**第2章 真实世界坐标系下的地图生成与定位方法设计**

2.1 引言

基于SLAM方法的即时建图和定位方法目前已经发展到的相当成熟，尤其是在无人机、无人车领域，基于视觉SLAM方法进行定位有着积极地应用，对于实际落地的场景，该系统输出的相机位姿和构建的地图必须具备真实尺度才能够进行导航与定位，此外依靠视觉构建的地图所处坐标系往往取决于初始化成功后的第一帧多建立的坐标系，这些问题的存在都使得传统SLAM系统难以有实际的使用条件。

对于当前传统的单目视觉SLAM算法，设备简易，处理数据较少，满足实时性的要求，但是也存在只能获取相机坐标系下得到相机位姿以及无法获取地图实际尺度的问题，无法展开实际应用；对于基于双目视觉的SLAM算法，可以解决地图尺度的问题，但面临成本较高，相机标定难度大且精度不高，鲁棒性较差等问题；对于融合视觉和IMU传感器的SLAM系统，可以获取真实尺度，以及得到真实世界坐标系下的相机和地图位姿，但是该系统对于IMU的精度要求较高，且在引入IMU后容易产生累计误差，难以初始化成功等问题。

针对上述问题，本章提出一种融合视觉传感器和二维码标签的单目视觉SLAM方法，在传统SLAM功能的基础上，可以得到相机和地图的真实尺度，并且根据一定的坐标转换，可以得到真实世界坐标系下的相机位姿和地图。在真实的应用场景中，布置的二维码和普通的自然关键点相比，更加容易捕捉到关键点，此外，比较固定，在重定位的流程中，往往会有更好的效果。该系统能够具备以下优良特性：能够对地图进行保存，复用和更新，通过不断完善的先验地图提高系统的鲁棒性；添加二维码信息增强传统SLAM中的重定位问题，此外通过二维码的尺度在仅适用单目相机的情况下估算出真实地图的尺度；依靠二维码中坐标系和真实世界的坐标系的转化，获取待估计物体在真实世界下的位姿。本章所提出来的结合二维码的视觉SLAM算法流程如图2.1所示，主要过程包括二维码。

2.2 二维码检测与识别方法

2.2.1 ArUco二维码的构成特点

二维码和一般的自然关键点相比较，具备比周围环境亮度更低得到的显著特点，如图2.2所示；且二维码本身是一个四边形的区域，可以凭借该特点约束出后续相机的位姿的绝对尺度；并且每一个二维码通过解码都可以获取到一个独一无二的对应ID号序号，在SLAM重定位的过程中，可以避免高重合区域的误检，在通过视觉检测二维码得到的过程中，我们希望尽可能多的检出场景中的所有二维码，随后再通过编码对误检值进行剔除。

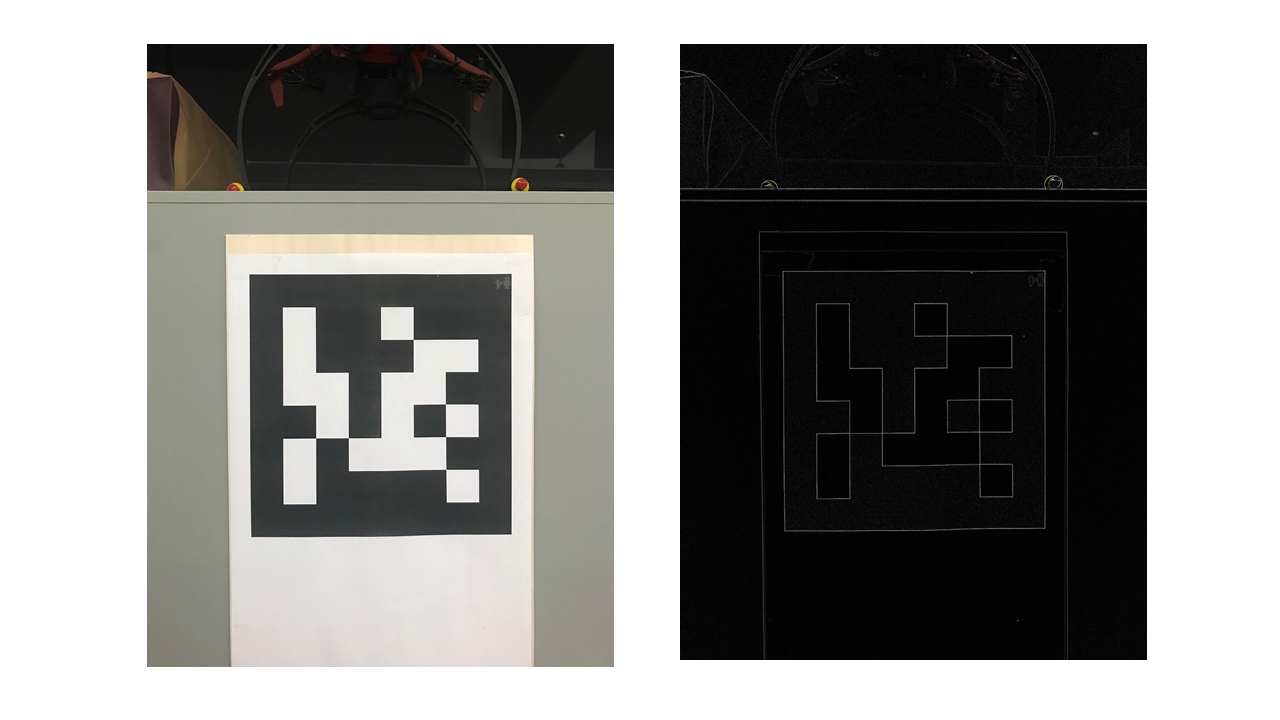


图2.2 ArUco二维码原始图像（左）和梯度图（右）

2.2.2 ArUco二维码的检测和识别过程

ArUco的检测和识别过程主要包括检测出二维码4个角点在图像中的位置，以及被检出的二维码的ID序号，在检测4个角点的位置时，需要检测图像中的线段和构成二维码的四边形。

在线段检测阶段，首先会计算整个图片中每一个像素的梯度强度大小和方向，随后对计算所得的梯度进行聚类，对所有满足聚类条件的像素点进行合并，可以的得到一组连续点，即检测出线段。

检测完线段后，进一步的需要检测构成二维码边缘的四边形，针对上一步中获取到的所有线段，对线段进行分组，若满足，上一线段的末端点和下一线段的起始点之间的距离小于某一阈值，即可首尾进行按照逆时针进行连接，若所有连接的线段数量达到4时，即认定生成的闭环可能为一个二维码的边缘四边形。

在检测二维码ID值之前，会先设定好一个包含所有二维码的字典，字典的大小即二维码的数量，字典中元素的大小即是每一个二维码的位数量。检测出构成二维码的四边形后，需要对图像进行透视变化规范图像，随后通过设定阈值分离出二维码上的黑色位和白色位，通过位数情况即可判断该二维码是否是字典外的不合规值，以及字典中的特定ID值，识别结果如图2.3所示。通过这样的方式检测和识别二维码具备非常好的鲁棒性，并且可以对错误值进行效验。

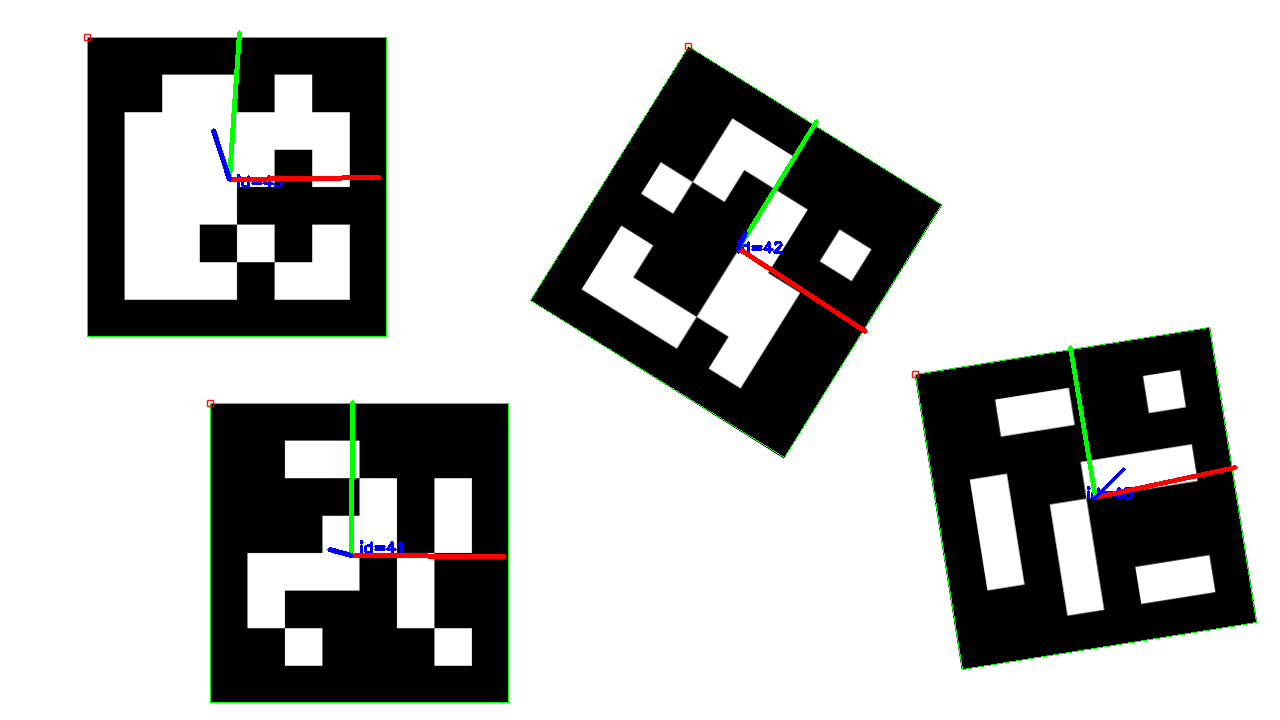


图2.3 ArUco二维码识别结果示意图

2.2.3 估计相机外参

对于传入的单帧图像，在提取完二维码的四边形轮廓，4个角点以及检测对应的唯一ID值后，接下来会估计检出的二维码位姿，包括平移向量和旋转向量，相机的外参主要是通过PnP方法，利用4对点来求解。

为了简化定位的过程，以二维码的坐标系作为世界坐标系，二维码边长为S，则四个角点的坐标分别为A（-S/2，S/2，0），B（S/2，S/2，0），C（S/2，-S/2，0），D（-S/2，-S/2，0），在图像中的对应像素点为a（）， b（）， c（）， d（）。因为相机的内参K提前标定，则三维空间中的点和像素坐标中的点之间的转换关系可以表示为：

通过以上公式，可以通过二维码到的四个角点求解出相机的旋转矩阵和平移矩阵。

2.3 基于二维码的SLAM系统描述

2.3.1 包含二维码的地图描述

在SLAM运行的过程中，可以生成一套地图系统，对于同一场景，该系统能够复用于后续的建图和定位过程，通过这样得到的方式可以添加对系统的约束，提高待估计量的精度。传统SLAM系统组成地图的元素主要包括关键点和关键帧，通过对关键点的描述符进行匹配，可以较好的的估计出相机的位姿。但关键点描述符的计算和匹配过程一般都比较耗时，而且对于重复场景，非常用于出现匹配错误的情况，考虑到这一情况，本文在此基础上，又添加二维码信息作为构成地图的元素，来进一步优化地图以获取更加准确得到相机位姿。本文中地图的构成包括以下3个集合：关键点集合p，关键帧集合f和二维码m集合，每个集合之间的数据元素相互耦合关联。

关键点集合p = {x,v,d},其中每个元素代表三维空间中的一个点，该点的描述包括在地图坐标系中的三维坐标x，观测方向v以及该点的描述符信息d，考虑到SLAM系统的实时性，选择BRIEF描述符来加快匹配过程。关键帧集合f = {T，&}，其中每一个关键帧包含一个外参矩阵矩阵T（旋转矩阵，平移矩阵），该外参矩阵是全局参考坐标系到相机参考坐标系的转化，&是相机的内参矩阵，该参数包括相机焦距，光学中心以及畸变参数，这些参数需要在运行SLAM系统之前就准确测量得到。二维码集合m={s,M,x1,x2,x3,x4}，其中的每一个二维码包括边长s（需要保证场景中的所有二维码的为正方形且所有二维码的尺度完全一致），二维码由其自身坐标系到全局坐标系的转换矩阵M，x1,x2,x3,x4分别代表二维码四个角点在其自身坐标系下的坐标位置。通过对包含二维码的自然场景进行离线建图，可以得到如图2.4所示的地图，其中正方形代表检出的二维码，蓝色矩形代表检出关键帧，红色点代表检出关键点。

其中每一类地图元素之间还具备一定的耦合关系，1）对于任意一个关键点，除了自身的属性外，还包括被观测帧序列，即所有可以观测到该关键点的关键帧的集合，以及在这些帧中出现的像素坐标位置；2）对于任意一个关键帧，还包与其关联的其他关键帧序列，以及在该关键帧中所有观测到的关键点和坐标，通过这些约束可以使得估计结果有更好的鲁棒性。

2.3.2 引入二维码的影响

SLAM是一个在导航的过程中，进行建图的工作，在使用单目相机时，即使目前表现较好的ORB-SLAM或者LDSO等算法也很难达到预期想要的效果，或者说存在很多的限制。第一个问题即是，所构成的地图的尺度是未知的，这样就造成无法在实际的导航任务中发挥作用，其次，在纯旋转的移动过程中，会导致算法失效，第三个就是单目视觉SLAM往往需要场景中存在比较丰富的材质便于跟踪，最后还存在一个问题，当场景中的视点发生变化，或者场景重复度较高时，在使用BOW词袋方法进行重定位时表现效果往往会很受到很大的限制。

除了用自然点来解决SLAM工作外，还有部分方法提出了仅仅使用人为设定的二维码来估计相机的位姿的方法，例如SPM-SLAM方法等，这种方式就可以较好的解决上述问题。因为二维码的放置不需要按照特定的规则，可以按照实验人员根据具体场景来布置，这样做就可以的得到一个由二维码构成的地图，并且，地图中的二维码都是具有尺度概念的。但是这样做存在的问题就是，我们需要在场景中布置足够多的二维码标志物，那么在大尺度的场景中，这样做就会提高实验的难度，因为在相机移动的过程中，要时刻保证至少有两张二维码出现在本帧中。

基于上述两种情况，本文提出了一种融合自然特征点和人工放置的二维码的SLAM方法来解决各自的问题和限制。首先，对于实际实验环境中，能够获取到真实的尺度概念，其次结合了自然点和二维码的SLAM方法也还可以可以只仅仅检测自然点，或者只检测二维码来进行位姿估计，因此对于大多数场景，都可以有更好的鲁棒性，第三个优点在于每一个二维码都有一个单独准确得到的ID值，那么在高度重复的环境中，也能保证匹配的正确性，最后对于大尺度的场景，可以结合自然点进行跟踪，二维码进行重定位来保证SLAM系统的长期稳定。

如图所示，对自然场景的局部区域进行建图，上图为自然区域的原图，在该区域中，有一部分的纹理较为丰富，其余部分则为贫纹理的白墙面，在白墙面上人为布置二维码标志，下图为根据该区域重建出的3D地图，由图可见，在丰富纹理的区域，自然点和人为布置的二维码都可以检出，但在贫纹理区域则，则基本只有二维码被检出，因此在贫纹理的区域就可以依靠二维码发挥作用。

2.3.3 基于二维码的SLAM过程描述

本文所提出的基于二维码的SLAM系统运行流程如图2.5所示，与一般常见SLAM系统相比较，本文主要提出了结合关键点和二维码标记进行结合使用。该SLAM系统的具体流程如下：

地图初始化：在SLAM中对地图的初始化可以获取相对尺度信息和参考坐标系的信息，主要是依靠多帧图像序列中关键帧或者二维码来完成，时在一般情况下初始化的流程为:首先选择第一和第二帧（f0，f1）运行两套初始化方法，如果任意一套方法初始化成功，那么就直接进入跟踪环节；如果两套方案都初始化成功，则优先考虑使用二维码的初始化方式；如果都没有成功，那么则固定第一帧，换取第三帧重复上述过程（f2，f3.。。）。

对于基于关键点的初始化方法，它能够并行的计算初始帧之间的单应性矩阵和本质矩阵的鲁棒性方法，其中单应性矩阵一般用于场景为平面时，其他场景则使用本质矩阵。在SLAM的过程里，只有检测到了足够大的视差时才会开始初始化，在单应矩阵和本质矩阵之间选择一个作为解算初始化的矩阵，由于为了尽可能使得解具有足够的鲁棒性，导致由关键点进行初始化时很困难的。并且最重要的一点是，从关键点进行地图的初始化，是无法获得尺度估计的，因此当在图像序列中一旦检测到二维码，系统就会对地图得到尺度进行调整以获得包含真实尺度的地图。

对于基于二维码的初始化方法，也是只要帧之间存在足足够的视差，就可以得到帧和二维码在地图中的位姿信息，但在某一帧中检测到多个二维码标志时，则可以通过选择在该帧中最小化投影误差最小的二维码来解算初始化信息。使用二维码来进行初始化的一个优点就是，地图中的参数都具备正确的尺度，因此就可以直接利用来为后续无人机的定位提供信息。

跟踪模式：如果在某一帧中估计出相机的位姿，则SLAM系统就会以前一帧作为起始点来估计当前帧的相机位姿，相机的位姿估计是将当前帧中观测到的一组地图三维点和二维码角点进行重投影误差最小化得到过程，因此在该过程中主要包括两个步骤：寻找地图点的对应关系和通过最小化投影误差获取相机位姿。

在寻找地图点对应关系环节，首先，查找在前一帧中观察到的地图点与参考关键帧之间的匹配，因为他们很有可能再次出现在当前帧。通过这些匹配可以解算出当前相机的初始估计；然后，通过将地图点投射到参考关键帧的相邻帧中来找到额外的匹配。对于每个地图点，首先需要检查其视角与相机方向之间的夹角是否小于阈值，若大于则丢弃该点，然后，检查相机与地图点之间的欧氏距离是否在关键点的尺度不变性区域内，随后，计算其投影，最后，将地图点的描述符与图像中搜索半径内的关键点进行比较。选择搜索半径中距离最小的两个关键点，且仅当描述符之间的距离比大于0.8,以及最小的描述符距离低于最低阈值τd, 则与匹配成功。一旦所有的匹配都被计算出来，重复的匹配就会被移除。最终，关键点结合和检测到的二维码标记获得最终的精确位姿。

对于相机的位姿估计，以

插入关键帧：关键帧只有在向系统添加新信息时才会添加到地图中，以便实现平稳可靠的跟踪。因此，必须设定一些规则来确定是否需要将当前帧添加至关键帧的集合。本文提出以下关键帧的提取规则：

1.如果当前帧至少有一个新的二维码标记出现(该标记没有出现在原本的地图中)，则将该二维码和该帧添加到地图中。二维码是第一次在该帧中被发现时添加的，因此也不能在帧中明确地估计该二维码的姿态。在这种情况下，标记的位姿被设置为无效，当观察标记的其他关键帧被添加到地图中时，将对其进行估计。2. 如果该帧包含至少一个二维码标记，且该二维码标记在地图中具有无效位姿，如规则1所描述，则该二维码的位姿可以在该帧中进行估计，则添加该帧并更新该二维码在地图中的位姿。

3.如果帧包含至少一个二维码标志,并且距当前帧最近的关键帧的距离大于一个阈值τb,该帧也会被添加进关键帧。

4 如果当前帧中匹配的映射点数量低于在参考关键帧中检测到的映射点总数的百分比τk，该帧也会被添加进关键帧。

第一次在一个帧中观察到一个新的标记时，它将与该标记一起作为关键帧添加。可能出现两种情况。首先，该标记被清楚地检测到，从而可以估计其位姿m m。这是一个初始的姿态，随着其他关键帧被添加到地图中，需要进行后续的改进(第4.4节)。第二种情况是标记被含糊地检测到。然后，添加标记，但其姿态设置为无效，直到可以从多个关键帧获得它。自一个新的关键帧添加每当τb的相机移动距离与其他关键帧(规则3教派的关键帧插入。4.3),更多的帧观察标记将被添加到地图。如果这些关键帧中的任何一个允许对标记位姿进行明确的检测，那么它将被注释。此外，当至少有两个关键帧模糊地观察标记时，其位姿与地图初始化时相同(4.1节)。

。前三条规则的目的是考虑到当前关键帧中的可见二维码标记来检查是否添加关键帧，由于这些规则而没有被认为是关键帧的帧则通过最后一条规则来添加。

本文中的SLAM系统再添加地图点是会遵循一种强鲁棒的策略，当一个新的关键帧被添加到地图时，系统有机会添加新的地图点，以及增加对现有地图点的支持。对于每个关键点，搜索参考关键帧的相邻关键帧之间可能的对应，由于关键帧的姿态是已知的，所以采用了外极限制来减少误检。一旦在地图上添加了一个点，将以一下侧率策略验证该点是否会被加入地图。基本思想是，该点必须至少在接下来的三分之二帧中可见，直到再向地图添加两个关键帧。

地图优化：每次添加新的关键帧或检测到闭环时，都需要更新地图，以便加入新的信息，并剔除不正确的地图点。需要优化的元素是关键帧位姿k T、地图点位置p x和二维码位姿m。全局优化是一个缓慢耗时的过程，一次再SLAM中一般会议多线程的方式另外处理地图管理。然而，由于它是一个稀疏问题，因此可以使用稀疏版本的Levenberg-Marquardt算法对其进行有效优化。只有在闭环结束之后才需要优化整个映射。当一个新的关键帧被添加到地图，只有地图关键帧共享点需要更新，因此整个优化问题都属于局部优化。

筛选关键帧：与地图点的保存策略相似，关键帧剔除策略避免了关键帧的无限增长。但是，必须同时处理关键点和二维码标记。其主要流程是，每个二维码标记和地图点应至少被三个关键帧所观测到，以便在进行优化过程时实现良好的三角化。对于每一个二维码标记m，从关键帧集合中找到可以观测到的关键帧，并筛选出三个和该关键帧距离最近的关键帧，这些关键帧不会被筛除。对于剩余的关键帧，在不会显著影响观察到的地图点的情况下会筛选掉，即某个关键帧中的地图点如果存在τc %的比例以上会被至少三个其他关键帧观测到，那么该帧会被被认为是冗余的，而使用其他三个关键帧代替。

回环检测和校准：在SLAM的运行过程中，当相机机在环境中移动时，不可避免地会产生漂移，必须对其进行检测和校正。在系统中，回环检测必须区分关键点和二维码标记，使用关键点进行闭环检测时，一旦添加一个新的关键帧时，系统会另开辟一个线程进入地图管理模式，因为SLAM系统会一直保留一个BoW数据库以检测是否插入关键帧，具体策略为: 测量最近添加的关键帧与图G中相邻关键帧之间的相似性，保留最低的s min，然后对数据库进行查询，得到最相似的非邻居关键帧，相似度大于s min。对于符合闭环的每个候选项，将使用插入的关键帧计算其关键点对应关系。若对应数量足够大，则利用对应地图点的三维位置，利用RANSACPnP方法，求出它们之间变换矩阵T 。如果转换的内置点的数量足够高，则认为回环检测成功，随后进行校正。

另一方面，在处理任何其他帧之前，必须立即检测由二维码标记引起的闭环，如果在跟踪之前不纠正回环漂移，会导致相机的跟踪问题，因为未纠正的帧位姿可能与二维码所计算出来的位姿差异较大。例如在某一个场景中，系统从一个二维码标记开始初始化，相机在场景中移动并回到最初的起始位置，在这个过程中，会不断地叠加漂移误差，当相机第二次到达初始位置并再次观察二维码标记时，估计的相机姿态与实际会较大，当使用二维码标记或关键点检测到循环闭合时，必须纠正漂移。

重定位：如果在跟踪的过程中失败们则必须进行重定位，重定位也是首先考虑使用二维码来完成，如果在某一帧中明确检测到二维码标记，则利用该二维码来重新计算位姿，随后，通过查找地图对应关系和重新估计相机姿态来优化位姿。如果当前帧位姿不能从二维码标记中准确地估计出来，就使用关键帧结合数据库进行查询，查询与当前帧相似的关键帧，并分析最佳关键帧。在回环检测出中，将当前帧的关键帧与所选关键帧进行匹配，并采用RANSACPnP法估计相机的位姿T,如果内置点的数量足够高，则认为重新定位成功，系统重新进入跟踪模式。

2.4 坐标系统转换方法

2.4.1 坐标变换的数学描述

在SLAM系统中，常用的坐标系存在2个概念，分别是相机坐标系和世界坐标系。其中世界坐标系是我们最需要考虑的一个坐标系，世界坐标系代表的是真实世界下，物体所参考的坐标系，一般都是固定不动的，可以给运动的物体提供控制方法，对于世界中的任意一点，可以描述为P（（Xw,Yw,Zw）。相机坐标系代表的是移动的物体自身的坐标系，由于相机是刚性物体，一般以相机的坐标为准，对于相机中的任何一点，可以描述为P（Xc,Yc,Zc)。

因为SLAM系统中会涉及到多个坐标系，不同的坐标系之间需要进行转化，空间中点的坐标系变换是一个欧式变换，一般由旋转加平移构成。首先只考虑坐标的纯旋转问题，即只有旋转，没有平移产生，两个坐标系的基底分别为（e1，e2，e3），（e1，e2，e3），对应的坐标分别(a1,a2,a3),(a1,a2,a3)。则有：

其中旋转矩阵R由两组基之间的内积组成，表示坐标系旋转前后同一个向量的坐标变换关系。旋转矩阵是一个行列式为1的正交矩阵：满足 特点，则有

a = Ra

在欧式变换中，除了旋转还有平移：

a=Ra+t

可以用旋转矩阵R和平移向量t 就能描述空间中的坐标变换关系,在三维向量的末尾添加1，称为齐次坐标。矩阵T同时包含了旋转和平移，称为变换矩阵。

坐标系代表的是移动的物体自身的坐标系，由于相机是刚性物体，一般以相机的坐标为准，对于相机中的任何一点，可以描述为P（Xc,Yc,Zc)。

2.4.2 获取真实世界坐标系方法

在SLAM系统中所描述的定位信息都是基于地图坐标系或者相机坐标系，但是在实际的应用中，就需要将所有的定位信息全部转换至真实世界坐标系，才能为物体后续的移动提供控制方法。我们通过传统的SLAM方法，只能够得相机在地图坐标系中的位姿，现得益于二维码自带坐标系的特性，可以经过一定的转换可以获取相机在真实世界坐标系下的位姿，可以为后续控制提供真实的位姿信息。

首先，声明所有可以得到的坐标信息，可以通过SLAM系统得到，二维码在SLAM地图坐标系下的位姿Tmarker（camera），相机在SLAM地图坐标系下得到位姿Tcamera（camera），二维码在真实世界坐标系的位姿Tmarker（world）。经过坐标变化，可以的得到相机在二维码下的位姿，Tcamera（marker） = Tmarker（camera）（-1）Tcamera（camera），真实世界坐标系在二维码坐标系下的位姿Tworld（marker）= Tmarker（world）（-1），那么相机在真实世界坐标系下的位姿Tcamera（world） = Tworld（marker）（-1）Tcamera（marker）

= Tmarker（world）Tmarker（camera）（-1）Tcamera（camera）

其中，二维码在真实世界坐标系的位姿可以按照实际场景进行设置，为了计算方便，直接设定第一个二维码的中心点为世界坐标系三维坐标的中心点，且x轴朝右，y轴朝上，z轴垂直于二维码平面向外，如图所示，那么第一个二维码的在世界坐标系下的坐标为（0,0,0），其余二维码则可以按照真实尺度获取。

在SLAM的实际运行前，应该保证二维码的实际布置坐标和设定的二维码在实际世界坐标系下的坐标一致，避免SLAM过程中位姿的抖动问题，先验地图可以设定以配置文件的形式记录，这样可以针对不同的场景和不同的二维码布置情况，直接修改配置文件即可。

在SLAM的运行过程中，应该尽可能保证每一帧都可以看到至少一个二维码，，对于观测不到二维码的图像帧，SLAM算法将利用特征点匹配进行定位，当观测的图像帧中有多个二维码时，那么就需要考虑到多个二维码的协同影响，处理方式有如下几种形式：

1 提前选择某一个参考二维码，无论相机观测到哪一个二维码，都根据观测到的二维码和参考二维码之间的位姿关系，求解相机在世界坐标系下的位姿。

2.对于观测到的多个二维码标记，根据上一帧计算出来的位姿，可以计算出当前帧距离最近的一个二维码，以该二维码作为参考，计算出此时相机在世界坐标系下的位姿。

3. 对于观测到的多个二维码标记，分别结合各个二维码计算出此时相机在世界坐标系下的位姿，对于所有位姿，求取平均值即可。

以上三种方法都可以解算出相机在真实世界坐标系下的位姿，其中方法1）的计算过程最为简略，并且全程只利用一个参考二维码可以避免在频繁更换参考二维码的过程中造成的位姿抖动问题，但是会存在线性放大相机位姿误差的情况，该方法难以对误差进行优化。方法2）会应用到历史位姿信息，可以有效降低相机误差，当时会频繁更换参考二维码，这样也会造成相机位姿估计过程的抖动问题。方法3）和方法2）类似，在处理误差时会结合多个参考二维码的信息，使得估计位姿的系统鲁棒性提升。针对不同的场景，需要结合具体的方法，对于二维码远距离线性摆放的场景，应该避免使用方法1），有限考虑方法2）和方法3）；对于实时性要求较高的场景，则优先考虑方法1）。

2.4.3 完整流程

根据以上章节，可以确定出在真实世界下相机的位姿结果，在具体的工程实践中，除了结合二维码的SLAM算法的研究，还需要考虑到先验地图的设定，数据的传输，相机的设定等问题，具体流程。、

地图设定：在无人估计位姿时，如果有先验地图的加入，则可以添加约束进一步优化位姿结果。在载入地图的过程中，如果该场景是第一次测量，没有先验的地图信息，则创建一个空的global\_map来存储地图信息，并将Map\_Flag字段设定为NotFirst\_Map；反之，则直接载入历史先验地图。在整个定位流程结束后，将更新后的地图保存在global\_map中。

载入先验二维码地图：根据实际世界坐标系中各个二维码的位姿情况，将其记录在配置文件中，包括每个二维码的实际尺度，ID值以及位姿情况。

设定SLAM参数：对于结合二维码的SLAM系统，需要对系统中的参数进行设定，主要包括SLAM的运行模式（SLAM模式，重定位模式），特征点的提取方式（orb、brief、SIFT等），二维码的检测类别（ARUCO\_MIP\_36h12、），关键帧筛选阈值，线程的数量，是否复用地图等等，这些SLAM相关的参数都需要在运行前设定。

设定相机参数：相机的参数一般包括相机内参，以及曝光，快门时间和增益等，对于相机内参需要提前进行标定，后面三个参数需要根据实际场景调试后确定出最佳数值。

设定数据传输系统：在工程实践中，一般都采用ROS（机器人操作系统）来对数据进行传输，其用于编写机器人软件程序的一种具备高灵活轻度的软件架构，可以使得数据在多个进程中进行传输，本文主要涉及SLAM进程和无人机飞控进程，这两个进程分别对同一个节点进行数据得到发布和订阅以完车数据传输。

2.5 本章小结

\begin{table}[h]

\centering

\caption{PointGrey相机参数}

\label{tab:chap1:marker\_pose}

\begin{tabular}{C{3.6cm}C{6.4cm}{C{3.6cm}C{6.4cm}}

\toprule

\textbf{参数} & \textbf{数值} & \textbf{参数} & \textbf{数值} \\

\midrule

Frame Rate & 30 FPS & Megapixels & 1.3 MP \\

Lens Mount & CS-mount & Readout Method & Global shutter \\

Pixel Size & 3.75 µm & Sensor Name & Sony ICX445 \\

Resolution & 1288 × 964 & Required Accessories & Lens, Cable, Host Adapter (USB 3.1 Gen 1), Tripod Mount Adapter (ACC-01-0003) \\

Sensor Type & CCD & Part Number & CM3-U3-13S2C-CS \\

Sensor Format & 1/3" & Chroma & Color \\

ADC & 12-bit \\

\bottomrule

\end{tabular}

\end{table}