人工智能课程报告

韩瑞琛 2015202030

**一.背景知识：**

Arduino：Arduino能通过各种各样的传感器来感知环境，通过控制灯光、马达和其他的装置来反馈、影响环境。板子上的微控制器可以通过Arduino的编程语言来编写程序，编译成二进制文件，烧录进微控制器。对Arduino的编程是通过 Arduino编程语言 (基于 Wiring)和Arduino开发环境(基于 Processing)来实现的。基于Arduino的项目，可以只包含Arduino，也可以包含Arduino和其他一些在PC上运行的软件，他们之间进行通信 (比如 Flash, [Processing](/item/Processing" \t "_blank), MaxMSP)来实现。

深度学习： 深度学习是[机器学习](/wiki/%E6%9C%BA%E5%99%A8%E5%AD%A6%E4%B9%A0" \o "机器学习)中一种基于对数据进行[表征学习](/wiki/%E8%A1%A8%E5%BE%81%E5%AD%A6%E4%B9%A0" \o "表征学习)的算法。观测值（例如一幅图像）可以使用多种方式来表示，如每个像素强度值的向量，或者更抽象地表示成一系列边、特定形状的区域[等](/wiki/%E5%B0%BA%E5%BA%A6%E4%B8%8D%E8%AE%8A%E7%89%B9%E5%BE%B5%E8%BD%89%E6%8F%9B" \o "尺度不变特征转换)。而使用某些特定的表示方法更容易从实例中学习任务（例如，人脸识别或面部表情识别[[6]](" \l "cite_note-6)）。深度学习的好处是用[非监督式](/wiki/%E9%9D%9E%E7%9B%A3%E7%9D%A3%E5%BC%8F%E5%AD%B8%E7%BF%92" \o "非监督式学习)或[半监督式](/w/index.php?title=%E5%8D%8A%E7%9B%91%E7%9D%A3%E5%BC%8F%E5%AD%A6%E4%B9%A0&action=edit&redlink=1" \o "半监督式学习（页面不存在）)（英语：[Semi-supervised learning](https://en.wikipedia.org/wiki/Semi-supervised_learning" \o "en:Semi-supervised learning)）的[特征学习](/wiki/%E7%89%B9%E5%BE%81%E5%AD%A6%E4%B9%A0" \o "特征学习)和分层[特征提取](/wiki/%E7%89%B9%E5%BE%81%E6%8F%90%E5%8F%96" \o "特征提取)高效算法来替代手工获取[特征](/w/index.php?title=%E7%89%B9%E5%BE%81_(%E6%9C%BA%E5%99%A8%E5%AD%A6%E4%B9%A0)&action=edit&redlink=1" \o "特征 (机器学习)（页面不存在）)（英语：[Feature (machine learning)](https://en.wikipedia.org/wiki/Feature_(machine_learning)" \o "en:Feature (machine learning))）。

[表征学习](/wiki/%E8%A1%A8%E5%BE%81%E5%AD%A6%E4%B9%A0" \o "表征学习)的目标是寻求更好的表示方法并创建更好的模型来从大规模未标记数据中学习这些表示方法。表示方法来自[神经科学](/wiki/%E7%A5%9E%E7%BB%8F%E7%A7%91%E5%AD%A6" \o "神经科学)，并松散地创建在类似[神经系统](/wiki/%E7%A5%9E%E7%BB%8F%E7%B3%BB%E7%BB%9F" \o "神经系统)中的信息处理和对通信模式的理解上，如[神经编码](/wiki/%E7%A5%9E%E7%BB%8F%E7%BC%96%E7%A0%81" \o "神经编码)，试图定义拉动神经元的反应之间的关系以及[大脑](/wiki/%E5%A4%A7%E8%84%91" \o "大脑)中的神经元的电活动之间的关系。

至今已有数种深度学习框架，如[深度神经网络](/wiki/%E6%B7%B1%E5%BA%A6%E5%AD%A6%E4%B9%A0" \l "%E6%B7%B1%E5%BA%A6%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C" \o "深度学习)、[卷积神经网络](/wiki/%E5%8D%B7%E7%A7%AF%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C" \o "卷积神经网络)和[深度置信网络](/w/index.php?title=%E6%B7%B1%E5%BA%A6%E7%BD%AE%E4%BF%A1%E7%BD%91%E7%BB%9C&action=edit&redlink=1" \o "深度置信网络（页面不存在）)（英语：[Deep belief network](https://en.wikipedia.org/wiki/Deep_belief_network" \o "en:Deep belief network)）和[递归神经网络](/wiki/%E9%80%92%E5%BD%92%E7%A5%9E%E7%BB%8F%E7%BD%91%E7%BB%9C" \o "递归神经网络)已被应用在[计算机视觉](/wiki/%E8%AE%A1%E7%AE%97%E6%9C%BA%E8%A7%86%E8%A7%89" \o "计算机视觉)、[语音识别](/wiki/%E8%AF%AD%E9%9F%B3%E8%AF%86%E5%88%AB" \o "语音识别)、[自然语言处理](/wiki/%E8%87%AA%E7%84%B6%E8%AF%AD%E8%A8%80%E5%A4%84%E7%90%86" \o "自然语言处理)、音频识别与[生物信息学](/wiki/%E7%94%9F%E7%89%A9%E4%BF%A1%E6%81%AF%E5%AD%A6" \o "生物信息学)等领域并获取了极好的效果。

**二.项目情况介绍：**

**1.项目目标：**

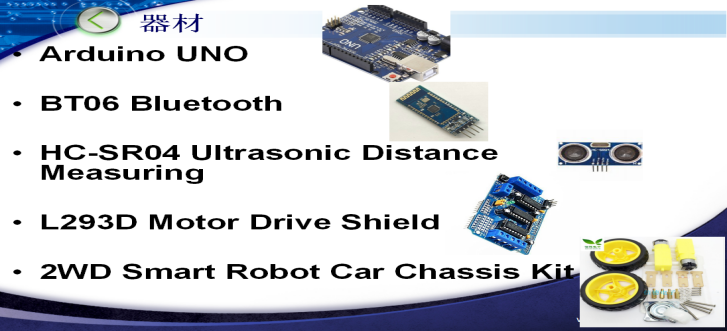
我们组的实验研究的主要内容是通过完成智能小车实现对深度学习的算法实践，我们组的实验阶段可分为1.完成对智能小车的组装并能实现小车能够按照随机方向行走，并能通过超声波传感器的信息避开障碍物。 2通过手机上的蓝牙串口模块，将智能小车和手机通过蓝牙连接，来实现通过手机对智能小车的实时监控和控制，小车通过超声波传感器获得数据会传回手机，手机可以通过控制小车的转向、启动和停止，我们组后面以讯飞科技语音识别为内核改写了手机蓝牙串口，使得我们可以通过识别出使用者说出的指令中某一个关键字或关键词如“左”，来实现对语音控制智能小车。 3.通过智能小车上的手机摄像头和手机与电脑上蓝牙连接，使电脑能实时依托获得手机摄像头拍摄的画面，依托神经网络模型进行物像识别，再通过小车上安装的机械臂实现寻找物体，抓取物体的过程。

**2.项目具体实现：**

**2.1 第一阶段：**

**2.1.1 材料准备：**

我们组准备的材料清单如下：



以及面包板、导线和亚克力机械臂。

在安装小车的过程中，我们组用到的Arduino上的接口与功能如下：

（1）digital pin 0(RX)：接收来自计算机的指令信息，与蓝牙串口模块上的TXD相连。

（2）digital pin 1(TX)：发送指令信息给计算机，与蓝牙串口模块上的RXD相连。

以上两个接口用作手机蓝牙与Arduino交换信息。

（3）VCC：电源接口，相当于正极。

（4）GND：电源接口，相当于负极。

以上两个接口用作与充电宝相连，为Arduino板及电机供电。

**2.1.2 组装小车：**

我们小组在组装小车的过程中采用的是先将小车分成几个独立的部分，分别测试各部分的功能正确后，再将各部分通过焊接等手段组成一个完整的整体。

小车组装完成后：



**我在此过程中参与了小车的组装和对uno板和Arduino代码调试。**

**2.1.3 利用Arduino实现小车自动随机行走和避开障碍物**

我们的Arduino的代码主要是用老师提供的代码的基础上进行修改完成，在对uno板的调试过程中，我们发现我们小车的右轮无法运转，我们通过不断的测试各接口，后发现AF\_DCMotor Rback\_motor(4);中的我们uno板的M4接口是损坏的，所以我们将其中的4改为了2，而且在运转过程，发现右轮的运转速度低于左轮许多，于是我们通过不断调整RLRatio参数的值，使得两轮的速度达到了较好的平衡。

**2.2 第二阶段：**

**2.2.1 加入蓝牙功能**

我们在安卓手机上下载了蓝牙串口助手的app，在一开始的测试中出现了蓝牙与超声波传感器同电机驱动板连接有bug，超声波传感器无法读出距离、蓝牙可以连接但控制不了等问题，我们后面发现这是因为Uno板外部接口已与电机驱动器连接，所以我们将传感器连接在uno板的内部接口，成功解决了这个问题，并且能够通过手机发送前进，左转，右转等基础功能。

**2.2.2 加入语音识别**

我们在实验过程中发现通过蓝牙串口助手实现对小车的控制似乎不是很方便，于是我们组就想通过加入语音识别的功能，先通过识别出施令者话中的关键字转换成相应的指令，再通过蓝牙发送给小车，来实现对小车的远程控制，韩瑞琛同学在搜索语音识别的相关资料时发现讯飞科技对语音识别的研究已经比较的成熟，我们决定以讯飞科技的语音识别技术SDK为内核，结合蓝牙串口助手，编写出了新的手机app，当施令者对手机说出指令后，讯飞科技的语音识别技术SDK就会根据接收到的声波在其数据库中进行搜索，我们编写的新的app根据其搜索后返回的文字结果，通过匹配这些文字中是否有关键字“左”等，再将匹配的结果转换成相应的数字指令发送给小车来实现通过语音对小车的语音控制。

**我在以上两步工作中，参与一开始的加入的蓝牙功能的测试和调整工作，在加入语音识别时，找到了讯飞科技的语音识别技术SDK，并在新的app的编写过程中参与了调试和测试的工作。**

**2.2.3 图像识别的前期准备工作**

通过在网上搜索和学习一些图像识别的简单代码，并准备了用于图像识别的训练集。

**2.3 第三阶段**

**2.3.1 图像识别：**

我们组利用Tensortflow提供的Object Detection API提供的常用的模型训练出可以识别出例如鸡蛋等比较小的数十种较小的静止物体的深度神经网络模型，因为我们组希望能最终实现用机械臂对识别出的物体进行抓取的操作，所以考虑到我们组的小车的能力的限制，该深度神经网络模型最为适宜，对于我们用于测试的图片，该模型的物体识别函数能够完成识别图片中的物体，并对图片进行重新渲染，使能够显示出物体的名称额预测可能性的指数。

**2.3.2 对实时视频中的物体进行识别：**

我们组先在装载在小车上手机上安装能通过wifi实时传输手机摄像头所拍摄的图像的软件，并获取了视频流的URL地址，然后利用OpenCV提供的VideoCapture接口读入这个URL地址的视频流内容，在这个过程中，我们组遇到的最大的问题就是实时传输的延迟比较大，使得实验结果与我们所期望的还有差距，因为wifi的传输速度以及深度网络模型识别速度的限制，使得渲染图像的延迟较大接收到实时传输过来的视频后，利用MoviePy提供的VideoFileClip接口读入视频片段，并用物体识别函数为参数调用fl\_image()函数，通过以上的操作，就可以用每一个重新选然后的帧替换原来的帧，从而得到一个动态识别物体的视频片段。

**我在这一阶段的工作的参与中学习和使用了Tensortflow，感受到了Tensortflow功能的强大和便捷，也对深度学习有了一定的了解和认识。**

**三.学习总结**

在本门课程的学习中我最大的感受就是这个项目的完成过程中，大家一起共同学习，共同进步，因为这个项目的完成既基于老师课上所教授的知识，但又高于课堂，需要我们自己去不断补充课外的新的知识，因为我们在完成过程中一起不断为小车添加新的功能，而这需要我们自己去学习实现。因为在完成课程报告前我们的安装机械臂和抓取物体的功能还未实现，所以未写入课程报告，希望我们在接下来的一周能顺利实现这一功能。

对于课上所学习的各种模型，我只做了一些简单的尝试而没有做深入的学习，以下是一些实验截图：

