

教育部高等学校大学计算机课程教学指导委员会

# 中国大学生计算机设计大赛



## 软件开发类作品文档简要要求

作品编号：\_\_\_\_\_

作品名称：\_\_\_\_\_ 基于无人机水面垃圾清理系统 \_\_\_\_\_

作 者：\_\_\_\_\_

版本编号：\_\_\_\_\_ 1.0 \_\_\_\_\_

填写日期：\_\_\_\_\_ 2019 年 4 月 30 日 \_\_\_\_\_

### 填写说明：

- 1、本文档适用于**所有**涉及软件开发的作品，包括：软件应用与开发、大数据、人工智能、物联网应用；
- 2、正文一律用五号宋体，一级标题为二号黑体，其他级别标题如有需要，可根据需要设置；
- 3、本文档为简要文档，不宜长篇大论简明扼要为上；
- 4、提交文档时，以 PDF 格式提交本文档；
- 5、本文档内容是正式参赛内容组成部分，务必真实填写。如不属实，将导致奖项等级降低甚至终止本作品参加比赛。

## 目 录

第一章 需求分析.....	3
第二章 概要设计.....	3
第三章 详细设计.....	4
第四章 测试报告.....	5
第五章 安装及使用.....	6
第六章 项目总结.....	9

# 第一章 需求分析

由于当前水域污染日益加重，一些水深较浅、空间狭窄的水域，打捞船不便工作，人工下水打捞由于水下环境不明存在危险。而目前的清理机器主要依靠人工操控，独立性、智能性不强，而无人机作为一种空中作业机器，不会受限于水底环境和空间，便于进入人和打捞船不便进入的环境中清洁水面垃圾。作品主要面向池塘，鱼塘和湖面等存在垃圾的水域。本作品主要功能为通过无人机搭建的摄像头进行目标检测，确定水面垃圾坐标，规划出较优路径并开展水面垃圾清理作业，明显提高了工作效率，降低了成本。无人机性能稳定，飞行较为平稳，能够在很短时间内有效识别到垃圾，而且能支撑无人机进行大规模的垃圾清理。

# 第二章 概要设计

根据需求分析进行模块功能设计，系统总共分为三个功能模块。

1. 无人机飞控和图像传输模块：用于控制无人机飞行距离、转向角度等行动，传输无人机摄像头拍摄到的实时视频流。
2. 图像目标检测模块：用于识别无人机视角中的水面垃圾，并获取各处水面垃圾相对于无人机的坐标信息。
3. 路径规划模块：根据图像目标检测模块得到的垃圾坐标进行路径规划，无人机再根据规划好的路线依次完成作业。

三个模块均可在终端监视其工作状态，各模块数据传送关系如图 1 所示，系统工作流程图如图 2 所示。

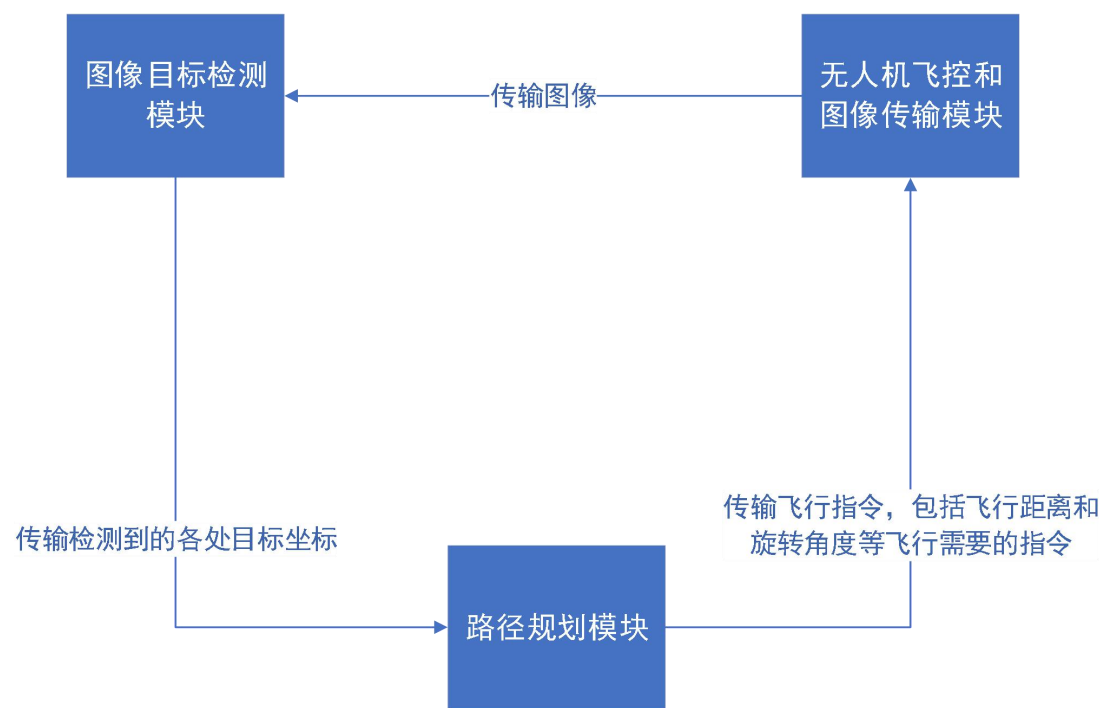


图 1 模块数据传送示意图

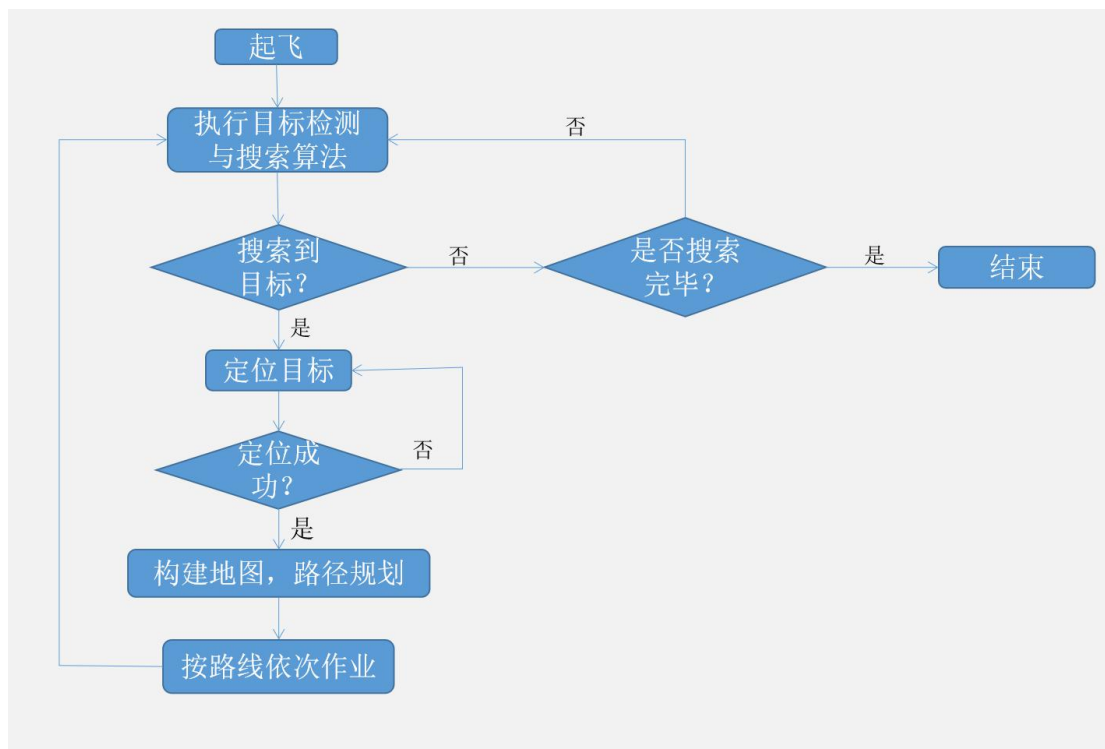


图 2 系统流程图

## 第三章 详细设计

1. 界面设计：本作品为实物类作品，用户交互界面主要是 Python 控制台和视频流接收窗口，如图 3 所示。

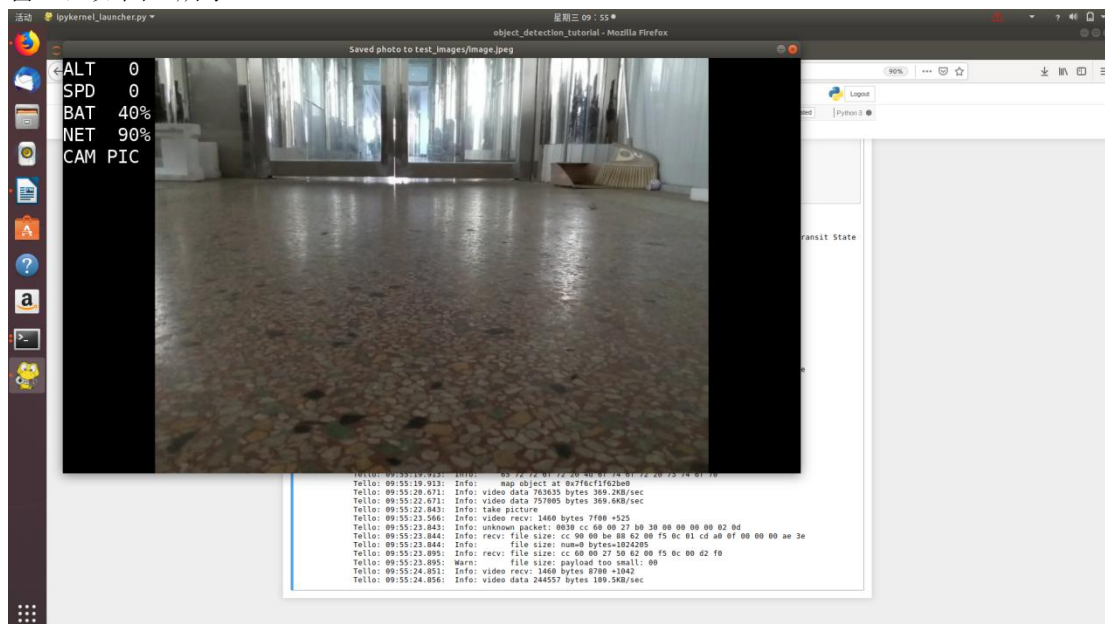


图 3 交互界面示意图

2. 关键算法：

(1) 图像目标检测算法：传统的目标检测算法识别速度比较慢，而且对于运动中的目

标识效果较差。本作品使用基于神经网络结构的 R-FCN 目标检测神经网络,在 Faster RCNN 网络基础上进行改进,将 Faster RCNN 中的 ResNet 中最后不共享计算的 10 层也放到了前面共享计算的 subnetwork 中,最后比较消耗时间的用来 prediction 的卷积只剩下 1 层,减小计算量,对于运动中的目标的识别效果也比较好。R-FCN 算法的运行步骤大致如下。

- ①首先,将需要处理的图片进行预处理,包括图像尺寸缩放等操作。
- ②接着将预处理后的图片送入我们已经训练好的分类网络中,固定其对应的网络参数。
- ③在已经训练好的最后一个卷积层获得的 feature map 上存在 3 个分支,第 1 个分支就是在该 feature map 上面进行的 PRN 操作,获得相应的 ROI;第 2 个分支就是在该 feature map 上获得一个  $K \times K \times (C+1)$  维的位置敏感得分映射,用来进行分类;第 3 个分支就是在该 feature map 上获得一个  $4 \times K \times K$  维的位置敏感得分映射,用来进行回归。
- ④在  $K \times K \times (C+1)$  维的位置敏感得分映射和  $4 \times K \times K$  维的位置敏感得分映射上面分别执行位置敏感的 ROI 池化操作,获得对应的类别和位置信息。

(2) 基于遗传算法的路径规划:将无人机路线规划需求抽象出来就是 TSP 问题。对于 TSP 问题,传统的求解最优路径的算法其时间复杂度为  $O(n^n)$ ,代价过大。而使用遗传算法围绕了三个主要步骤:选择—复制、交叉、变异,模拟染色体的遗传过程,进行求解。通过遗传算法迭代求解出次优解花费时间较少,且计算结果能够满足实际需求。遗传算法的运行大致步骤如下。

- ①输入目标点位的坐标,生成坐标字典。
- ②选择—复制:从种群中挑选出指定数量的染色体进行复制,重新计算每个种群的适应度等信息,与已经保存的最优染色体进行比较。如果比已经存在的适应度还要高就进行最优染色体的更新。如果最优染色体没有更新,则在新生成的种群中找到适应度最低的,用最优染色体替换掉。
- ③每一代的繁衍都让种群中相邻的两个种群进行染色体交叉,直接交换由随机数产生的染色体片段。交叉完成后要重新计算每个种群的适应度等信息,并进行染色体更新,与②操作相同。
- ④根据一定的概率执行变异操作,用于调整染色体中每一段的顺序。
- ⑤将计算得到的最优路径序列,已经序列中每个点的坐标作为函数返回值返回。

## 第四章 测试报告

- 1. 无人机飞控与图像传输协议模块测试:
  - (1) 动作指令测试:对无人机接收前进、转向等动作指令精度进行测试,测试结果如表 1 所示。

表 1 动作指令测试结果

指令名称	误差	执行时间(从发出到无人机接收到指令的时间)
前进	<10cm	0.000143sec
后退	<10cm	0.000142sec
顺时针旋转	<8°	0.000148sec
逆时针旋转	<8°	0.000146sec
上升	<10cm	0.000143sec

下降	<15cm	0.000161sec
----	-------	-------------

(2) 图像传输测试：对接收无人机使用 UDP 协议传输图像的各项指标进行测试，测试结果如表 2 所示。

表 2 图像传输测试结果

	视频流	图片
每秒传输速率/接收每张图片时间	369.2KB/sec	1.528114sec

2. 图像目标检测模块测试：对目标检测模块的识别速度和识别准确度进行测试。其中，由于测试识别准确度需要大量测试数据，故准确度测试使用 VOC 07 数据集进行测试，测试结果如表 3 所示。

表 3 图像目标检测测试结果

识别准确率 (%)	处理每张图片所需时间
79.3	2.720324sec

3. 路径规划模块测试：根据目标点数量（不包括原点）进行路径规划的时间测试，规划出的路径长度与最优路径长度相对误差（（规划长度-最优长度）/最优长度）测试，测试结果如表 4 所示。

表 4 路径规划测试结果

目标点数量	耗费时间	与最优路径的相对误差
10	0.891790	0.198183
20	1.281300	0.231466
50	2.401642	0.221748
100	4.131506	0.137454
200	7.465850	0.201885

## 第五章 安装及使用

1. 环境说明：Ubuntu18.04, Anaconda3-4.2.0, Python3.6, TensorFlow1.13.1

2. 需要的 Python 库：numpy, os, tellopy, av, opencv-python, time, six, moves, urllib, sys, tarfile, traceback, zipfile, math, distutils.version, collections, io, matplotlib, PIL, object\_detection.utils, utils, pygame, datetime, threading, socket, subprocess。

3. 安装过程：

(1) 安装以上环境，安装路径选择默认，并为 TensorFlow 安装 jupyter-notebook。

(2) 切换到代码文件所在的目录下，打开终端，切换到 TensorFlow 安装环境下打开 jupyter-notebook，如图 4 所示。

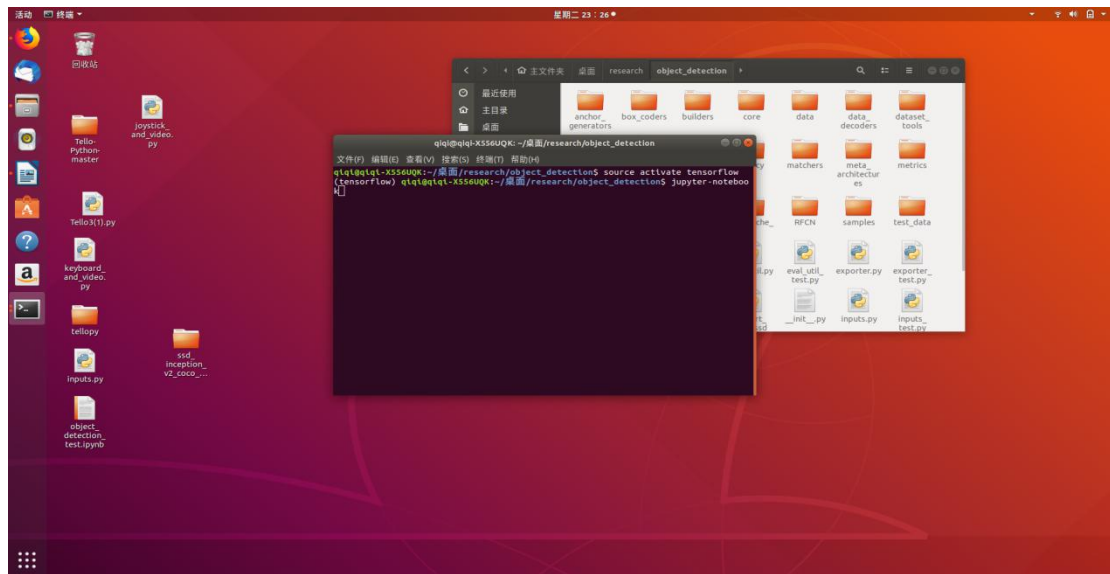


图 4 切换路径，打开 jupyter

(3) 在 jupyter-notebook 中打开代码文件，如图 5 所示。

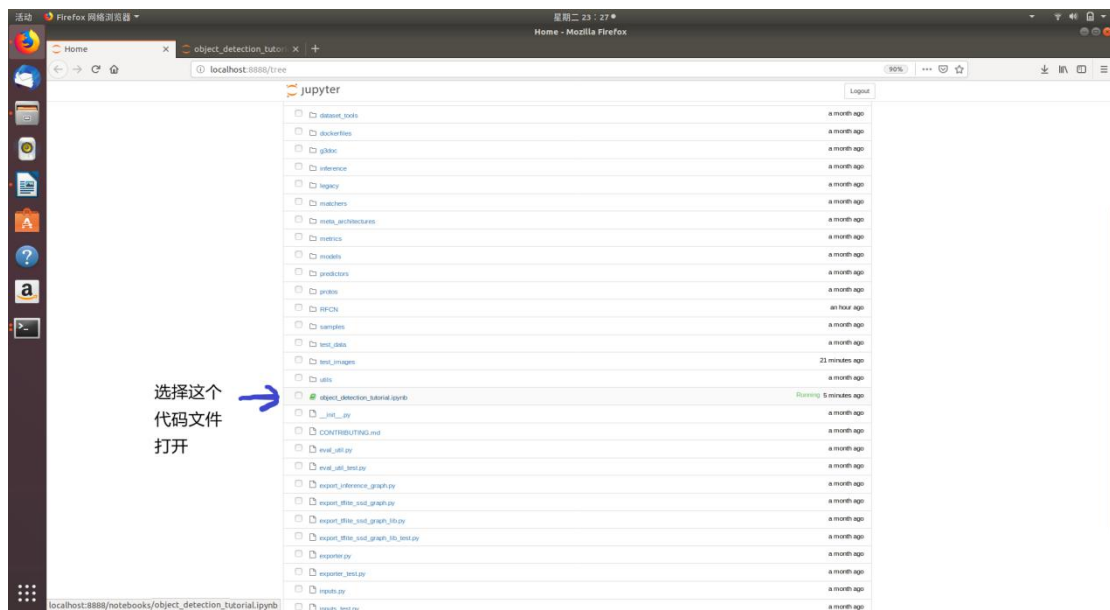


图 5 打开代码文件

(4) 打开无人机电源开关，搜索无人机的 WiFi 信号并连接，如图 6 所示。(注：(3)、(4) 两步可交换顺序)。



图 6 连接无人机 WiFi

(5) 选择 jupyter-notebook 的 Kernel->Restart&Run All 即可运行代码，如图 7 所示（每次运行都要使用此种方法，若不需运行，可以选择 Kernel->Restart&Clear Output）。

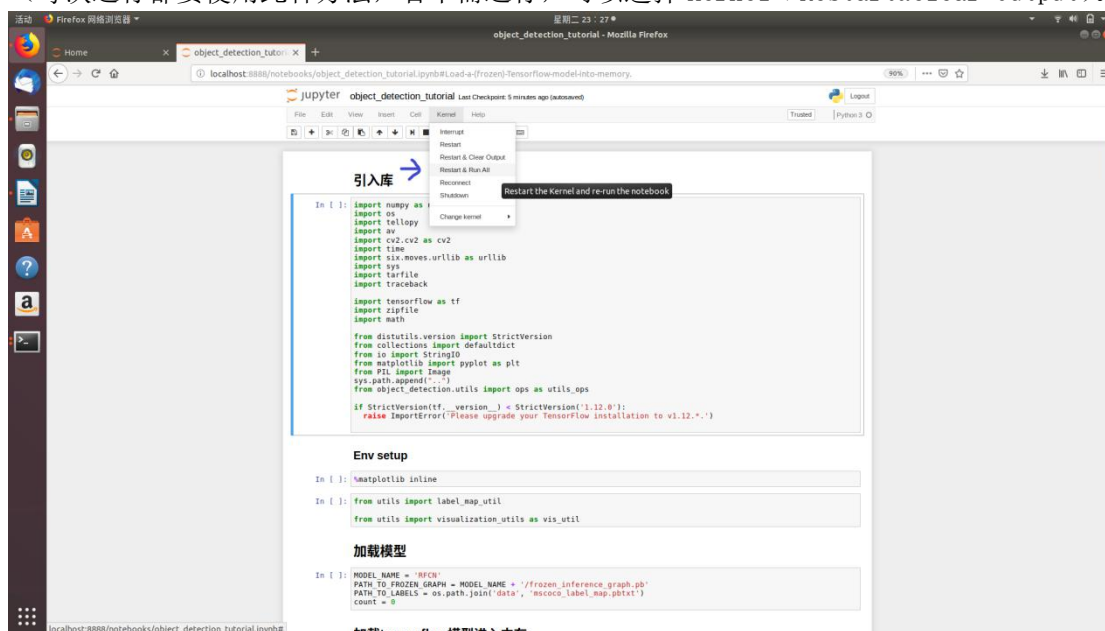


图 7 运行代码



## (6) 效果展示



图8 图像识别效果

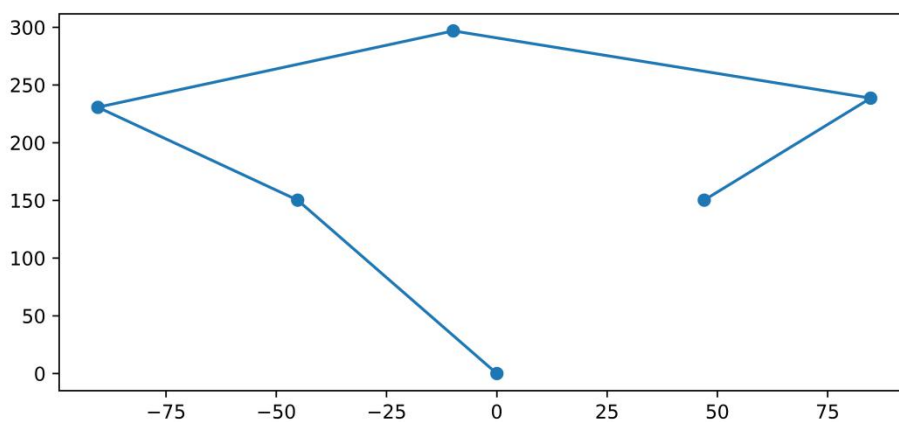


图9 路径规划效果

## 第六章 项目总结

在作品开发过程中，刚开始没有理清头绪，大家都不知道该做些什么。后面我们将所要做的事情都罗列出来后，才有了目标任务。队长主要负责组织协调，方案设计和技术实现，两个队员则一个主要负责在网上查询和我们作品相类似的开发产品以及产品的测试，另外一个队员主要负责作品创意和答辩方面的内容。大家各行其职，后面再汇总到一起，再统一改

编一下。在开发过程中遇到的困难有无人机在运行时所拍到并保存的照片在打开时为空白，在网上查阅资料后，改用了 opencv 方法将图片格式转换，解决了这一问题。后面在无人机检测时，无人机识别能力不够强，不能够准确的检测到垃圾，经过多次训练，才能较为准确的识别。起初我们没有加路径规划这一功能，后面发现如果不加清理垃圾的效率则会降低，通过路径规划，设计出最优路径，可以大大降低所需成本。通过这次作品开发，我们每一个成员都提升了自己的能力，接触到了更多关于人工智能领悟的知识，另外还学到了一门新语言——Python 语言，这次对我们每一个人而言，都是一次很好的锻炼，提升了个人水平。作品采用无人机完成垃圾清理作业，其优点在于可以依靠无人机搭建的摄像头获取图像，确定目标坐标，再进行路径规划，算出最佳路径，降低了成本，具有很大的实用性。