4 实验结果与评估

对动物群体，如鸟群、昆虫群、鱼群等的社会行为的研究与探索，多年来一直吸引着多领域科学家们。。高速高清数码相机的发展以及计算机性能的大幅提升，使三维运动轨迹的跟踪和测量成为可能，但因为群体数量庞大、而且彼此相似，无法依靠外形特征分辨、以及动态跟踪时频繁出现的遮蔽问题，使得这一研究非常有挑战性。

果蝇（*Drosophila*）生命周期短、易繁殖，是生物科学研究中常用的一种模式生物，生物学家们经常用它们来研究生物遗传学和神经学相关课题。目前，科学家们已完成对其全基因组的序列分析，利用遗传学方法，已经可以对其任意一个基因进行操作^[2]。同时，果蝇作为一种有明显群体行为的生物，研究其群体的飞行行为将有利于揭示更多有关果蝇运动的神经机理，比如它们如何在群体情景下做出避碰的决策，其飞行行为如何受到性别比例、物种比例的影响等等^[2]。

本课题使用两台同步高速相机，对三维空间中飞行的果蝇群体进行拍摄，使用全局最优匹配算法进行轨迹的匹配和跟踪，并使用多视图几何知识进行果蝇轨迹的三维重构，在较低的时间复杂度下得到了较为准确的果蝇群体的三维轨迹。

本文在果蝇群体飞行实测拍摄之后，取其中 200 帧图像做进一步的轨迹跟踪与三维重构。实验中的一些参数设定如下：

- $\alpha - \beta$ 滤波器的参数： $\alpha = 0.8, \beta = 0.7$ 。
- 两视图中二维轨迹的立体匹配的迭代次数：6 次。因为实验发现迭代次数大于 6 次时，匹配结果并不会有明显改善。
- 轨迹的最小长度：6 帧。长度小于 6 帧的轨迹视为因目标检测错误、跟踪错误或匹配错误造成的实际不存在的非正确轨迹。
- 第??节中的轨迹重叠阈值 δ ：2 帧，即，互相可以连接的两条轨迹间重叠长度的容忍阈值 2 帧。
- 第??节中的匹配误差 ϵ ：5 个像素，表示立体匹配时果蝇个体 y 方向位置误差的容忍阈值为 5 个像素，两条轨迹间若有点对的 y 方向位置误差超过此阈值，则这两条轨迹视为不匹配。
- 第??节中的观察变量与预测的状态变量空间距离阈值 σ ：4 个像素，若观察变量与预测的状态变量的空间距离大于 4 个像素，则这二者视为不关联。数据关联阈值 T_ϵ ：6 帧，即，若连续 6 帧或以上某条标记为有效的轨迹没有成功关联到对应时刻的观察变量，则此条轨迹标记为已结束。

本文的果蝇群体跟踪算法使用 Matlab 编程实现，计算二分图最佳匹配部分使用 Matlab-C++ 联合编程，加快的程序的运行速度。

4.1 相机标定

在相机标定步骤中，一共拍摄了标定板 8 个不同角度的照片进行标定，所拍摄的标定图片如图??和??所示：

相机标定的具体实现采用加州理工大学的 Matlab 相机标定工具包^[2]。标定算法基于张正友的平面标定板的标定方法^[2]，标定物为印有黑白棋盘格的标定板，棋盘小黑白格的尺寸为 1.5cm*1.5cm。

经过相机标定，就得到了相机的内部参数 \mathbf{K} 和外部参数 \mathbf{R}, \mathbf{C} ，基于这些参数，就可进行果蝇飞行轨迹的三维重构。相机标定后的相机及标定板三维位置重现如图??。

图 4-1 左相机视图标定图像

图 4-2 右相机视图标定图像

图 4-3 两台相机及标定板的三维位置重现

4.2 目标检测

在图像的目标检测步骤中，拍摄的每幅图像先转换为灰度图像，之后进行反色变换（背景照片也做相同处理），然后所有这些图像都减去背景照片，以此方法剪除图像的背景。再对图像进行二值化处理，此时图像背景为黑色，一只只果蝇个体则是黑色背景中的一个小白点（图像分辨率为 1024*1024，帧率约 120fps，每只果蝇个体约占 10 至 20 个像素）。然后求得这一个个小白色区域的重心坐标，作为果蝇的二维图像坐标值。为了去除那些停栖在有机玻璃箱壁上静止不动的果蝇，这里增加一步操作，即，对于这些求得的果蝇二维坐标位置进行判断，若某一个点连续多帧都保持不变，则判定该果蝇没有运动中，并去掉该点。

目标检测结果为，左边的相机视图中平均每帧的图像中有 216.4372 只飞行的果蝇，右边的相机视图中平均每帧的图像中有 178.3920 只。

为了检验目标检测过程的准确性，在做完目标检测之后，随机挑选了其中 20 帧图像进行手工目标检测，查看检测出的果蝇个体数量，与用目标检测算法检测出的个体数量进行对比，发现误差在 1% 左右，这一误差主要由果蝇间的遮挡现象所导致。而在二维单视图跟踪的步骤中可以很好地处理遮挡问题，可以得到准确的果蝇群体飞行轨迹。因此，经检验，本文使用的目标检测可以得到较准确的果蝇二维坐标位置。

4.3 二维单视图多目标跟踪结果

二维单视图目标跟踪的结果为：左相机视图一共跟踪到 766 条轨迹，右相机视图一共跟踪到 706 条轨迹。

为了检验跟踪的准确性，在完成多目标跟踪后，随机抽取了其中 20 条轨迹，将其在每一帧的二维坐标点标注在拍摄的原图像中，人眼判断其准确性，发现跟踪结果无误。且当果蝇个体间非常靠近甚至在图像平面重合时，本文的算法仍能做到准确无误的跟踪。图??展示了两只果蝇互相靠近然后远离的过程，图中标记出的圆圈中，两只果蝇分别标记为 A 和 B，通过这几帧的单独检查，可以看出，它们的跟踪结果准确无误。最后，二维单视图多目标跟踪得到的两个相机视图中所有果蝇的二维轨迹（使用相机图像坐标）如图??。

图 4-4 果蝇个体遮挡发生时的准确跟踪

图 4-5 (a) 左相机视图中跟踪到的所有果蝇的二维轨迹；(b) 右相机视图中跟踪到的所有果蝇的二维轨迹

图 4-6 跟踪到的完整的果蝇群体三维轨迹

4.4 立体匹配 & 三维轨迹重建与重连接

进行立体匹配及三维轨迹的重构与重连接之后一共得到 794 条轨迹，其三维图像如图??。轨迹数目比左右两个相机视图中二维跟踪得到的轨迹数目略多的原因是，可能还存在一些轨迹，没有充分的进行重连接，仍然断裂为数条轨迹。

为下一步工作对果蝇的群体行为作进一步分析，在得到果蝇的三维运动轨迹之后，进一步计算每只飞行的果蝇在每帧的飞行速度，最后，将果蝇的三维坐标点描绘以不同颜色，代表不同的速度。这里，本文取了长度大于 20 帧的较长的轨迹，计算轨迹上每一点对应的当时果蝇的速度，结果如图??。

4.5 评估分析

利用本文提出的大规模果蝇群体跟踪的方法，可以高效、准确地得到三维空间中大规模果蝇的运动轨迹。该方法主要将大规模果蝇群体的轨迹跟踪与重建问题分解成为三

图 4-7 果蝇飞行的轨迹和对应时刻的速度值

个概率统计中的求最大后验概率问题，并转化为一个统一的数学框架，即，线性分配问题。使该问题很好地得到了解决。

在两维单视图目标跟踪步骤中，利用本文提出的观察变量与预测的状态变量之间的数据关联算法，可以保证果蝇个体跟踪的准确性。

在两个相机视图的二维轨迹的立体匹配步骤中，本文提出利用两条轨迹的最大共极线长度来度量两条轨迹的匹配程度，综合考虑了果蝇个体的运动信息和两条轨迹间的几何关系，在果蝇个体间互相非常相似，无法用外表及颜色、纹理特征进行分辨的情况下能得到可靠的轨迹匹配。同时，可以克服拍摄中因为果蝇个体间的互相遮挡造成的跟踪错误或者跟踪丢失的困难。

本文在进行立体匹配、轨迹三维重建之后，增加了一步操作，即三维轨迹的重连接。可以避免因二维单视图目标跟踪错误或者果蝇个体间的互相遮挡而致的轨迹中断为若干片段的问题，最终得到每只个体完整的飞行轨迹。

实验表明，该算法复杂度低、准确度高，并且有很好的扩展性，可以推广用于其他大规模群体的跟踪，如空中飞行的昆虫群、鸟群，水中游动的鱼群等等。另外，也可以使用更多台高速相机进行同步拍摄，这样可以进一步减少因为目标检测误差、个体间互相遮挡而导致的三维轨迹计算和匹配错误。使用更多台高速相机，在定标时通常用一台相机作为基准，其他若干台相机分别与之做一遍 Stereo Calibration。在重建时，每两个相机视图间做一次匹配，最后再将每次匹配得到的轨迹进行关联即可。

本算法的不足之处与改进的方向有：

- 有一些参数需要根据实际情况手工调整，而这些参数的选择会影响算法运行的实际效果。进一步的研究应包括怎样减少需要调整的参数数量或者怎样可以让参数做自适应调整。
- 需要一次性全部处理完所有图片，所以只能先做完全部实验，导出所有图像中再做处理，不能在实验时实时处理、实时跟踪。若以后的研究需要跟踪算法的实时性，则需要在该方面做进一步改进。
- 在二维单视图多目标跟踪时，假设果蝇做匀速飞行。事实上，在果蝇刚起飞、或者准备降落、或者变换方向时，速度是在不断改变的。如果换用更复杂更准确的模型可能可以进一步提高多目标跟踪的准确性。

5 总结与展望

长久以来,自然界中群体运动行为就备受多领域科学家的关注。所以,人类早在两千多年以前就开始记录、研究动物的群体运动行为。在这两千多年中,多领域、多学科的科学家们都在尝试探索动物群体运动背后的规律和意义。但由于科学技术水平和研究环境所限,并不能进行准确地定性分析。近年来,随着高速高清数码相机的发展以及计算机性能的大幅提升、计算机视觉领域的形成和相关技术的发展,使得准确、高效的三维运动轨迹的跟踪和测量成为可能。只有得到大规模运动群体中每只个体的准确地运动轨迹,物理学家及生物学家才能更深入地研究群体行为背后的原理及本质,如大规模群体的同步行为、个体间的避碰行为,个体间信息的传递等等。

虽然硬件技术已不断进步,但大规模群体运动的跟踪仍然极具挑战性。因为群体数量庞大、群体中各个体互相非常相似,所以如何进行目标检测和跟踪难度很大,目前较为成熟的依据形状、颜色等特征进行目标检测和跟踪的算法在这里都不能适用。另外,由于群体的数量和密度都非常大,从数百到成千上万只个体集中在很小的区域,所以拍摄过程中个体间的遮挡现象频繁发生,往往导致跟踪的中断或错误。

基于上述困难,本文提出了可以自动、准确、高效地进行大规模果蝇群体三维运动跟踪的算法,能很好的解决上述问题。本文提出的跟踪算法将大规模果蝇群体的三维运动的跟踪分解为七个步骤,即:相机标定、相机拍摄、图像预处理、目标检测、目标跟踪、立体匹配、三维轨迹重建与重连接。

在相机标定步骤中,本文使用张正友提出的标定方法,使用已知尺寸的黑白棋盘格标定板,多角度拍摄标定板进行定标。最后,选取了定标版 8 个不同角度下拍摄到的定标图像,使用加州理工大学开发的相机定标 Matlab 工具包,得到较为准确的相机的内部参数和外部参数。

在相机拍摄步骤中,本文使用多相机同步摄影测量实验平台。在正式拍摄之前每个相机先拍摄一张图像作为背景图片,方便后续的目标检测操作。然后,将果蝇放置于一个透明的立方体有机玻璃箱中,用两台高速相机在箱侧面从不同角度对果蝇进行同步拍摄,在箱后侧放置两个白色大功率无频闪光源,可以获得很好的拍摄效果。图像背景明亮干净,果蝇个体在图像上为黑色粒子状,后续操作可以方便地去除图像背景,得到果蝇个体二维坐标。

在图像预处理步骤中,本文挑选了其中 130 帧图像进行果蝇轨迹跟踪。

在目标检测步骤中,先将每张图片都减去背景图片,然后将图像做二值化处理,最后计算出图中每个小白色区域的重心坐标,作为果蝇的二维图像坐标。

目标跟踪问题的实质是一个概率推断问题,即,利用观察变量对状态变量进行推断。本文使用 $\alpha - \beta$ 滤波器进行状态更新。然后,将观察变量与预测的状态变量的数据关联问题建模为一个最大后验概率问题。再将其转化为一个线性分配问题,使用二分图最佳匹配问题的相关算法求解。

本文将两个相机视图间轨迹的立体匹配建模为另一最大后验概率问题,利用果蝇个

体的运动信息来帮助完成轨迹间的匹配,将立体匹配问题转化为一个基于两个相机视图轨迹间的最大共极线长度 MECL(maximum epipolar co-motion length)的二分图最佳匹配问题。充分考虑了轨迹间的几何约束和个体运动的约束,解决了个体大小、外形、颜色互相相似而造成的匹配困难。在匹配过程中,为了解决个体间遮挡造成的匹配错误和匹配丢失,算法实现中采用迭代求解,保证应匹配的轨迹尽量都匹配上,减少匹配丢失的发生。

最后,增加三维轨迹的重建与重连接的步骤,该步骤是为了解决由于二维单视图跟踪的错误或者果蝇个体间相互遮挡而造成的轨迹中断问题。在定义轨迹间距离之后,将三维轨迹的重连接转化为另一线性分配问题,同样适用 KM 算法(Kuhn-Munkras Algorithm)求解。

为了验证本文的跟踪算法的正确性,我们对三维空间中飞行的果蝇群体进行了实际测量和拍摄。大约拍摄了 200 余只果蝇的飞行轨迹,通过目标检测,左、右两个相机视图中平均每帧分别检测到 216.4372,178.3920 只果蝇个体。经手工抽样检测,准确率在 99%左右。在二维单视图多目标跟踪步骤中,左、右两个相机视图分别跟踪到 766,706 条轨迹。最后,经立体匹配、三维重建和三维轨迹重连接,一共得到 794 条完整的三维轨迹。

当然,本文的跟踪算法和相关工作还有很多不足之处,在下一步的工作中,将对以下几方面内容进行进一步探索和研究:

- 在目标检测步骤中,寻找更普适、更准确的物体检测算法。使其可以用于检测各种不同外形的物体,如水中游动的鱼、空中飞翔的鸟、微观世界中的细菌等等。对于这些外观形状更复杂的个体,还需要研究更有效、更准确的检测算法,同时应有更强的鲁棒性,可以克服图像的噪声。
- 在二维单视图多目标跟踪时,探索更准确、更有效的方法来估计更为复杂的运动,如突然变方向、突然变速度的运动等等。可以考虑如粒子滤波器之类的方法,当然,对于应用更复杂的方法而导致的算法时间、空间复杂度增加的问题,也应予考虑。可以考虑各种算法优化,降低复杂度,或者加入并行计算等思想加快运行速度。
- 在三维轨迹重连接步骤中,利用向前、向后预测的果蝇位置或者利用两条轨迹间重叠的几帧的信息来定义两条轨迹间的距离,使得一些因果蝇个体间遮挡或二维单视图跟踪错误导致的轨迹中断问题得到了解决。下一步的工作将考虑如何更好地定义轨迹间的距离,使得更多中断的轨迹能够重新连接。
- 本文使用两个高速相机组成的多相机同步测量平台,进一步的工作将考虑如何加入更多相机进行同步测量。使用更多相机同步拍摄,可以在发生遮挡或跟踪丢失,果蝇飞离某些相机的视野时,有更多相机拍摄到的图像做参考,可以减少果蝇跟踪的丢失,进一步减少个体间遮挡带来的跟踪错误。另外,使用更多相机时,还需要解决如何进行标定(标定板放置的角度,误差传递等),如何摆放光源、如何进行二维轨迹间的立体匹配、如何进行三维轨迹重连接等问题。

- 下一步工作将利用已获得的准确的果蝇群体三维轨迹进一步对果蝇的群体行为进行分析，例如果蝇飞行时的避碰机制等等。
- 今后的工作中，将探索如何将此算法用于跟踪其他大规模运动群体，例如跟踪水中的大规模鱼群、空中的大规模鸟群、围观世界的细胞群等。

希望通过自己在大规模群体三维运动相关方向的研究，可以对生物、物理等领域学者们的研究有所帮助。希望通过对大规模群体三维运动的研究发现更多大规模群体运动背后的机理，增进人类对自然的认识。

致谢

在此毕业论文即将完成之际，四年的大学生活也即将走到尽头，我即将离开母校，开始博士阶段新的人生与新的生活，心中无限感触，觉得自己的大学生活充满了幸运，充满了收获。我要感谢我的母校，在母校的四年中，我学到了很多，收获了很多，经历了四年充实的大学生活。

另外，在四年的求学生活中，我有幸得到众多师长、同学的帮助，在此，谨对他们表示衷心的感谢。

首先，非常感谢郑红老师，本论文是在郑红老师的指导下完成的，郑老师的耐心指导与热情帮助使我能顺利完成我的毕业设计。郑老师治学严谨、待人诚恳，本科学习生活中，在科研和论文等方面都得到了郑老师的指导和帮助，使我获益匪浅。在今后的学习、科研道路上，我将铭记恩师的教诲，争取更大的进步。

其次，要感谢复旦大学计算机学院图像图形与信号研究所的陈雁秋教授，陈老师是我今后五年直博阶段的导师。自从保研来到复旦大学计算机学院以来，在学习和生活中，陈老师都给予了我无私的关怀、指导与帮助，本论文的选题、实验以及论文的书写也都得到了陈老师的指导。陈老师治学严谨、思维活跃、待人宽厚、经常激发我们对身边科学现象的思考和探索，陈老师对科学研究锲而不舍的探索精神激励着实验室的所有同学。

感谢罗勇军老师，带领我们走进 ACM，安排我们一次又一次的比赛和集训。感谢苏纯洁老师，两次数学建模国赛、两次数学建模美赛都有幸得到您的指导，才取得了今天的成绩。感谢何高奇老师、冯翔老师、班导师朱瑛老师在本科生的学习、科研、生活中给予我的指导与帮助。此外，感谢每一位教导过我的老师，你们不但教给了我丰富的知识，更教会了我做人、做事的道理。

感谢 ACM 比赛一起拼搏的众队友：杜宇学长、王铮学长、田浩炜学长、陈月学姐、张越宇、陈光、林鹏、顾泽天，跟比赛中收获的乐趣与友谊相比，比赛成绩也就算不上什么了。也感谢华理 ACM 队的所有队员，这是一个志同道合，团结奋斗的集体，在 ACM 队的三年是大学中最美好最怀念的时光。

感谢历次数学建模比赛的队友们：姜叶飞学长、陈月学姐、张越宇、周筱晴、商晨菲、黄超。虽然除了 ACM 队员，其他的数模队友基本都只是在数模比赛的短短几天一起并肩作战，但那些通宵讨论、写论文的时光永远难忘。

感谢复旦大学计算机学院图像图形与信号研究所计算机视觉实验室的所有学长学姐们，从去年 12 月进实验室到现在，在学习、实验、论文等方面都得到了你们热情的帮助。你们孜孜不倦与执着努力的研究精神、团结一致的团队精神一直感染着我，今后的五年中，我将与大家一起努力，争取有更多一流的科研成果。

感谢带领我走进魔方世界的金晓波学长，没有大一那次精彩的魔方讲座，也许就不会有机会接触神奇的魔方世界。感谢华理魔方社的社友们：薛鸿翔学长、单晨曦、陈其超、蒋孝杰、任意等等。我永远不会忘记在食堂每一次快乐的魔聚，永远不会忘记我们一起参加过的每场比赛，永远不会忘记 UCUS 2011 赛季华理总积分第一那捧杯的时刻。

感谢我的室友们，多少日子，我踩着半夜 12 点的钟声匆匆回宿舍，多少日子，我大半夜开着电脑切题，感谢你们的理解与支持。

感谢计 086 班的所有同班同学，虽然大家相处的时间不多，但我会永远记得曾经属于这样一个团结和睦的集体。

最后，感谢我的家人，从多年求学生活，到如今继续选择科研之路，你们都给了我无限的理解、支持与鼓励。在我受到挫折沮丧时，你们成为了我避风的港湾；在我遇到困难时，你们无私的给予我帮助，用自己多年的人生经验给我提供宝贵的意见；经济上，你们成为我坚强的后盾，让我衣食无忧，专心学业；是你们让我可以自由地追逐梦想，祝你们身体健康，工作顺利！我一定不辜负你们的期望，尽自己最大努力，攀登科学高峰。