# 实验所完成的主要研究工作

## 主要研究工作概述

### 概述

近年来,计算机视觉技术领域取得了长足的进展,在目标检测、语义分割等方向都有大量出彩的技术出现,同时计算机视觉技术的落地与应用也愈发重要.本项目聚焦我国的视觉障碍人群,利用语义分割,光流估计等计算机视觉技术,构建基于双目相机的盲人户外智能语音辅助系统,帮助盲人更加安全便捷地进行户外行走.

首先,项目研究了室外语音辅助系统的搭建与实现,第一步构建手机终端程序,作用是采集定位信息与方位朝向信息.第二步构建服务端,运行盲道分割与偏移判断,障碍物检测与避障等核心代码,最后通过蓝牙耳机对用户进行偏离警示.

之后,对感知与规划过程进行了重点研究.利用自建盲道人行道数据集,通过轻量级实时语义分割网络bisenetv2进行训练,实现了对盲道与人行道的像素级语义分割.

### 系统核心功能

通过利用本系统,视障人士出行时得以选择安全且正确的行走路线,能够在语音的引导下走上盲道同时避开途中的障碍物,当遇到需穿过马路情景时能够引导用户在斑马线上通过马路,最终到达正确目的地。

本系统的核心技术包括感知,规划,决策,控制四个方面,其中重点与难点在于感知与规划.

## 基于神经网络的室外目标语义分割方法与后处理研究

图像语义分割任务是计算机视觉领域重要的基础研究任务，其任务是给定一张图像，要求算法为图像中的每个像素预测其所属的语义类别.而针对盲道与人行道进行语义分割是智能导盲系统的重要组成部分, 在兼顾实时性的同时提高盲道分割的效率和质量具有重大意义.

盲道与人行道作为特征较为明显的两类道路标识线,在输入图像较为清晰,背景环境简单,样本类型单一的情况下,利用二值化,canny边缘检测等传统图像处理算法能够取得较为良好的分割效果[1],且分割速度极快,但由于现实环境复杂多变,相机防抖的缺失,盲道种类繁多而传统图像处理方法依赖人工设计特征等原因, 而深度学习泛化能力较强,利用深度学习技术进行语义分割更加适宜本项目的需求.

### 系统感知需求

本系统的核心宗旨是让盲人尽可能的走在安全的地方,而盲道与人行道作为相对更加安全的区域是我们所提倡前往的,因此对于盲道与人行道的感知不能够仅仅停留在检测的程度,还需要进行进一步的语义分割,提取延伸方向,与抽象化区域等工作.

最终需要感知到的信息包括如下:

1. 盲道与人行道的走向
2. 盲道与人行道的可行走区域
3. 盲道起点与用户当前所处位置的角度与距离

此处主要使用相机获取的RGB图像信息.  
 进行规范化描述即为:

input:predicted\_img  
   
 output:  
 blind\_road\_find=1//是否发现盲道  
 blind\_road\_distance=0.5m//盲道与使用者的距离  
 blind\_road\_angle=-30°//盲道在使用者的左边30度角位置  
 on\_blind\_road=1//使用者是否位于盲道上  
 blind\_road\_departure//使用者是否正在偏离盲道  
   
 sideways\_find=1//是否发现人行道  
 sideways\_distance=3.5m//人行道与使用者的距离  
 sideways\_angle=30°//人行道在使用者的右边30度角位置  
 on\_sideways=1//使用者是否位于人行道上  
 sideways\_departure//使用者是否正在偏离人行道

### 基于 BiSeNet V2 网络模型的盲道与人行道分割

BiSeNet是一种用于实时语义分割的网络结构,于2018年由Yu[2]等人在ECCV2018上提出,其使用用于保留原图像信息的Spatial Path和用于获取较大感受野的Context Path,构成了一个小型且高效的双边分割网络.2020年Yu等人再次提出了BiSeNet V2[3]

我们针对盲道与人行道自身的特征及实际环境下采集的图像特点进行预处理以及数据增强，建立了盲道数据集,并利用该数据集基于BiSeNet V2进行训练,得到最终模型,其可以解决在满足实时性的基础上解决盲道与人行道的语义分割问题.

#### 实验数据集的构建

由于目前没有专门的大型盲道分割数据集, 因此我们采取了人工拍摄了盲道人行道图像与网络中盲道人行道图像数据集[4]相结合的方式,为了保证模型的泛化能力,我们采用了不同时间,不同天气,不同场景的盲道与人行道图像,利用Labelme进行标注,得到标注后的json文件, 通过编写python脚本批量将 json 文件转换为包含标签掩码的图片.

最终我们的数据集包括868张打标签的 512 × 512的图像,其中包含盲道标签的图像612张,包含人行道标签的图像412张,训练集:验证集:测试集=7:2:1,

以下是标注数据集的一些实例

#### 训练细节

在正式训练前,采用了数据增强技术,对数据进行随机垂直与水平翻转,随机图像截取,随机图像模糊,随机亮度,随机对比度,随机色调等处理.

损失函数是对系统在不同参数值下损失的描述，衡量的是模型的预测值与真实值的不一致程度，是衡量一个模型好坏的重要标准,我们选取交叉熵损失函数作为损失函数. 训练过程中选用的优化方法是随机梯度下降法(Stochastic Gradient Descent, SGD), 学习率动态调整方式为polynomial.

训练时的将Batch Size设置为4,学习率设置为0.005,迭代70轮.

#### 实验结果分析

总体来说,loss收敛良好,在验证集上miou达到0.9182,Kappa一致性检验值为0.936,一致性较好.

在验证集上miou达到0.9182

以下是测试集上的效果与标注效果比对,可见除较远处人行道难以识别外,模型效果良好. 在实际测试的推理过程中,每张图片平均预测用时约253ms,能够满足系统实际需求.

### 基于语义分割结果获取盲道与人行道走向

由于要平衡"走安全的路"与"走便捷的路"两种决策,为避免走安全的路却走在错误的方向的情况出现,需要获取盲道与人行道的走向,以实现行走大方向的不偏移.

由于语义分割后的结果是像素级别的分割,需要进行进一步分析出更多抽象化特征

对于盲道而言,由于是一整块区域,无需进攻更多处理,只需使用最常用的最小二乘法获取最优拟合直线即可.而对于人行道则需要进行更多处理.

#### 人行道走向获取

人行道检测存在以下典型应用场景:

考虑到盲人智能辅助系统的全局规划需求,我们前文进行语义分割得到的只是像素级别的类别判定, 未对人行道的走向分析与区域抽象化进行研究,因而只能够判定某区域是否存在人行道,当面临更加复杂的需求如对比人行道走向与全局路径时,便难以进行.

我们采用如下算法进行人行道区域与走向的提取方式:

Input:  
 语义分割预测结果图像  
  
 Algorithm:  
 初始化阈值bw\_height,bw\_width;  
 处理图像,过滤出只有人行道的二值图;  
 对图像进行侵蚀处理;  
 对此二值图像提取轮廓列表(道路候选点)contours;  
 for contour in contours:  
 获取轮廓contour最小外接矩形box;  
 如果 box.高>bw\_height 或 box.宽>bw\_width:  
 box[0]加入left\_line列表;   
 box[2]加入right\_line列表  
 利用RANSAC算法过滤与拟合道路边界点;

##### 道路候选点提取,过滤与拟合

语义分割后能够得到像素级别的类别判定, 由于像素颜色代表着分类标签,可以容易地过滤出只有人行道的二值图,通过侵蚀处理,对当前的二值图像通过光栅扫描进行轮廓提取[5],并求取每个轮廓的最小外接矩形的四个顶点坐标,分别作为左边界点与右边界点.此时便获取了人行道的候选点.

随机一致采样性(RANSAC)算法假设数据中包含正确数据和异常数据(或称为噪声),常中被用来直接过滤提取道路边界点后的噪点, 我们将以上获得的道路边界点用RANSAC曲线模型进行拟合,从而得到边界线与交点.

如下是对一个人行道区域进行抽象的完整过程

## 基于光流估计的盲道偏移检测研究

### 测角与测距

#### 双目立体视觉测距原理

在计算机视觉(computer vision)相关理论中，双目立体相机在同一时刻拍摄左右两幅图像对，通过提取图像特征点并进行立体匹配，利用三角关系计算目标点在左右图像中的视差，由此计算出环境的立体深度图，可以测量计算出环境物体的立体信息。

#### 测定物体与盲人的相对角度

### 静态判断盲人与盲道的相对位置

#### 坐标系定义

#### 盲人坐标系

#### 平面二维坐标系建立

#### 实时检测实验

在户外盲道上进行测试，判断算法的适用性。图5是人与盲道的相对位置的检测结果与数据

(a)位置1(b)位置2(c)位置3

图5 基于坐标系的盲道偏移检测

|  |  |  |  |
| --- | --- | --- | --- |
|  | 位置2 | 位置1 | 位置3 |
| 位置判定 | blind road is on your left | On the center | blind road is on your right |
| 盲道位置判定 | 不在盲道上 | 在盲道上 | 不在盲道上 |
| 偏移指数 | 1.6718 | 0.638822 | 3.57504 |
| 真实情况 | 在盲道右侧 | 在盲道中间 | 在盲道左侧 |

表1 定位结果

### 基于视频判断盲人的行走偏移趋势

#### 连续点偏移策略

为了更好地矫正盲人的行走路线，会抽取连续三帧对盲人的行走趋势进行判定。如果盲人连续三帧向盲道的一侧走去，就判定盲人有偏离盲道的趋势，并语音播报矫正盲人。

如果盲人未在盲道内，就判断盲道在盲人的哪一侧，并语音播报指示给盲人听。

1. 获取输入的连续三帧数据

盲人的行为是动态的，不能通过单单的一帧去判断盲人的动向，因为正常人走路是摇晃的，并不会呈直线走路，所以只能通过连续的三帧图像去粗略判断盲人的大致走向。

(a)位置1(b)位置2(c)位置3

图6 偏移趋势测定

2. 给出偏移趋势

行人在这三帧中是不断向右偏移的，所以判定为行人有偏离盲道的趋势，并向盲人提出矫正。

|  |  |  |  |
| --- | --- | --- | --- |
|  | 位置1 | 位置2 | 位置3 |
| 偏移方向 | 右 | 右 | 右 |
| 偏移指数 | 0.785993 | 0.822911 | 0.82395 |
| 真实情况 | 偏向右边 | 偏向右边 | 偏向右边 |

表2 偏移数据

#### 基于稀疏光流的盲人行走偏移趋势估计

光流[1]法是利用图像序列中像素在时间域上的变化以及相邻帧之间的相关性来找到上一帧跟当前帧之间存在的对应关系，从而计算出相邻帧之间物体的运动信息的一种方法.

其基于视频连续图像中像素亮度一致性和相关性以及邻域元素结构相似性等原则,估计相邻帧中像素的对应关系,求取物体在成像平面中的运动矢量,而稀疏光流则是对图像稀疏特征点的运动跟踪。

考虑到算法的实时性,相较于稠密光流,稀疏光流能够更加满足实时性的需求,我们采用Lucas-Kanade算法进行计算稀疏特征集的光流.

##### 准备工作

考虑到本系统所要求的实时性,我们希望光流判断用时要少于0.15s,且不易受移动物体的影响.

为了方便统计光流向量,我们生成HOF光流直方图进行可视化特征.即计算光流向量与横轴的夹角,计算的夹角结果分配到各自的区间（这里面每**30度**定义一个区间）,即如下图所示,将0-30度作为0类,30-60度作为1类向量,自从得到12类特征向量.

考虑到在盲道行走场景中,图像是运动的,图像中的像素点的向后移动(我们将本分类向量中即**2类与3类向量**近似看作像素点向后移动)即代表着相机(盲人)的向前移动.

此外,我们假设正常情况下盲人不会倒退行走,因此对于像素点向前偏移的矢量,我们更加关注其水平分量.

因此我们将类别为0,1,10,11,12的特征向量代表像素点向左偏移,即人向右偏移.

同理我们将类别为4,5,6,7,8,9的特征向量代表像素点向左偏移,即人向右偏移.

得到类如下图的光流直方图

##### Lucas-Kanade算法

Bruce D. Lucas 和 Takeo Kanade在1981年提出的Lucas Kanade（LK）算法成为了稀疏光流算法的重要技术，它仅依赖于围绕某个兴趣点的一些小窗口导出的局部信息.

LK算法是一种两帧差分的光流估计算法，其基本思想基于以下三个假设。

**亮度恒定**：场景中目标图像的像素看起来在帧到帧移动是不发生改变。对于灰度图像（对于彩色图像同适用）这意味着像素的灰度值不会随着帧的跟踪改变。

**时间持续性**：图像上相机的移动随时间变化缓慢。实际上，这意味着时间的变化不会引起像素位置的剧烈变化，这样像素的灰度值才能对位置求对应的偏导数。

**空间一致性**：场景中相同表面的相邻点具有相似的运动，并且其投影到图像平面上的距离也比较近。

后人提出用图像金字塔的方法来解决大运动的跟踪的问题, 结合Lucas-Kanade方法,通过建立金字塔,在多尺度下计算光流, 金字塔光流的思想是对每个图像帧向下采样,分别建立多级金字塔,当采样到足够小以后,相邻图像帧之间的运动将变得很小,以至于可以看成是物体的运动随时间变化很缓慢的情况,这时候就可以用Luacs-Kanade方法计算目标的光流,再将计算出来的光流向底层投影,计算下一层的光流,直到估算出原图像帧的光流。

##### 特征点选取策略

为了推断出更多规律,我们首先选取**完全直线行走**场景进行测试

为了寻找能够更好描述场景运动的特征点选取方法,我们采用如下特征点选取策略进行测试与对比:

\3. Shi-Tomasi角点检测[2]并选取最多100个强角点

\4. 遮挡除盲道外的区域,角点检测选取最多100个强角点

\5. 盲道正中心的+-200像素,间隔20

\6. 盲道区域内随机选取100个特征点

\7. 1920*1280分割成16个480*320的小块,考虑到行走过程中像素点的移动可能会偏离图像区域,只选取中间9个的区域进行光流判断

\8. 1920*1280分割成128个120*160的小块,选取中心进行判定

此外为了尽量减少光流计算错误的情况,我们去除横向或纵向移动太大的特征向量.正如前文所讨论的我们将2或3类向量为占比最高量看做盲人向前直线行走的标志.

##### 特征点选取策略结果分析

我们选取直线行走的三帧并统计其判断直线准确率与运行速度,正确率表格如下

|  |  |  |
| --- | --- | --- |
| **选取策略** | **正确率** | **用时(s)** |
| 角点检测选取最多100个强角点 | 0.33 | 0.02 |
| 遮挡除盲道外的区域,角点检测选取最多100个强角点 | 0 | 0.02 |
| 盲道正中心的+-100像素,间隔20 | 0.33 | 0.04 |
| 盲道分为3块,3块中心的+-50像素,间隔20 | 0.66 | 0.04 |
| 取盲道区域10\*25 | 0.66 | 0.05 |
| 取盲道区域20\*50 | 1 | 0.1 |
| 全局分块为20\*50 | 0.66 | 0.1 |
| 全局分块为20\*100块 | 0.33 | 0.26 |

考虑到本系统所要求的实时性,我们希望光流判断用时要少于0.15s,另外考虑到摄像机拍摄距离有限,盲道具有更高的辨识度,同时不易受移动物体的影响,我们对**取盲道区域特征点**的策略进行进一步研究.

部分选取策略详细过程记录

###### Shi-Tomasi角点检测并选取最多100个强角点

###### 遮挡除盲道外的区域,角点检测选取最多100个强角点

###### 取盲道区域20\*50

###### 全局分块为20\*50

##### 选取盲道区域特征点策略的研究

## 基于深度图的障碍物感知与避障策略研究

### 基于深度图的障碍物感知研究

#### 算法步骤

### 基于RRT的避障策略与路径规划研究