

#注：删除部分无关内容。本文档属于论文，不适用声称的开源协议。不含源代码和实现细节。

多传感器拼接全景图像中目标跟踪定位技术研究

北京理工大学

摘要

在全景图像中进行目标的跟踪定位是当前图像处理领域中的热门研究方向。相对于传统的二维平面图像，全景图像包含更多细节、覆盖更大的视场，因此有着诱人的发展前景。先利用多个传感器采集多路图像，然后利用图像拼接技术合成全景图像，是得到全景图像的方法之一。本文即是以一台集成八个摄像头的全景相机作为实验设备，来采集图像并生成全景图。同时，本文在建立了全景相机物理模型的基础上，进行了对全景图像中目标的检测、跟踪和定位方法的研究。

本文利用 `opencv` 平台进行程序编写和算法的实现。我们采用先在单路视频中目标跟踪定位，再变换为全景视觉下的方位角和俯仰角信息的方法，来实现多传感器拼接全景图像中目标的跟踪定位。本文结合实验中全景相机的物理模型，对各路视频的处理顺序进行了设计，减少了程序的计算量。最终，在较低的性能开销下实现了全景图像中目标的跟踪和定位。

关键词：全景相机；全景图像；多传感器；目标跟踪定位

Abstract

Target tracking and localization in panoramic images is a hot research area in the field of image processing. Compared with the traditional two-dimensional image, panoramic image contains more information and covers wide-field-of-view, so it has attractive prospects for development. Using many image sensors to collect images, then we mosaic and synthetic panoramic image is one of the main methods of panoramic image acquisition. We use a panoramic camera which eight cameras as the experimental equipment to collect images. At the same time, based on the geometric model of panoramic camera, the target detection, tracking and localization method of panoramic image are studied.

This paper uses opencv as the platform for programming and to implement the algorithm. We use the method of tracking and locating the target in a single channel video, and then transform it into the coordinate position of the panoramic vision, in the end we achieve the goal of tracking and localization the target in multi sensor mosaic panoramic image. Based on the geometric model of panoramic camera in experiment, the processing sequence of different video sequences is designed, and the amount of computation is reduced. Finally, the target tracking and localization in panoramic images are achieved with lower performance overhead.

Keywords: Panoramic camera; Panoramic image; Multi-sensor; Target tracking and positioning

目录

摘要	I
Abstract	II
第 1 章 绪论	1
1.1 研究背景、目的和意义	1
1.2 国内外研究现状	1
1.3 研究内容与论文结构	3
1.3.1 问题提出和研究的主要内容	3
1.3.2 论文结构与章节安排	3
第 2 章 多传感器全景相机模型的构建	5
2.1 全景图像的发展历程	5
2.2 全景图像采集系统的种类和特点	5
2.2.1 利用鱼眼镜头采集全景图像	6
2.2.2 利用全景云台采集全景图像	6
2.2.3 利用多传感器全景相机采集全景图像	7
2.3 全景图像投影模型	8
2.3.1 全景图像投影模型种类	8
2.4 单路图像坐标与空间坐标的映射关系	10
2.4.1 实验设备几何结构	10
2.4.2 单路图像的平面坐标到球面坐标的映射	11
2.5 本章小结	12
第 3 章 目标检测与跟踪定位	13
3.1 单路图像运动目标检测	13
3.1.1 运动目标检测理论	13
3.1.2 使用帧间差分法进行运动检测	16
3.2 单路图像目标的跟踪与识别	18
3.2.1 模板匹配算法简介	19
3.2.2 利用模板匹配法进行目标跟踪	21
3.3 全景视觉下进行目标跟踪的方法	22

3.4 全景视觉下的目标定位	25
3.5 本章小结	27
第 4 章 系统实现与评价	28
4.1 多传感器图像的采集	28
4.2 图像处理与数据输出	29
4.2.1 图像处理流程	29
4.2.2 数据输出	31
4.3 本章小结	32
第 5 章 总结与展望	33
5.1 本文工作总结	33
5.2 未来工作展望	33
致谢	错误!未定义书签。
参考文献	35

第 1 章 绪论

1.1 研究背景、目的和意义

在机器视觉领域，运动目标检测以及跟踪一直是重点研究方向，发展到今天，出现了各种各样的检测和跟踪算法。对于跟踪算法来说，需要解决的关键问题是如何获取目标的空间坐标信息。随着时代的发展，越来越多的应用场景需要一种能够在大视场、全方位图像进行目标检测跟踪的算法。利用全景图像进行目标检测跟踪，可以满足上述要求。全景图像的目标跟踪方法不同于传统跟踪方法。特别是对于多传感器拼接成的全景图像来说，进行目标跟踪时，还要考虑全景相机的几何参数和光学参数带来的影响。

监控领域是运动目标检测跟踪算法应用最早、也是最成熟的领域之一。近年来，越来越多的应用从传统的单目监控方案过渡到双目监控方案，因为与单目监控系统相比，双目监控系统能够提供更大的视野，对于被目标跟踪系统而言，双目系统也能提供更多的信心，如速度和距离。目前双目监控系统被广泛应用于交通测速等领域。但是双目系统仍然具有一定的局限性，无法做到 360° 方位角覆盖，仍然存在丢失目标的情况，对于复杂运动、图像遮挡等问题，也不能很好的解决。而利用多传感器采集图像，再对其进行拼接合成，最终得到的全景图像，可以满足全方位的监控需求。如何利用拼接合成的全景图像进行目标检测和跟踪，成为新的研究方向。近年来很多研究人员已经进行了许多研究，但是一些问题仍然没有得到很好的解决。例如，由于全景图像信息量大、分辨率高，现有的很多算法处理的速度较慢，难以满足实时性跟踪的要求。另外，利用多传感器采集图像并合成全景图像，对于视频采集设备、图像处理设备的硬件性能要求也比较高。

本文将在传统目标跟踪定位算法的基础上，利用实验室一套多传感器全景相机采集图像并拼接合成全景图像，然后进行全景图像目标检测和跟踪定位算法的研究。

1.2 国内外研究现状

在全景图像中进行目标跟踪定位离不开传统的跟踪定位技术。传统目标跟踪定位技术的第一步是进行运动目标的检测识别，运动检测的方法有很多，经典，同时也是最常用的方法是背景减除法^[1]。与这种方法类似的是临帧差法^[2]。但是这些方法处理多目标图像时效果较差。1981年，Horn 和 Schunck 首次提出使用光流法来描述运动目标^[3]，这种方法将多个目标独立开来，也不需要预知图像背景等信息，适用面很广。但是这种方法计算量比较大，处理起来比较耗时，无法满足实时性要求。但是随着近年来计算机处理速度的提高，光流法有着较好的发展前景。根据背景与摄像机的相对运动关系，运动检测分为静态背景和运动背景两种场景。静态背景指的是摄像机相对于背景静止，图像中只有目标在运动^[4]；运动背景指的自然是摄像机相对于背景是运动的，这时目标相对于背景和摄像机都是运动的，检测和跟踪的难度变大。一般来说，背景差法难以解决背景运动场景下的目标检测跟踪问题^[5]，但是实际应用场合大多数静止背景的应用场景，所以背景差法在目标跟踪检测中应用很广泛。临帧差法虽然精度不高，但由于算法简单高效，应用也十分普遍^[6]。

目标跟踪的方法有很多种，可以简单归纳为三类：基于目标特征、基于模型、基于区域^[7]。本文使用的模板匹配法即是一种基于区域的跟踪方法，通过比较目标图像和搜索区域的相关关系。在识别到目标之后，需要对目标进行提取，提取其图像用以继续匹配跟踪，提取其坐标用以进行定位。

针对全景图像的运动目标跟踪监测是近年来的研究热点之一，得益于全景技术的普及，生产和生活中对相关技术的雪球越来越多。国外的众多公司及机构如 IBM、MIT、微软公司、Google 公司、CMU 等，都在全景图像处理方面投入了大量资源。国内全景图图像处理领域比较领先的，有中科院自动化研所、中科院上海微系统与信息所等。处理的方法主要有两种，一种是直接对拼接后的全景图像进行处理^[8]，另一种是对多个摄像头采集得到的图像分别进行处理，将坐标等信息经过变换，得到全景信息^[9]。目前，制约全景图像处理算法发展的问题主要仍然是算法复杂，导致执行的速度不高。

上海交大以及加州大学伯克利分校的周明辉、陈思聪等人，介绍了一种将全景图展开为柱面并进行实时目标跟踪的方法，能够比较快速的在全景图像中进行目标跟踪，同时也提供了一种全景图像展开和坐标映射的方法^[10]。西安理工大学的邓颖娜、朱虹等人，针对多传感器全景视图中各路图像的重叠问题

做了一些研究，解决了重叠区域给目标跟踪带来的影响^[11]。这些研究人员的工作给本文的研究带来了极大的帮助。

1.3 研究内容与论文结构

1.3.1 问题提出和研究的主要内容

目标跟踪定位算法主要有两部分构成，第一步是进行目标检测，第二步是提取出目标的坐标信息等，进行持续跟踪。运动目标的检测方法已经很成熟了，比较流行的是采用帧差法。目标跟踪定位方法也有很多，但是对于不同的图像采集设备来说，适用的跟踪定位方法往往也是不一样的。全景图像采集设备的原理和结构与传统摄像头相比已经有很大的不同，跟踪定位的方法也有所不同。这是本文需要研究的一个问题。

本文需要研究的另一个问题是全景图像目标检测、目标跟踪定位算法的执行效率问题。全景图像由于其像素高等特点，导致很多算法执行速度慢，难以实现实时跟踪的要求。本文希望通过实验，选出一种执行速度较快、处理结果较理想的算法。

本文首先将分析实验中使用的多传感器全景图像采集设备，建立全景投影模型，分析全景图像空间坐标的映射关系，并分析全景图像跟踪定位算法的难点。然后，对单路图像进行目标检测和目标跟踪算法进行研究和实现。最终选择合适的算法结合实验设备进行多传感器拼接全景图像的跟踪定位技术研究，并对其效果进行分析评价。

1.3.2 论文结构与章节安排

本文分为五个章节。

第一章为绪论，在这一部分，介绍在全景图像中进行目标跟踪定位的研究背景及意义，以及国内外的一些研究成果和研究现状，同时概括了本文的结构。

第二章主要介绍实验所用全景相机的物理模型，以及拼接后全景图像的展示模型。这一部分，我们也分析了不同种类全景相机以及不同的全景图像模型

的原理、特点以及优缺点。最终针对本实验所用的设备进行具体的分析。

第三章详细介绍了全景视觉下目标跟踪和定位算法的原理和实现方法。首先我们在单路视频图像中进行目标检测、跟踪算法的研究和检验。尝试使用帧差法作为目标检测方法，然后使用模板匹配法进行目标跟踪，在目标无遮挡条件下效果良好。接着，我们对多传感器全景协同的目标检测跟踪算法进行了研究和实现。使用先单路检测跟踪、把坐标转换为全景视觉下坐标的方案。在这一部分，我们也结合第二章中建立的全景相机几何模型，推导出从单路图像平面坐标到空间方位角的计算方法，可以确定出全景图像中目标的坐标方位。第四章介绍本文目标跟踪定位系统的实现。首先结合硬件设备，研究了从全景相机提取单路视频，并保证各路视频间的同步性、实时性的方法。然后介绍图像处理部分的算法流程和跟踪定位结果的输出。

第五章对本文的研究结果进行了总结和评价。在最后，分析了本文算法的一些不足之处，并对未来的优化工作做了展望。

第 2 章 多传感器全景相机模型的构建

全景图像的采集方式有很多种，根据采集原理和采集设备的不同，分为三类：利用鱼眼镜头采集全景图像、利用全景云台采集全景图像、利用多传感器拼接合成全景图像。不同的采集方式适用于不同应用领域。在以上三种采集方式中，多传感器拼接合成全景图像的方式由于其分辨率高、细节丰富、实时性强的特点，比较适合进行全景图像运动目标检测和跟踪定位算法的研究。

全景图像展示方式主要有三种，按照投影模型来分，分别为圆柱形投影、正方体投影和球体投影。球体投影比较适合描述目标的空间位置坐标关系，符合人类对空间位置的直观认知，因此我们选择球面模型作为全景图像展示模型。

下面我们将分析评价不同的全景采集方式和全景图像模型，选择适合实验设备的全景模型并建立起从全景图像到空间坐标的对应关系，将单路图像的坐标映射到空间方位角中。

2.1 全景图像的发展历程

生活中我们所接触到的图像大多是记录二维信息的平面图像，例如传统的画像、相片等。全景图像本质上仍然是平面图像，但是可以通过一定的方法展示和记录三维信息，能使人感受到立体的感觉^[12]。

最早的全景相机出现于 19 世纪，Cirkut No.10 相机采用一个扇状物控制的时钟机械装置，在 10 英寸的相纸上记录了 360° 的图像，该相机因此称为相机史上的传奇之一^[13]。20 世纪出现了很多经典的全景相机设计。有利用宽幅相机底片拍摄以获得更大视场的“宽幅全景机”，也有“摇头机”、“转机”等新奇的设计。“转机”即是通过在水平方向上旋转相机，以此拍摄 360° 视场的图像。这种拍摄方式与今天通过全景云台拍摄全景图片的方法非常类似。直到今天，“转机”还经常被应用于集体照的拍摄。中国的“海鸥”牌“转机”就曾在 90 年代风靡全国^[12]。

2.2 全景图像采集系统的种类和特点

2.1.1 利用鱼眼镜头采集全景图像

我们把视角接近 180° 的镜头称为鱼眼镜头。使用鱼眼镜头进行成像，可以一次性在一张图像内收录收录较广的视角。有很多全景相机利用鱼眼镜头的这一特性进行全景图像的拍摄，近年来比较流行的有 理光 THETA，该相机在机身前后各放置一枚鱼眼镜头，一次性拍摄前方和后方 180° 视场图像，最后合成全景图像。

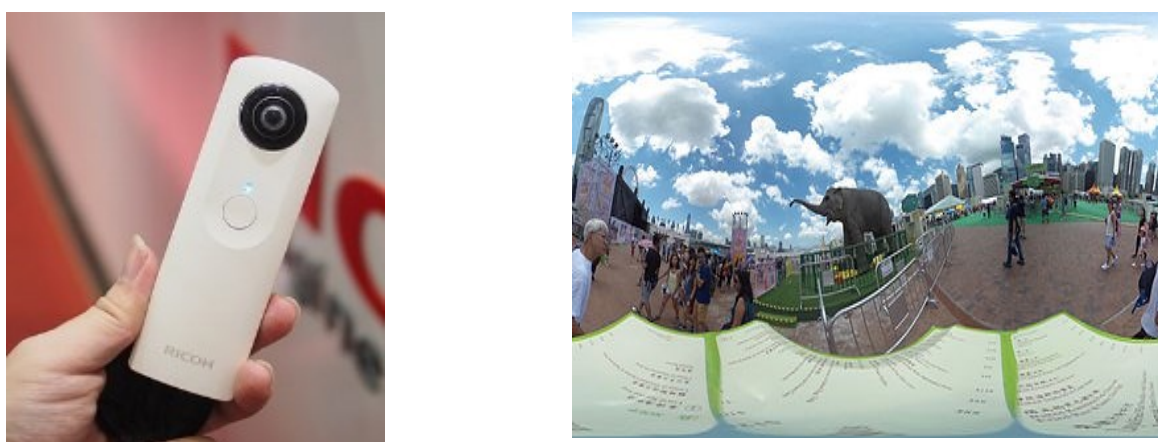


图 2-1 理光 THETA 全景相机及其所成的全景图像

利用鱼眼镜头可以通过较少的相机一次性得到全景图像，效率比较高。但是缺点也是显而易见的。用这种方式得到的全景图像畸变难以矫正，目标跟踪定位的结果精确度很难保证。这种全景图像采集方式由于成本较低，被广泛应用于民用摄像领域，以及一些监控领域。

2.1.2 利用全景云台采集全景图像

在摄影行业中，为了保证拍摄过程中相机的稳定性和保持特定的拍摄角度，往往需要三脚架和云台对相机进行支撑和固定。云台是安装在三角架或其他固定设备上的一个转向轴，可以通过它对相机姿态进行调整。全景云台是云台的一个种类，通过内部的机械设备或机电一体化设备，绕着云台中心轴在水平或竖直方向旋转，拍摄多张图像，最终合成全景图像。



图 2-2 思锐 PB-10 三轴全景云台及其三个旋转轴

利用全景云台采集全景图像，只需要一台相机，成本比较低，因此在摄影领域被广泛应用。但是，这种方法得到的一般为单幅图片，难以采集动态视频，因此不适宜于进行目标检测和跟踪定位研究。

2.1.3 利用多传感器全景相机采集全景图像

多摄像头全景采集设备由若干个按一定规律排列的相机组成。该设备工作时，多个相机同时采集图像，然后按照相机摆放的空间位置关系，对一组图像进行拼接，最终可以合成全景图像。



图 2-3 海康威视 DS-2DP1636-D 多传感器全景监控摄像头

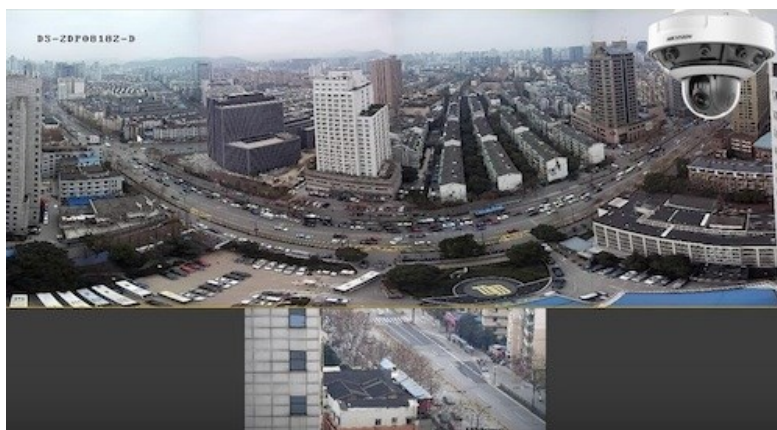


图 2-4 海康威视 DS-2DP1636-D 成像效果

这种全景图像采集系统可以实现 360° 无死角拍摄，同时分辨率非常高，细节丰富。由于采取了多组摄像头同时采集图像的工作方式，实时性良好，可以实时进行全方位的目标检测和跟踪。目前这种方法被应用于高端安防领域、虚拟现实图像采集领域、汽车无人驾驶技术研究、地图测绘领域等。

2.3 全景图像投影模型

2.3.1 全景图像投影模型种类

我们按照全景图像的展示形式来对其进行分类。实际上，我们最终所看到的全景图像，是全景拼接图像投射到不同模型上的结果，目前主流的投影模型有三种，分别是正方体模型、圆柱体模型和球体模型。需要注意的是，利用超广角镜头得到的图像效果与球体模型投影得到的图像近似，旋转云台拍摄的全景图像与圆柱形投影的效果也近似^[14]。

(1) 立方体投影

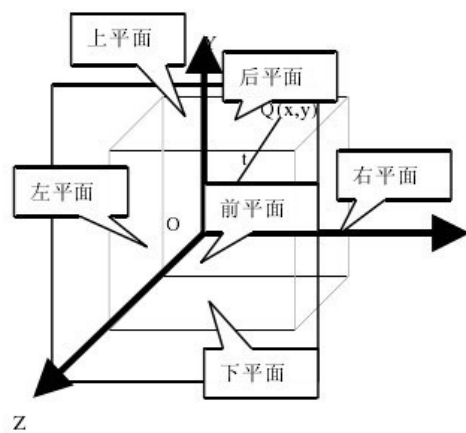


图 2-5 立方体投影

将全景图像投射到立方体的六个面上，观察者在立方体的中心进行观察。这种方法使得观察者可以在水平的 360° 和垂直的 180° 方向上查看全景图像。

(2) 圆柱面投影

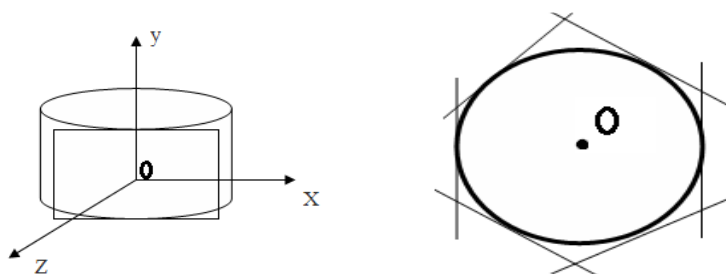


图 2-6 圆柱面投影侧视图（左）和俯视图（右）

这种方法是将 360° 拍摄的若干张图像投影到圆柱的侧面上，观察者在中心可以查看 360° 的水平方向上的信息。

柱面模型变换关系相对来说比较简单，另外视觉效果也比较好，应用比较广泛。

(3) 球面模型投影

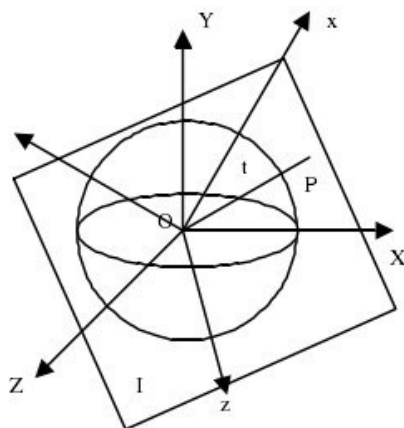


图 2-7 球面模型示意图

这种方法将全景图像投射到空间球面中，观察位置位于球心。观察者可以观察空间任意角度图像。这种方法可以很方面直观的得到目标的真实方位信息，符合人类对于空间的认知，因此在全景图像的展示中被广泛应用。

2.4 单路图像坐标与空间坐标的映射关系

2.4.1 实验设备几何结构



图 2-8 环视众邦全景相机外观

实验所用图像采集设备为环视众邦公司设计制造的多传感器全景监控摄像头，如图 2-8 所示。该摄像头由 8 个相同的摄像头组合而成。其中，中心摄像头相面与水平面平行；沿着中心摄像头均匀分布 7 颗摄像头。

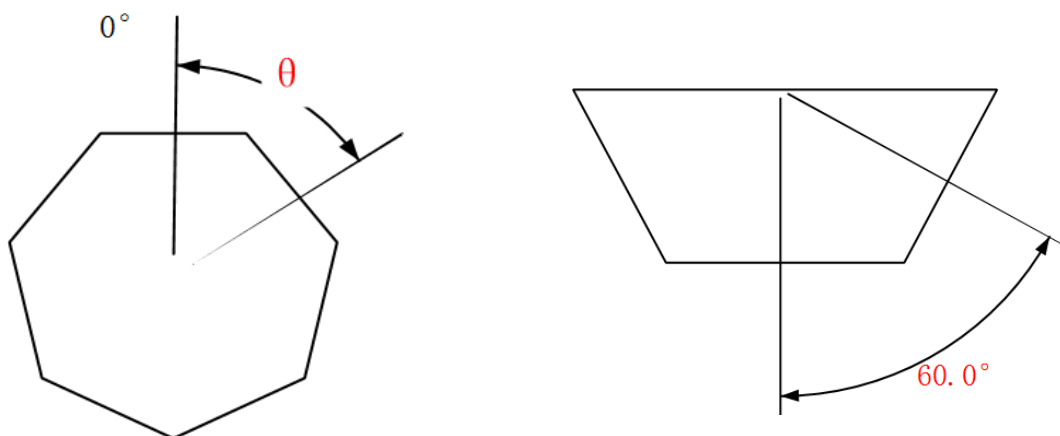


图 2-9 监控摄像头水平面投影（左）和铅垂面投影（右）

（1）周边摄像头的角度关系

定义中心摄像头像面平行于水平面，由于周边摄像头是七个摄像头均匀排列，因此该全景相机的俯视图切面可以简化为正七边形模型。图 2-9（左）为相机俯视图切面，周边七个摄像头的光轴分别垂直于七边形的每条边。按照几何关系，可以得到周边七个摄像头相邻两个在水平投影上的夹角 α 为 $360^\circ \div 7 = 51.43^\circ$ 。

（2）周边摄像头与中心摄像头光轴夹角

如图 2-9（右）所示，周边摄像头像面与中心摄像头像面（水平面）之间存在一定夹角。两个像面之间的夹角为 120° ，因此周边摄像头光轴与底部摄像头之间的夹角为 $180^\circ - 120^\circ = 60^\circ$ 。

2.4.2 单路图像的平面坐标到球面坐标的映射

设平面图像上某一像素点坐标为 (x, y) ，球面模型的半径为 f 。设球面上一点平面坐标为 (θ, φ) ，则其球面坐标系下坐标为 $k(\sin\theta, \cos\varphi, \sin\varphi, \cos\theta, \cos\varphi)$ ，其

中 k 为比例系数，即两者成正比关系。

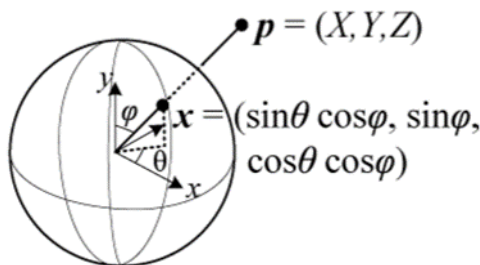


图 2-10 由空间三维坐标到球面坐标的投影

由上面的对应关系，得到单路图像平面坐标系上一点 (x, y) 到该点在球面坐标系下对应点的坐标 (x', y') 的对应关系为：

$$x' = f\theta = s \tan^{-1} \frac{x}{f} \quad (1-1)$$

$$y' = f\varphi = s \tan^{-1} \frac{y}{\sqrt{x^2 + f^2}} \quad (1-2)$$

当我们得到图 2-10 中的 θ 和 φ 角，就可以找到平面图像中某一点在球面图像中的位置。这两个角在空间中与水平方位角和俯仰角有关。

2.5 本章小结

本章介绍了全景相机的三种类型：带鱼镜头的全景相机、带全景云台的全景相机、集成多摄像头的全景相机。实验中我们使用的全景相机是一台集成八个摄像头的全景相机。为了方便研究，将其物理模型简化为两个平面上的投影，并得到了该模型的几何参数。本章最后介绍了全景图像的三种模型，对球面模型进行了简单分析。

第3章 目标检测与跟踪定位

为了减少算法的计算量,我们首先在单路视频图像中对运动目标进行跟踪定位,然后将其方位信息映射到全景图像中。当目标离开该路视频后,再在下一路视频中对运动目标进行跟踪定位。对目标进行跟踪定位的过程分为两个部分:首先是运动目标检测,检测出运动目标所在的图像区域;然后是提取目标坐标信息进行定位,并持续更新目标区域图像,更新其坐标信息,实现持续的跟踪。

在全景相机中,需要根据相机的几何结构和摄像头参数计算目标的空间方位信息。结合全景图像的投影模型,可以在全景图像中对目标进行标记。

3.1 单路图像运动目标检测

3.1.1 运动目标检测理论

运动目标的检测是指在视频图像中分辨出目标所在图像区域块的过程。如果要进行实时检测,则需要连续对目标区域进行更新。然后利用提取出的实时目标图像区域,进一步分析出目标的当前姿态或其他特征。

运动物体的检测过程和运动物体跟踪之间的联系非常紧密。一般来讲,运动物体的检测是运动物体跟踪的基础。运动目标跟踪流程中有两个很重要的过程:对目标进行分类、对目标进行连续跟踪定位。对这两个过程来说,运动检测的效果非常重要,因为运动检测给出的目标是动态的,实时的,可以把目标实时的信息传递给下一个处理流程。经过多年的发展,运动检测出现了很多经典的算法,今天的很多算法也是在这几种算法的基础上发展出来的。一般我们把运动检测方法分为三类:(1)帧间差分法(2)光流法检测(3)背景差分法。

(1) 背景差分法

背景差分法是一种简单有效的运动目标检测方法,应用最为广泛。它适应于场景背景和摄像头相对静止的情况,即摄像机固定的情况。首先,对无运动目标的视频背景图像进行采集,进行背景建模。之后需要设计一种方法对背景

模型进行更新，不同的算法有不同的更新方法。然后逐帧读取视频，将每一帧图像与背景模型做减运算，得到前景图像，前景图像实际上包含运动目标的信息。最后对前景图像进行处理，得到运动目标图像^[15]。所以利用背景差进行运动检测算法的关键是如何建立一个合适的背景模型以及如何对这个模型进行实时的更新，这两个问题如果解决的好，运动检测甚至是后期的目标跟踪效果效果都会得到提升^[16]。

使用背景差分法进行运动物体的检测这种方法，没有涉及很多复杂的计算过程，在静止背景的应用场合检测的效果比较好。但是有一些非常明显的缺点：①背景高速变化时检测效果下降；②外界光照条件变化检测效果变差；③物体阴影对检测结果干扰很大，这是因为在处理的过程中物体阴影会被当做是物体的一部分甚至是单独的一个运动物体。

面对光照条件变化、图形阴影、遮挡等问题，很多学者希望用不同的背景建模方法去解决。我国学者齐美彬采用了按像素出现概率建模的方法^[17]，这种方法对阴影去除有一定效果；国外的 Stauffer 和 Grimson 两位学者使用多个高斯分布混合模型对视频背景进行建模^[18]，而国内学者黄鑫娟等人对其方法进行了优化，使得高斯模型具有自适应性，可以适应快速变化的场景^[19]。

（2）帧差法

帧差法是通过比较相邻两帧图像的差别来检测出运动物体的。当背景静止并且物体运动时，相邻两帧图像的唯一差别就是运动物体的位置，将两帧图像相减，就可以提取出运动物体。在这一过程中，我们应该先对当前帧、前一帧或后一帧图像均进行灰度化处理，然后用当前帧图像与前一帧或后一帧图像做差值运算。

$$D_{t-1} = |\omega \times I_{t-1} - \omega \times I_t| \quad (3-1)$$

其中。 I_{t+1} 为当前时刻图像的灰度化图像， I_t 为前一时刻图像灰度化图像，最后将 D 进行 3×3 均值滤波和二值化处理，可以较为准确检测到物体的轮廓信息。

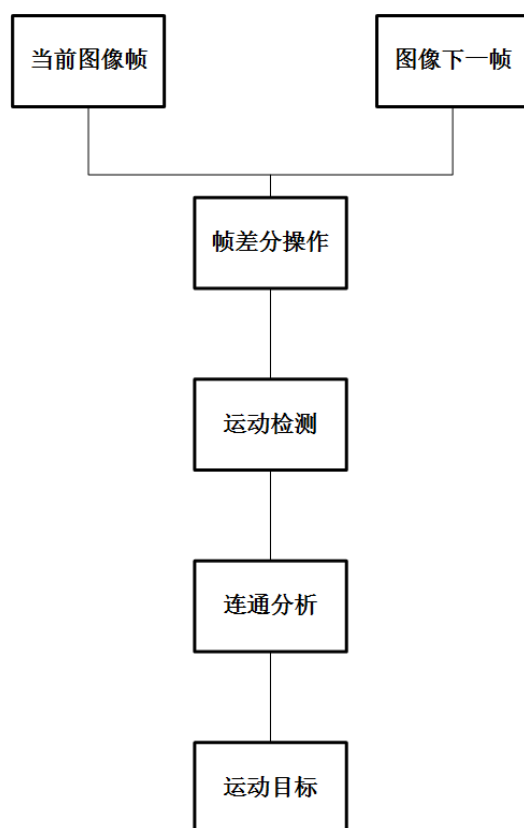


图 3-1 帧差分法流程图

这种方法的好处是计算量小，处理速度比较快。同时可以解决遮挡、照明条件变化的问题。但是缺点也很突出，受噪声影响很大，易受到干扰，最终对于目标的定位精确度不高^[20]。

按照帧间差分所用的图像帧数目，分为相邻双帧差法和三帧差法，三帧差法与双帧差法相比，可靠性更高^[21]。

为了提高该算法的可靠性，很多学者提出了改进后的算法并得到了较好的效果。美国的 Shahinfard 等人把参与运算的图像帧数提高到五帧，并且不单单提取灰度信息，而是加入图像颜色信息，最后提取出运动物体前景图像^[22]。由于加入了颜色信息，提高了处理帧数，该方法对高速运动物体和低速运动物体的检测效果都很好。

（2）光流法

光流指的是图像中运动像素点的瞬时速度场。通过检测运动物体对应响应度

点光流的变化，即计算其位移向量的变化，初始化运动物体的轮廓图像，进而对运动物体的运动信息进行提取。这种方法可以得到运动物体的完整信息，即使是在运动相机的应用场景下，也能实现较好的运动检测效果。但是光流法计算量比较大，算法一般都比较复杂，并且对于外界噪声适应性也不好。在经典的光流法算法基础上，后来又发展出 Horn&Schunck^[23]算法、Lucas&Kanade 算法^[24]以及块匹配算法等等。

3.1.2 使用帧差法进行运动检测

（1）帧间差分

获取视频流中当前帧图像和上一帧图像，做减运算。

（2）降噪处理

在帧差法的处理过程中，我们首先需要对图像进行了灰度化和二值化的操作，在这个过程中，需要选定二值化阈值，选择二值化阈值时需要考虑图像的亮度水平。由于我们采用固定阈值的方法进行二值化，在外界光照条件也会发生变化的情况下，极易产生椒盐噪声，而中值滤波可以较好的消除椒盐噪声^[25]。中值滤波进行滤波的方法，是先计算出当前像素附近一定区域内所有像素的中值（一般为一个正方形区域），然后用这个中值去替换这个正方形区域内所有像素的值。我们采用 3×3 矩阵的中值滤波器进行中值滤波操作。在 `opencv` 中，利用 `medianBlur` 函数进行中值滤波操作：`medianBlur (src, dst,i)`，其中，`src` 为输入图像，`dst` 为输出图像，`i` 为指定的窗口大小，必须为奇数。我们使用 3×3 的矩形窗口，因此 `i` 的值为 3。

（3）形态学操作

当提取出目标的二值化图像后，目标内部的深色部分或背景中的浅色部分会形成内部的空洞或外部的空洞，不利于我们提取出完整的目标图像，如图 3-2 所示。为了得到连通、独立的目标图像区域，需要去除掉这些空洞。

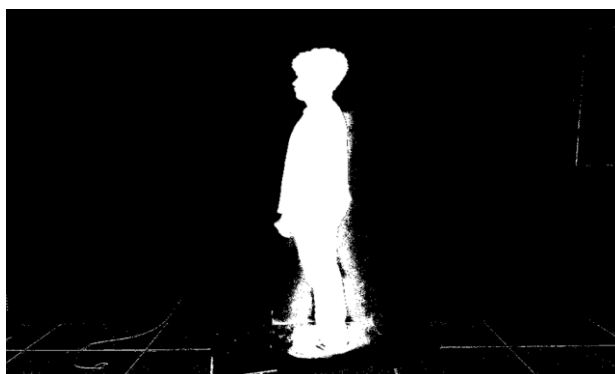


图 3-2 二值化效果示意图

膨胀和腐蚀是形态学中常见的两个操作，腐蚀可以消除图像中小的噪声区域，膨胀可以填补物体中的空洞。在实际操作中，我们往往需要同时使用这两种操作。实际上，组合使用这两种操作的顺序不同，效果也不同，操作组合分为两种：开运算和闭运算。开运算指的是先对图像进行腐蚀运算然后再进行膨胀运算，它可以消除图像中细小的杂物（或者是图像中的噪点）、分离图像中连接不紧密的两个部分、平滑物体的轮廓。闭运算指的是对一个图像先进行膨胀操作然后再进行腐蚀操作，它可以填充物体内部的空洞、把连接紧密的物体合并为一个整体、平滑物体的边界。

运动检测的过程需要进行灰度化和二值化等操作，当使用阈值对图像进行二值化操作后，会得到运动目标大体的轮廓。但是这时目标的轮廓是不平整的。同时物体内部会有一些空洞、目标轮廓外部（背景区域）会有很多细小的杂物，同时还可能有噪声干扰。这时就需要进行开运算和闭运算优化处理效果。

在 `opencv` 中，使用 `morphologyEx` 函数进行开闭运算的操作，`morphologyEx(src, dst, operation, element)`，其中，`operation` 参数定义开闭运算，`MORPH_OPEN` 定义开运算，`MORPH_CLOSE` 定义闭运算。对二值化图像进行开闭运算，效果如图 3-3。



图 3-3 形态学运算效果示意图

（3）显示检测出的运动物体

对经过形态学运算后的图像进行处理，寻找像素值为 1（白色部分）的连通区域并进行标记。

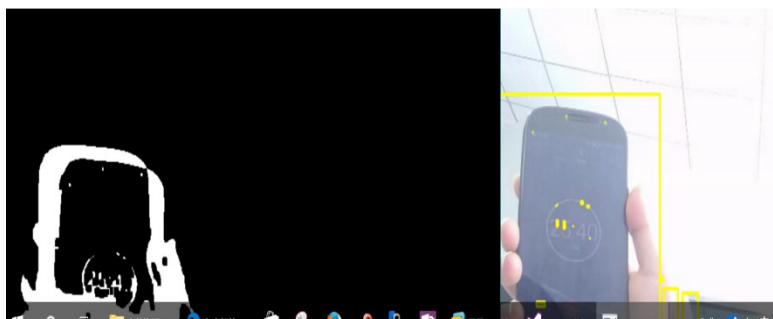


图 3-4 使用帧差法进行运动检测

3.2 单路图像目标的跟踪与识别

运动目标跟踪就是在图像序列中对运动目标的位置、形状等特征进行匹配，保证能实时获取目标的相关信息。这样我们最后就能得到运动目标的一个连续的运动轨迹。运动目标的跟踪算法按照原理分为四类^[26]：基于模型的方法、基于特征的方法、基于区域的方法、基于变形模板的方法。

在以上三种目标跟踪的方法中，基于特征的跟踪算法精度是比较高的，在运动物体被跟踪以后，如果出现了一切遮挡的物体或者是细小的干扰物体，基于特征的跟踪算法可以依据前期的特征继续跟踪运动物体。但是如何选择跟踪的特征是这一算法的难点，如果选择的特征过于单一，仍存在较大概率跟丢目标。所以一般会选择多个特征作为特征组合。这样一来又会带来计算效率的问题，选择的特征过多算法的计算量会增加，计算起来非常耗时^[27]。

利用 Opencv 中的模板匹配算法 `matchTemplate` 实现的运动目标跟踪实际上是属于基于区域进行跟踪这一方法。这种方法首先建立起目标的图像模板，根据模板在图像中进行匹配。但是基于区域的跟踪方法有一个共同的短板，一旦运动目标被遮挡，极易丢失需要跟踪的目标^[28]。

3.2.1 模板匹配算法简介

在模板匹配算法中，我们将需要跟踪的目标图像设为模板，在整个图像中进行匹配，检测匹配区域与模板区域图像的相关程度。如果相关程度达到了预定值，我们就认为，目标模板与此时的检测区域是相似的，即此区域可能是目标所在的区域。

首先，我们建立一个包含目标图像的模板，如图 3-5 所示。接着，我们对目标图像与原始图像进行比较，按照从左至右，从上至下的顺序进行“滑动”，一次只移动一个像素。每一次都进行一次度量，返回的匹配值表示目标模板与匹配区域的相似度。实现过程如图 3-6 所示。



图 3-5 建立目标模板



图 3-6 模板匹配过程示意图

在 opencv 中，使用 `matchTemplate` 函数实现模板匹配算法，提供六个匹配方法：

(1) 平方差匹配法：

这种方法利用平方差检测目标区域和搜索框相关度。计算结果为 0 表示最优匹配。结果越大两个区域越不相关。

$$R(x, y) = \sum (T(x', y') - I(x + x', y + y'))^2 \quad (3-2)$$

(2) 标准平方差匹配：

$$R(x, y) = \frac{\sum (T(x', y') - I(x + x', y + y'))^2}{\sqrt{\sum T(x', y')^2 - I(x + x', y + y')^2}} \quad (3-3)$$

(3) 相关匹配：

这种方法在目标模板和搜索框图像之间进行了乘法操作,结果越大，表示相关性越好，两者匹配结果越好。

$$R(x, y) = \sum (T(x', y') \times I(x + x', y + y')) \quad (3-4)$$

(4) 标准相关匹配：

$$R(x, y) = \frac{\sum (T(x', y') - I(x + x', y + y'))^2}{\sqrt{\sum T(x', y')^2 - \sum I(x + x', y + y')^2}} \quad (3-5)$$

(5) 相关系数匹配法：

这种方法先计算了目标模板的均值，然后判断搜索框图像与目标模板均值

的相关程度。结果为 1 表示相关程度很高，为 0 则表示毫不相关。

$$R(x, y) = \sum(T'(x', y') \times I'(x + x', y + y')) \quad (3-6)$$

其中：

$$I'(x + x', y + y') = I(x + x', y + y') - \frac{1}{\omega \times h} \times \sum_{x'', y''} I(x + x'', y + y'') \quad (3-7)$$

（6）标准相关系数匹配法：

$$R(x, y) = \frac{\sum(T(x', y') \times I(x + x', y + y'))^2}{\sqrt{\sum T(x', y')^2 \times \sum I(x + x', y + y')^2}} \quad (3-8)$$

以上匹配算法的复杂度依次增加，对设备处理能力的要求也是依次增加，但是匹配的精确度也越来越高。

3.2.2 利用模板匹配法进行目标跟踪

我们使用 opencv 的 matchTemplate 函数来实现目标跟踪算法。一般来说，我们选择相关匹配就可以得到不错的匹配结果，在实验中，我们选择最后一种方法标准相关匹配作为匹配方法进行模板匹配。流程如下：

- （1）选择一个目标物体的图像作为跟踪的目标模板，并进行保存。
- （2）使用 Rect 函数定义一个矩形搜索区域（searchWindow）。此区域应当包含目标模板，即搜索区域为目标所在的相邻区域。
- （3）计算目标模板（矩形）的形心坐标，输出坐标值。并标记目标模板。
- （4）实时更新目标模板及其坐标信息，同时更新搜索区域（searchWindow）的坐标信息，持续进行模板匹配并输出目标模板形心坐标。

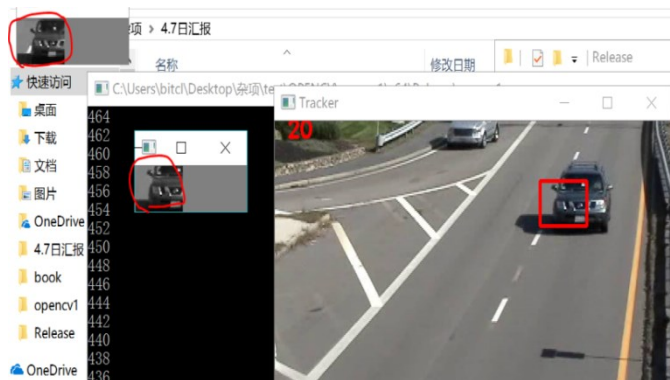


图 3-7 目标未被遮挡的跟踪效果

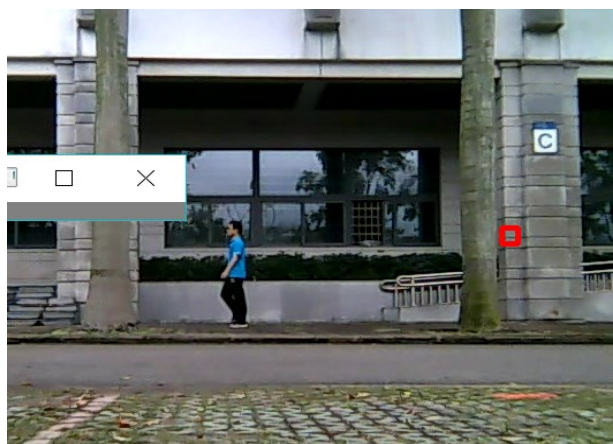


图 3-8 目标被遮挡的跟踪效果

当目标模板被遮挡时，跟踪标记框会停留在目标模板被遮挡的位置，程序丢失了跟踪目标，无法继续对目标进行跟踪，如图 3-8。当目标图像未被遮挡的条件下，该方法可以较好的对目标进行持续跟踪，如图 3-7。

对于不同的目标模板来说，他们的灰度值也不同。因此模板匹配法可以较好的识别区分不同的目标，可以实现多目标的检测跟踪。在图 3-7 中，有不止一辆的汽车出现在视频中，它们的轮廓和行驶速度等特征都是十分接近的。使用模板匹配法对其中一辆汽车进行跟踪，并没有跟错目标，也没有丢失目标。

3.3 全景视觉下进行目标跟踪的方法

在全景视觉下对目标进行跟踪，有两种方法：一种是先把多路图像进行拼接，合成为全景图像，然后直接在全景视频图像中进行运动目标检测、跟踪定位等操作。另一种是先在每一路视频图像中进行目标检测、跟踪定位操作，然后再对图像进行拼接显示，把单路视频中的目标坐标转换为全景视觉下的坐标进行输出^[29]，如图 3-9。

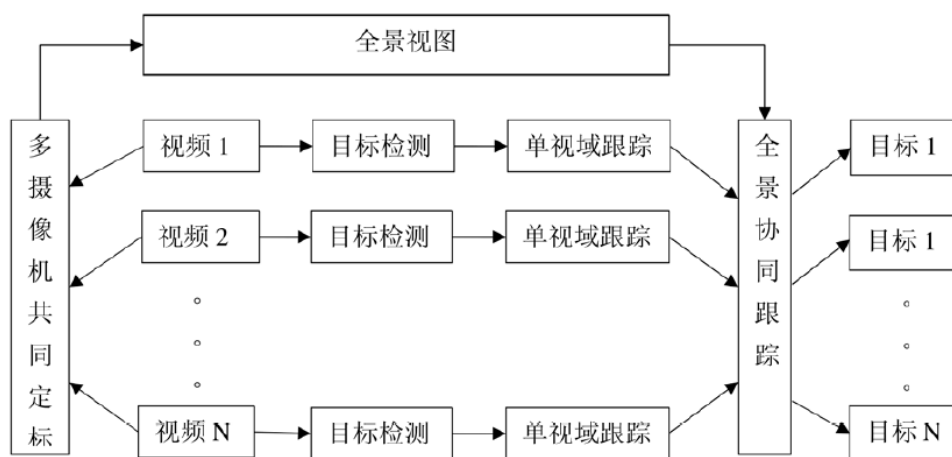


图 3-9 全景视觉下目标跟踪定位的一种方法

第一种方法在实现的过程中，由于全景图像的分辨率一般较高，直接对拼接合成的全景视频图像进行处理，处理速度很慢，无法满足实时性要求。实验中全景相机提取出的单路视频图像分辨率为 2048×1056 ，拼接后的全景图像分辨率可以达到 6144×3168 。直接在拼接后的全景图像中进行目标检测运算量非常大，因此我们采用第二种进行目标检测和跟踪。

在集成了多台相机的全景采集设备中，每台相机覆盖的视场是有限的，但是目标在移动的过程中会出现在不同的相机中。因此在目标检测和跟踪的过程中需要在不同相机之间进行切换，这是一种典型的跨视域目标跟踪场景^{[28][29]}。由于我们采用了先在单路视频中跟踪的跟踪方式，一个很棘手的问题，就是目标离开当前相机视场之后，难以确定目标接下来会在那一路相机的图像中，从而切换视频源，进行下一步跟踪^[30]。

在多摄像头目标跟踪系统中，在单路视频中丢失目标后如何继续搜索目标，有两种思路。

（1）对每台相机顺序搜索

当目标离开一路相机的视场后，程序会在其余路视频中寻找目标，查找的方式是对所有的视频序列进行无差别的遍历搜索。这种方式不会漏掉目标，但是计算量比较大。由于目标只会存在于一路视频图像中，所以实际上这种遍历

视频序列的检测方法有一些步骤是可以被简化的。

(2) 基于相机物理模型的搜索

在目标丢失后，我们对目标可能出现在哪一路相机中进行判断。然后在这一路视频图像中进行目标检测跟踪。

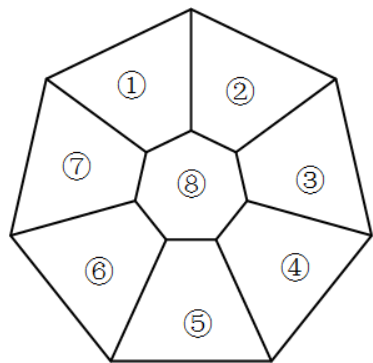


图 3-10 全景相机摄像头排列

图 3-10 是实验中使用的全景相机各个摄像头的编号。根据相机排列的位置关系可以发现，当目标离开某一路相机覆盖的视场后，其可能出现的区域是可以被预判的。例如，在目标离开 1 号相机视场后，接下来可能出现在 2、7、8 号相机的视场中；当目标离开 2 号相机后，可能出现在 1、3、8 号相机的视场中；当目标离开 8 号相机时，则可能会出现在 1 至 7 号的相机中。如图 3-11，我们在算法中加入了这一功能，将目标当前所在视频对应的相机序号送入程序，会返回下一路可能出现在哪几路相机对应的视场中。当目标在本路视频中消失时，算法会自动载入下一路视频进行检测跟踪，直到重新检测到目标。

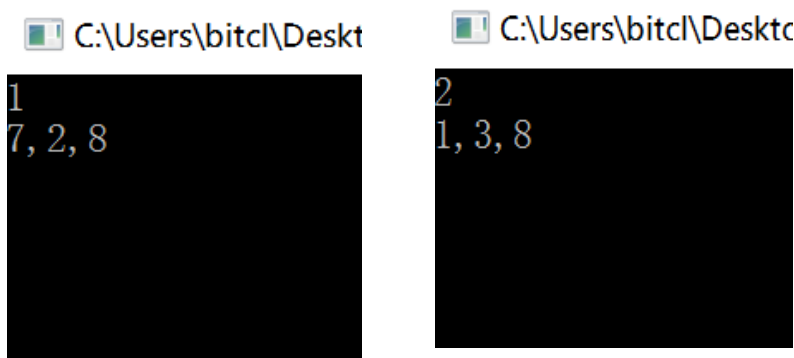


图 3-11 自动判断目标可能出现的位置

3.4 全景视觉下的目标定位

在前面的运动目标检测跟踪算法研究中，我们可以提取出目标的形心在单路视频二维平面内的二维坐标，定义为 (x,y) 。同时，我们也可以得到单路图像的中心像素坐标 (x_0,y_0) ，其中， $x_0=\text{cols}/2$ ， $y_0=\text{rows}/2$ ， cols 与 rows 分别为图像的宽度和高度，即图像在宽度方向的像素个数和高度方向的像素个数。由于采集得到的图像分辨率是固定的，假设为 1536×2048 。则图像宽度和高度方向上像素个数也是固定的，图像中心点坐标为 $(768,1024)$ 。

如图 3-12，水平的方位角 θ 定义为以 1 号相机光轴为 0° 起点，顺时针方向偏移的角度；俯仰角 φ 的定义为目标在铅垂面上偏离 8 号相机光轴的角度。

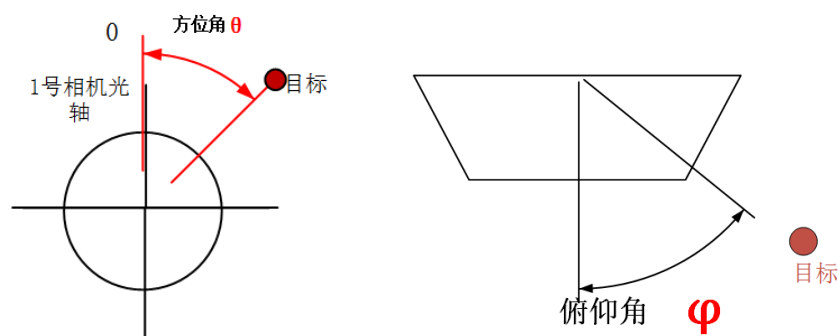


图 3-12 水平面上和铅垂面上方位角和俯仰角定义

(1) 水平面方位角计算

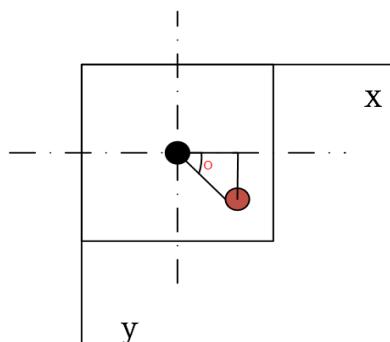


图 3-13 单路图像的二维坐标系

图 3-13 是单路图像中的一个二维坐标系，红色圆点为运动目标，定义坐标为 (x,y) ，图中黑色圆点为图像中心像素点，定义坐标为 (x_0, y_0) 。则图中直角三角形的两直角边分别为 $x-x_0$ 和 $y-y_0$ ，则图中：

$$O=\arctan\frac{y-y_0}{x-x_0} \quad (3-9)$$

对于 8 号相机，像面与水平面平行，因此水平面上的空间方位角正是 O 。

对于 1~7 号相机，每个相机对应的空间方位角为 $360^\circ \div 7=51.4^\circ$ ，如图 3-14 所示。在水平方向上运动目标与图形中心像素点坐标之差为 $(x-x_0)$ ， $(x-x_0)$ 与图像宽度之比乘以单路图像对应 51.4° ，即可以求出运动目标在水平面上与中心像素点之间的夹角。设 1 号相机光轴为 0° 位置，则 n 号相机中运动目标对应的空间方位角为：

$$\theta = \frac{360^\circ}{7} \times \left[\frac{(x_0-x)}{W} + (n-1) \right] \quad (3-10)$$

其中， W 为单路图像宽度， n 为图 3-10 中定义的相机序号

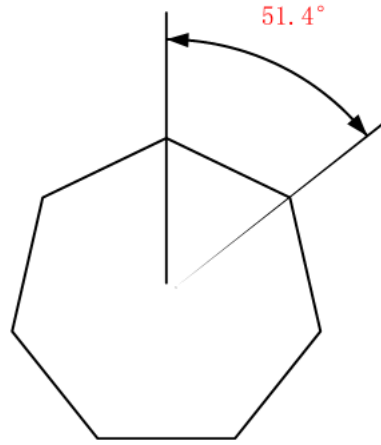


图 3-14 单个相机对应的有效视场角

(2) 俯仰角

对于排布在相机圆周上的 1~7 号相机，类比式 3-10，俯仰角为：

$$\varphi = \frac{y-y_0}{H} \times 60^\circ + 60^\circ \quad (3-11)$$

其中， H 为图像像素分辨率的高度。公式最后的 60° 为每个相机的光轴与铅垂线的夹角，如图 2-9 所示，是由相机的几何结构决定的。

当我们提出目标在单路视频中的坐标后可以带入上面的公式，计算出目标在空间中的位置，即全景视觉中的位置。

3.5 本章小结

在本章,我们研究了单路图像中的目标检测跟踪方法和全景视觉下目标跟踪的实现方法。对于单路图像的目标检测方法,我们分析对比了背景减除法、临帧差法和光流法,最终选择使用临帧差法作为实验方法。对于单路图像的目标跟踪算法,我们研究了一种基于区域的目标跟踪算法^[31]——模板匹配法,并使用 `opencv` 进行实验,在目标被遮挡和未被遮挡两种情况下对该算法进行验证。在目标未被遮挡的情况下该算法可以对目标进行持续的跟踪,效果良好。然后我们结合实验中使用的全景相机,实现了跨域跟踪的算法。最后,我们对目标的定位方法进行了讨论,结合第二章中的全景相机模型,进行目标的水平方位角、相对于水平面的俯仰角的求解,最终得到目标的空间位置关系。

第 4 章 系统实现与评价

整个系统分为视频采集、图像处理和结果输出三个部分。视频采集方面，由于我们选择了先对单路视频进行目标检测跟踪，再把目标坐标映射到全景图像中这种方法，所以我们需要分别采集出每一路摄像头的视频图像。图像处理方面，我们采用第三章中的方法，首先检测出单路视频图像中目标的坐标数据，再进行坐标变换。最后，我们希望输出目标在全景视觉中的信息——包括目标在全景图像（球面投影模型）中的水平方位角和俯仰角。

4.1 多传感器图像的采集

图像采集系统中在工作中，有两个重要的性能指标：视频信号的实时性和各路视频之间的同步性^[32]。特别是在多传感器组成的全景系统中，各路视频的同步性更为重要，如果各路视频在时间上不匹配，拼接出的全景图像就不能还原真实的监控场景，在运动目标检测和跟踪定位中，还会导致目标识别错误、定位不准确等结果。多路视频的实时采集和保存一般需要专业的视频采集卡完成，把相机的模拟信号转换为数字信号进行传递，实验所用相机内置了图像采集和转码设备。同时，我们了解到该相机支持监控设备行业的 ONVIF 协议，该协议支持 rstp 实时流媒体传输功能^{[33][34]}。我们使用 ONVIF Device Test Tool 15.06 分析工具对相机参数进行分析，并获取视频流地址。

```
HTTP/1.1 200 OK
Server: gSOAP/2.8
Content-Type: application/soap+xml; charset=utf-8
Content-Length: 2203
Connection: close

<?xml version="1.0" encoding="UTF-8"?><SOAP-ENV:Envelope xmlns:SOAP-ENV="http://www.w3.org/2003/05/soap-envelope" xmlns:SOAP-ENC="http://www.w3.org/2003/05/soap-encoding" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:wsa5="http://www.w3.org/2005/08/addressing" xmlns:wsd="http://schemas.xmlsoap.org/ws/2005/04/discovery" xmlns:tett="http://www.placeholder.org/ver10/tmp/schema" xmlns:text="http://www.placeholder.org/ver10/tmp/wsd" xmlns:xmime="http://tempuri.org/xmime.xsd" xmlns:xop="http://www.w3.org/2004/08/xop/include" xmlns:tt="http://www.onvif.org/ver10/schema" xmlns:wsbf="http://docs.oasis-open.org/wsrf/bf-2" xmlns:wstop="http://docs.oasis-open.org/wsn/t-1" xmlns:wsr="http://docs.oasis-open.org/wsrf/t-2" xmlns:dn="http://www.onvif.org/ver10/network/wsd" xmlns:tan="http://www.onvif.org/ver20/analytics/wsd" xmlns:tds="http://www.onvif.org/ver10/device/wsd" xmlns:tet="http://www.onvif.org/ver10/events/wsd" xmlns:wst="http://docs.oasis-open.org/ws/b-2" xmlns:timg="http://www.onvif.org/ver20/imaging/wsd" xmlns:tmd="http://www.onvif.org/ver10/device0/wsd" xmlns:tptz="http://www.onvif.org/ver20/ptz/wsd" xmlns:trc="http://www.onvif.org/ver10/recording/wsd" xmlns:tsp="http://www.onvif.org/ver10/replay/wsd" xmlns:trt="http://www.onvif.org/ver10/media/wsd" xmlns:trv="http://www.onvif.org/ver10/receiver/wsd" xmlns:tse="http://www.onvif.org/ver10/search/wsd" xmlns:ter="http://www.onvif.org/ver10/error" xmlns:trns1="http://www.onvif.org/ver10/topics" xmlns:wase="http://docs.oasis-open.org/ws/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:wsu="http://docs.oasis-open.org/ws/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" xmlns:tr2="http://www.onvif.org/ver20/media/wsd"/><SOAP-ENV:Header><SOAP-ENV:Header><SOAP-ENV:Body><dt:GetStreamUriResponse><dt:MediaUri><tt:Uri>rtsp://192.168.12.100:554/Onvif/live/8/2/<tt:Uri><tt:InvalidAfterConnect>false<tt:InvalidAfterReboot>false<tt:InvalidAfterReboot><tt:Timeout>PT0S<tt:Timeout><tt:MediaUri><tt:GetStreamUriResponse></SOAP-ENV:Body></SOAP-ENV:Envelope>
```

图 4-1onvif 分析工具分析结果

最终得到相机的直播地址为 rtsp://192.168.12.100:554/Onvif/live/n/1。其中

n 为相机编号，为 1~8 的数字，最后一位为流媒体传输的通道，我们选择 1 号通道，该通道提供高质量图像传输，最终传输的图像为 2048×1536 分辨率、h264 编码、帧数为 24.98 帧的高质量视频图像。我们可以使用开源播放器 VLC 播放器进行保存该视频流，也可以在 linux 系统下使用 ffmpeg 工具保存该视频流。图 4-2 为 VLC 播放器正在播放实验中全景相机一号摄像头图像。

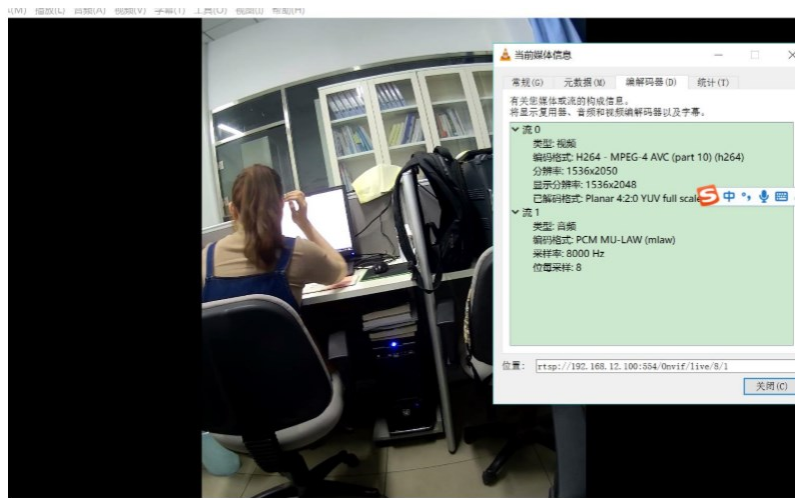


图 4-2 vlc 读取并播放网络视频流

4.2 图像处理与数据输出

4.2.1 图像处理流程

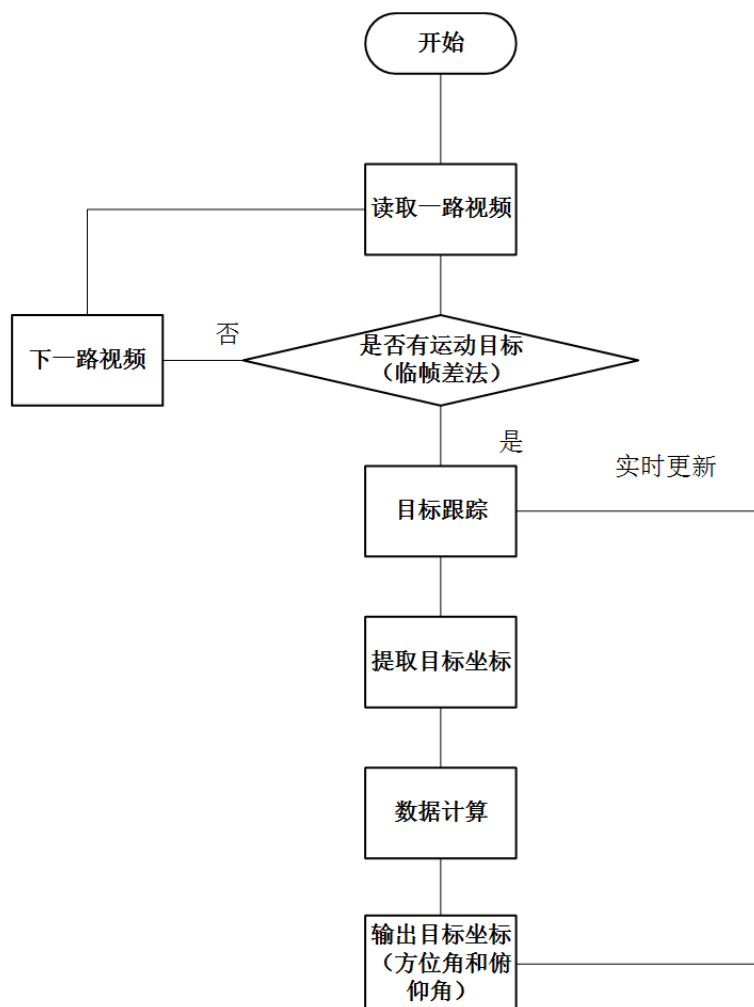


图 4-2 算法执行流程

首先，我们对某一路视频进行运动目标检测。如果该路视频有目标，则进行目标跟踪操作。如果没有目标，则目标的位置进行估计，读取下一路可能存在目标的视频。

判断该路视频是否有目标的方法是统计图像二值化后非零像素点数量小于一定的阈值即认为该路视频无目标。判断目标下一路可能出现的位置是根据相机的物理模型进行程序设计的，中间结果如图 3-11 所示。

在跟踪的过程中，我们对目标所在区域使用矩形框进行标记，同时计算该矩形形心，输出形心坐标作为目标在该路视频中的平面坐标，如图 4-3 所示。

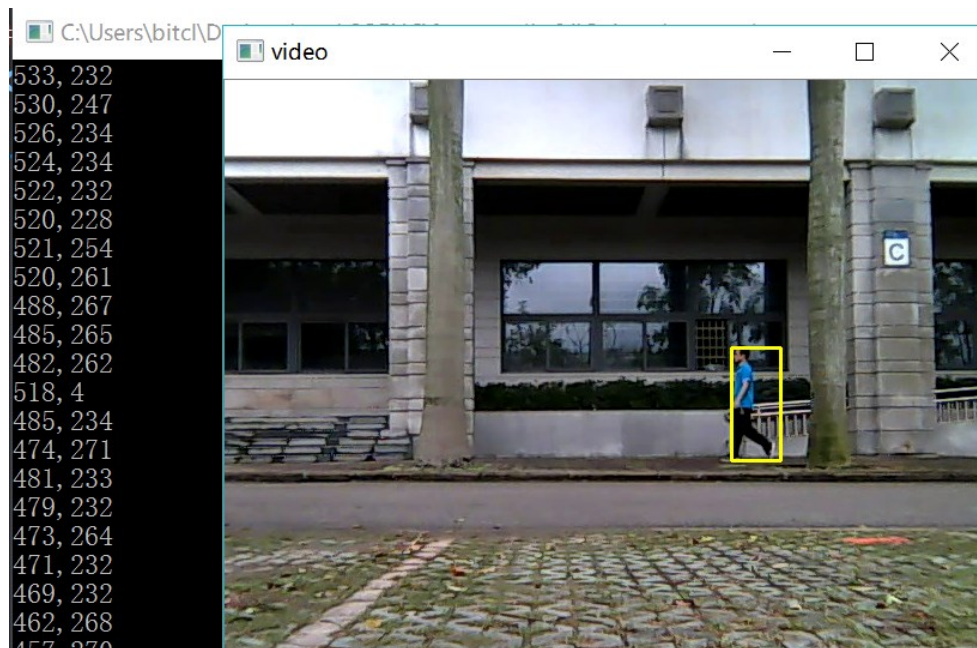


图 4-3 输出运动目标平面坐标

定义目标形心坐标 (x, y) , 单路图像中心点坐标为 (x_0, y_0) , 相机序号为 n , 图像高度为 H , 则:

对 1~7 号相机, 带入式 3-10 和式 3-11 计算出目标方位角和俯仰角; 对 8 号相机, 带入式 3-9 计算其方位角。

当该路视频中的目标丢失时, 自动切换下一路视频进行处理。下一路视频的选择方法按照第三章 3.3 小节介绍的方法, 利用相机的物理模型, 实现提前预判目标可能出现的区域, 减少程序的计算量。

4.2.2 数据输出

当提取出目标区域之后, 我们可以计算出在平面图像中目标形心的坐标, 如图 4-3 所示, 命令行窗口的输出即为目标区域形心坐标。在分析运动目标坐标的过程中我们发现, 目标的坐标会有瞬时性的跳跃, 如图 4-3 左侧数据所示, 前一刻目标坐标为 $(482, 262)$, 下一刻突然跳变为 $(518, 4)$, 这是因为在这一时刻目标被遮挡, 算法无法检测出目标真实位置。

在球面模型和全景视觉中, 我们需要目标的两个信息才能对其进行定位,

这两个信息是水平面上的方位角和铅垂面上的俯仰角。在上一节中介绍了根据目标形心坐标计算目标的水平方位角和俯仰角的处理方法。把测试视频定义为4号相机的视频图像，输出的结果如图4-4所示，结果为保留两位小数的单精度浮点数。



图 4-4 输出水平方位角和俯仰角

4.3 本章小结

本章介绍了目标跟踪定位系统中图像采集、图像处理和数据输出三个模块的具体内容。在图像采集模块中，我们结合实验中使用的全景相机，利用流媒体协议来同时采集多路视频图像，保证了实时性和各路视频的同步性。在图像处理模块中，先对单路视频进行目标跟踪定位，得到坐标在单路视频中的二维坐标，然后结合相机物理模型，计算出目标在空间中的方位角和俯仰角。为了减少算法计算量，根据相机的物理模型，设计了多路视频的处理顺序。最后，在数据输出模块，将计算得到的方位角和俯仰角数据显示在屏幕上并实时更新。

第 5 章 总结与展望

5.1 本文工作总结

本文结合实验室中的全景相机，对全景图像中的目标跟踪定位技术进行了研究。主要包括以下几个方面的内容：

（1）图像采集系统的调试

利用 onvif 协议和 rtsp 协议提取出全景相机中每个摄像头的图像并进行保存。并且保证了采集的实时性和各路视频的同步性。

（2）全景相机模型的构建

分析了实验室全景相机的几何结构，构建了相机的物理模型，获取了相机的外形参数。同时了解了全景图像的三种投影模型，特别是球面投影模型。

（3）单路视频目标检测和跟踪方法的研究

对单路视频图像进行目标检测跟踪是一项基础工作。本文分析比较了几种不同的检测方法，选择简单高效的帧间差分法作为运动目标检测方法，并进行了实验验证。同时也试验了模板匹配算法的跟踪效果，分析其优点和不足之处。

（4）全景图像下目标检测跟踪

选择了先对在单路视频图像中进行目标检测跟踪，然后把单路视频中目标坐标映射到全景图像坐标中的跟踪定位方法。在全景视觉中，我们需要的两个信息是目标的水平方位角和现相对于相机的俯仰角，结合实验所使用全景相机的物理结构，我们把单路视频图像中目标的二维坐标转换为全景视觉中的坐标。最终，我们实现了对目标的连续跟踪和全景视觉下的坐标输出。

5.2 未来工作展望

本文通过对单路视频图像进行跟踪定位，再转换为全景视觉下的空间坐标这种方式，实现了全景图像中目标的跟踪定位。因本人理论水平和研究能力限制，实现的算法还是有很多不足之处。主要有以下几点：

（1）目标被遮挡后难以继续跟踪

使用模板匹配法进行目标跟踪，在目标被遮挡后极易丢失目标，这也是所有基于区域的跟踪方法都会遇到的问题。在以后的研究中，将对目标移动的方向、位置进行预判，目标被遮挡后预估下一时刻位置，实现连续跟踪。

（2）多路视频的自动切换算法

本文结合实验所用全景相机各摄像头的排列关系，分析出单路视频中丢失目标后切换下一路视频的顺序。本文的算法仅仅针对本设备有效，不具备通用性。以后的研究中将结合目标运动的速度、方向预判下一路视频的顺序。

（3）目标定位的精度

由于实验条件限制，本文并未就定位的精确度进行验证。本文目标跟踪定位方法的前提是目标和全景相机之间有较远的距离。未来需要对本算法的结果进行验证。

（4）全景图像中目标的标记

本文不涉及图像拼接算法，对于全景图像中目标的定位是依据实验所用全景相机的几何结构得出的。最终得到的是全景视觉下目标的空间方位角等信息。在后面的研究中我们将结合图像拼接的算法，把目标的方位在拼接得到的全景图像中进行标记，便于直观的判断目标的空间位置。

参考文献

- [1]Stauffer C, Grimson WEL. Adaptive Background Mixture Models for Real-Time Tracking[C]Computer Vision and Pattern Recognition,1999.IEEE Computer Society Conference on. IEEE Xplore, 1999:252:2.
- [2]赵以恒.360°全景技术的应用及发展研究[J].中国高新技术企业,2014,(34):66-67.
- [3]宋永博.基于全方位视觉的便携式目标检测跟踪系统[D].北京理工大学,2015.
- [4]谢勤岚,陈红.基于多CPU的目标识别与跟踪系统设计[J].计算机测量与控制,2009,17(8):1525-1526.
- [5]高腾.静止背景下运动目标跟踪方法的研究[D].西北大学,2005.
- [6]李勍勍.运动目标检测与跟踪研究[D].四川大学,2001.
- [7]付秀明.基于运动序列图像的目标定位及应用研究[D].重庆理工大学,2015.
- [8]徐宇.多摄像机全景协同的运动目标跟踪算法研究[D].西华大学,2014.
- [9]邹超洋.基于多摄像头全景图像拼接的实时视频监控技术研究[D].华南理工大学,2014.
- [10]周明晖,胡士强,陈思聪.基于全景摄像头的柱面展开及实时目标跟踪[J].计算机工程,2013,39(11):1-4.
- [11]邓颖娜,朱虹.构造多相机全景视图实现有重叠区域的目标跟踪[J].西安理工大学学报,2009,25(2):189-192.
- [12]刘永信,魏平,侯朝桢.视频图像中运动目标检测的快速方法[J].仪器仪表学报,2002,(S3):163-166+176.
- [13]陈德印.360度全场景多目标识别与跟踪算法的研究与实现[D].南京邮电大学,2013.
- [14]李颖晶.基于全方位视觉的运动目标检测和跟踪方法研究[D].哈尔滨工程大学,2012.
- [15]王辉.背景差分图像处理[D].哈尔滨理工大学,2005.

-
- [16]李晨. 智能视频监控中的算法研究及一种新的背景差分模型[D].吉林大学,2008.
- [17]齐美彬, 王倩, 蒋建国.基于背景像素值频次最高假设的背景重构算法, 中国图象图形学报, 2008 13(4) : 723-728.
- [18]Stauffer, C., Grimson.W.E.L. Learning patterns of activity using real-time tracking. Pattern Analysis and Machine Intelligence IEEE , 2000 22(8):747-757 .
- [19]黄鑫娟, 周洁敏, 刘伯扬. 自适应混合高斯背景模型的运动目标检测方法. 计算机应用, 2010 30(1).
- [20]胡敬舒. 基于帧间差分的运动目标检测[D].哈尔滨工程大学,2013.
- [21]郭保文. 基于改进帧差法的视频分析服务器的设计与实现[D].中国矿业大学,2015.
- [22] Shahinfard, E, Sid-Ahmed, M.A., Ahmadi, M. A motion adaptive deinterlacing method with hierarchical motion detection algorithm. ICIP 2008. 15th IEEE :889-892.
- [23]Perez P, Gangnet M, Blake A.Poisson image editing[J]. ACM Transactions on Graphics, 2003, 22(3): 313.
- [24]张永亮, 卢焕章, 高劼,等. 一种改进的 Lucas-Kanade 光流估计方法[J]. 海军航空工程学院学报, 2009, 24(4):443-446.
- [25]袁新星. 基于中值滤波的高密度椒盐噪声图像去噪算法研究[D].湖北工业大学,2014.
- [26]徐宇.多摄像机全景协同的运动目标跟踪算法研究[D].西华大学,2014.
- [27]陈德印.360 度全场景多目标识别与跟踪算法的研究与实现[D].南京邮电大学,2013.
- [28]Lin D T, Huang K Y. Collaborative Pedestrian Tracking and Data Fusion With Multiple Cameras[J]. IEEE Transactions on Information Forensics & Security, 2011, 6(4):1432-1444.
- [29]Kang J, Cohen I, Medioni G. Persistent Objects Tracking Across Multiple Non Overlapping Cameras[C] IEEE Workshop on Applications of Computer Vision IEEE Workshop on Motion and Video Computing. DBLP, 2005:112-119.
- [30]Meyer M, Hotter M, Ohmacht T. New options in video surveillance

applications using multiple views of a scene[C] 1998 International Carnahan Conference on Security Technology, 1998. Proceedings. IEEE Xplore, 1998:216-219.

[31]刘晴. 基于区域特征的目标跟踪算法研究[D].北京理工大学,2014.

[32]李靖. 面向全景视觉的多路视频采集处理系统研究与实现[D]. 北京交通大学, 2011.

[33]储强. ONVIF 协议媒体服务的设计实现[D].哈尔滨工业大学,2013.

[34]黎屹. 网络流媒体协议研究及应用实现[D]. 电子科技大学, 2005.