**赋能垃圾分类系统**

**说明文档**

**目 录**

[第1章 前言 3](#_Toc100605403)

[1.1 项目背景及意义 3](#_Toc100605404)

[1.2 国内外研究现状 3](#_Toc100605405)

[1.3 总体方案 3](#_Toc100605406)

[1.4 文章组织结构 3](#_Toc100605407)

[第2章 系统分析 4](#_Toc100605408)

[2.1 可行性分析 4](#_Toc100605409)

[2.1.1 系统实施的影响分析 4](#_Toc100605410)

[2.1.2 经济可行性分析 4](#_Toc100605411)

[2.2 需求分析 5](#_Toc100605412)

[2.3 技术分析 5](#_Toc100605413)

[第3章 概要设计 6](#_Toc100605414)

[3.1 系统结构 6](#_Toc100605415)

[3.2 系统功能模块设计 6](#_Toc100605416)

[3.3 创新要点 6](#_Toc100605417)

[第4章 系统部署与维护 7](#_Toc100605418)

[4.1 系统安装与部署 7](#_Toc100605419)

[4.1.1 开发工具与技术 7](#_Toc100605420)

[4.1.2 应用对象与环境 7](#_Toc100605421)

[4.1.3 系统维护 7](#_Toc100605422)

[4.2 系统测试 7](#_Toc100605423)

[4.2.1 测试环境 7](#_Toc100605424)

[4.2.2 测试方法 8](#_Toc100605425)

[4.2.3 测试结果分析 8](#_Toc100605426)

[第5章 结语 12](#_Toc100605427)

# 第1章 前言

## 1.1 项目背景及意义

随着社会的快速发展，垃圾存量急剧上升，“垃圾围城”、“垃圾围村”等问题日益凸显。困扰中国的各个城市、乡村。垃圾分类是人类社会进步和生态文明的标志，是人人均可参与其中来保护环境和改善环境的方式。2017年3月，国家发改委、住建部发布《生活垃圾分类制度实施方案》，要求在全国46个城市先行实施生活垃圾强制分类。为深入贯彻落实习近平总书记关于垃圾分类工作的重要指示精神。针对目前垃圾分类种类繁杂、垃圾数量巨大，人工挑拣分类的难度大、速度慢等问题，我们设计出赋能垃圾分类检测系统，系统结合深度学习的相关技术，通过对待检测垃圾图片的识别和检验，实现不同种类垃圾的智能分类，并且对垃圾分类知识进行科普。系统能够在一定程度上实现垃圾的智能分类，这不仅减少了资源的浪费，而且对垃圾分类在群众中广泛普及起到了推动作用。

## 1.2 国内外研究现状

通过对国内垃圾分类政策的了解以及团队对垃圾分类现状的调研，深度学习能够有效地解决传统机器学习中手工特征提取不全和选择繁琐的问题，挖掘样本数据的内在规律，并带来性能的改善，广泛应用于计算机视觉[1]、自然语言处理[2]、数据挖掘[3]等领域。因此，基于深度学习的垃圾分类方法为垃圾分类指出了一条新的道路。

## 1.3 总体方案

我们设计的赋能垃圾分类系统是针对垃圾分类落地困难、管理难度大等问题提出的。我们计划将系统分为单张垃圾图片检测，多张垃圾图片检测以及视频检测等三个主要功能。首先，对于单张垃圾图片检测，我们将设计四个主要功能模块，分别为图片上传，图片识别，识别结果显示以及垃圾分类知识科普模块。其次，在多张垃圾图片检测中将对上传的文件夹中的多张图片进行批量化检测，并且对检测结果进行一定的统计与分析。最后，该系统将实现一个视频监测的功能，包括调用摄像头的实时性监测和本地视频上传系统进行检测。

## 1.4 文章组织结构

本文对赋能垃圾分类检测系统的构思、设计与实现进行详细的讲解。第一章，详细说明了赋能垃圾分类检测的背景和意义，第二章，对整个系统的可行性分析、需求分析、技术分析方面做出详细阐述，第三章，根据需求分析进行系统功能模块的设计，第四章，完成开发、部署及测试，第五章是对整个项目总结与思考。

# 第2章 系统分析

## 2.1 可行性分析

### 2.1.1 系统实施的影响分析

**1) 对社会的影响**

项目的开发与实现，符合当下社会对互联网+时代的要求，合理利用了社会资源，对社会的发展与进步起到了促进作用。

**2) 对健康的影响**

此次项目的开发与设计就是为了解决垃圾分类不当、堆积等问题对环境和人类健康造成的不利影响，并且整个项目提倡传播健康的信息。

**3) 对法律的影响**

系统的实施严格遵守中华人民共和国的法律法规，并倡导公民遵纪守法，顺应法律的约束与要求，不会对法律造成任何负面影响。

**4) 对文化的影响**

整个项目预期达到的界面效果，符合当下民众的审美，顺应文化的发展，对互联网乃至人类文化的发展起到了促进作用。

**5) 对环境的影响**

项目的实施是基于互联网环境，不会对自然资源造成破坏或浪费，而此次项目安全健康，符合国家网络安全法的要求与规范，更是能对人类环境起到维护和促进作用，有效推进垃圾分类落地和提升城市管理水平。

### 2.1.2 经济可行性分析

**1) 项目开发的成本分析**

整个项目在开发的过程中不会浪费资源，在开源的环境下进行系统的设计与实现，故整个项目的开发成分低，项目值得实施。

**2) 项目实施的成本分析**

整个项目在实施时，花费很少的人力，并节约物力的基础上完成，项目历时三个多月，实施成本较低。

**3) 项目维护的成本分析**

项目在后期只需要花费很少的人力和物力就能够维护，主要是对系统软硬件的维护、代码的维护、数据文件的维护、程序的维护。一个管理员和计算机设备就能完成，成本低。

**4) 项目实施的经济效益分析**

项目迎合当下垃圾分类政策的需求，在花费很少的开发、实施、维护成本的同时，能够有效提升垃圾分类的效率，合理利用互联网资源，避免资源浪费，综合来看，整个项目经济效益可观。

# 第2章 需求分析

垃圾分类系统实现的是对垃圾的分类功能，需要实现的功能模块主要分为两个模块：图片检测和视频检测。

## 2.1 图片检测

在图片检测功能模块中，我们需要根据应用在不同的场景中设置两种检测模式——单张图片检测和多张图片检测。

### 2.1.1 单张图片检测

在单张图片检测模式中，我们将需要对输入到系统的图片先进行展示，然后再将图片送入检测器中。检测器获取图片之后，应该首先判断图片中是否存在垃圾，如果存在的话将会对图片中的物品进行检测，反之则直接返回图片到系统，系统将其直接输出并告知用户检测失败。当检测完成后，检测器能够能到图片中的物体名称和物品类别得分，并且将物品的预测框画在图片上，随后将图片返回给系统。系统接收到图片之后将图片放置在检测结果区域，同时，系统将会显示出垃圾类别和物品名称。考虑到对大众进行垃圾分类的普及，系统应该在GUI界面中显示该类垃圾的相关知识。在检测完成之后，系统应该将检测器检测到的图片进行保存。

### 2.1.2多张图片检测

在多张图片检测模式中，我们将需要把需要检测的图片放到一个文件夹中。将待检测图片放入文件夹之后，系统应该让用户自行选择文件夹，这样也能够方便用户的操作。当用户选择好待检测图片所在的文件夹之后，系统应该对文件夹中的图片进行遍历，逐一送入文件夹图片检测器中。由于多张图片无法确定数量，因此无法显示在GUI界面上，因此多张图片检测模式采用不同的检测器进行检测，以返回需要保存的内容。该检测器首先会判断图片中是否存在垃圾，如不存在则会对其物品名称值进行标记，如果存在垃圾的话将会把带有预测框的图片，每张图片中的物品名称，物品名称得分，垃圾预测框位置坐标返回。为了便于分析，检测结果应该以两种方式进行保存，一种以带有预测框的图片的方式保存在文件夹中，另一种将得到的数据保存在CSV文件中，便于后续分析。

## 2.2 视频检测

在图片检测功能模块中，我们需要根据使用的情况不同设置两种检测模式——视频检测和摄像头监测。

### 2.2.1 视频检测

在视频检测模式中，系统应该让用户能够自己选定需要检测的视频文件进行检测。在视频检测过程中，用户应该能够看到视频的检测过程。同时，在视频播放过程中应该告诉用户如何进行保存。视频播放完毕之后，系统应该将检测之后得到的视频进行保存。

### 2.2.2 摄像头监测

在摄像头监测模式中，系统将能够实时的对摄像头所捕捉到的画面进行监测。同时，用户能够自行决定监测时长。在监测结束之后，系统应该将监测得到的视频进行保存。

# 第3章 算法与关键模块架构

## 3.1 系统总体结构

垃圾分类系统总体架构如图1所示。赋能垃圾分类系统总体分为两个部分，分别为训练部分和GUI部分。



图1 垃圾分类总体架构图

训练部分的主要模块分为三个——对标注好的数据集进行处理，使用YOLOX-CBAM对数据集进行训练和将模型输出。

GUI界面有两大模块，分别为图片检测模块和视频监测模块。在两大模块之下，分别都有着两种不同的模式，其中图片检测模块有单图检测模式和多图检测模式，视频检测模块有视频检测模式与摄像头监测模式。

## 3.2 算法架构

本系统主要使用的算法为基于Pythorch的YOLOX算法模型。为提升检测精度，系统在原YOLOX模型中融入了CBAM注意力机制。通过添加注意力机制，模型的检测准确率提升了0.56%。

### 3.2.1 YOLOX算法

YOLOX的算法模型图如图2所示。



图2 YOLOX算法模型

YOLOX中提取特征的骨干网为CSPDarkNet。首先，CSPDarkNet中使用了残差网络Residual，残差网络的特点是容易优化，并且能够通过增加相当的深度来提高准确率。其内部的残差块使用了跳跃连接，缓解了在深度神经网络中增加深度带来的梯度消失问题。其次，CSPDarkNet使用CSPnet和Focus两种网络结构。除外，使用了SiLU激活函数，SiLU是Sigmoid和ReLU的改进版。SiLU具备无上界有下界、平滑、非单调的特性。SiLU在深层模型上的效果优于ReLU。可以看做是平滑的ReLU激活函数。最后，骨干网使用了SPP结构，通过不同池化核大小的最大池化操作进行特征提取，以提高网络的感受野。

YOLOX的特征融合网络采用了PANet，用于加强特征提取的效果。PANet是传统金字塔网络FPN的改进版，它的特点就是对特征进行反复提取，其结构在完成FPN从下到上的特征提取后，还要实现从上到下的特征提取。

目标检测网络（Yolo Head）将进行三个不同尺度的预测。Yolo Head与YOLOv4所使用的版本不同，原本的分类和回归在一个1×1卷积里实现，YOLOX则将其分为两个部分，在最后预测的时候才整合在一起。最终对于每一个特征图（三个不同尺度）可以获得三个预测结果，分别为：Reg(用于判断每一个特征点的回归参数)、Obj(用于判断每一个特征点是否包含物体)和Cls(用于判断每一个特征点所包含的物体种类)。将Reg、Obj和Cls的结果进行堆叠后即可获得的预测结果。对预测结果进行解码操作后即可得到最终绘制于图片上的预测框。

### 3.2.2 CBAM注意力机制

CBAM算法来源于2018年发表在ECCV上的一篇论文CBAM: Convolutional Block Attention Module。CBAM是一种简单卷积神经网络注意力模块，能够有效的提高CNN的表达能力。在卷积神经网络任意给定一个中间特征图，CBAM将注意力映射沿特征图的通道与空间两个独立的维度进行注入然后将注意力乘以输入特征映射，对输入的特征图进行自适应特征细化，如图3所示。



图3 CBAM注意力机制网络结构图

通过实验，我们得出在YOLOX的CSPDarknet的网络输出使用CBAM注意力机制能够有效的提升模型的精度，改进后的网络结构图如图4所示。



图4 YOLOX-CBAM架构图

## 3.3 关键模块架构

系统的GUI模块主要有两个部分——图片检测模块和视频检测模块，模块架构图如图5所示。在运行GUI模块之前，系统会导入我们使用YOLOX-CBAM在自制垃圾数据集上训练得到的模型。



图5 模块架构图

### 3.3.1 图片检测

1) 单图监测

在单图检测模式中，我们设计了图片上传、图片检测、结果显示以及垃圾分类知识科普四大功能模块。图片上传功能可供用户将待检测图片上传至系统用于后续的检测任务。图片检测功能模块是整个GUI模块中最为关键的功能模块，针对任务不同，检测功能模块也会有一定区别。图片检测功能模块能够检测出系统所给出图片中的物品名称、垃圾种类以及物品位置预测框。检测结果显示功能模块将检测器的输出显示在GUI界面上，方便用户查看检测结果。垃圾分类知识科普功能主要是针对检测出来的垃圾类别的相关信息进行知识科普。

2) 多图检测

在多图检测模式中，我们设计了三个功能模块，分别为选择图片文件夹、文件夹图片检测和保存结果。选择图片文件夹功能模块将能够让用户把含有待检测图片文件的文件夹。文件夹图片检测模块能够将所选择文件夹中图片送入检测器，并将结果返回。在文件夹中图片检测结束之后，结果保存功能模块会将结果分别保存在CSV文件和文件夹中并进行分析。

### 3.3.2 视频检测

1) 本地视频检测

在本地视频检测模式中，我们设计了三个功能模块——选择视频文件、视频画面检测以及结果保存。选择视频文件功能模块能够选择待检测视频文件。视频画面检测功能模块会将视频画面送入检测器，检测器输出结果后再返回进行显示。结果保存功能模块将会把视频检测得到的画面保存到特定文件夹中。

2) 摄像头监测

在摄像头监测模式中，我们设计了三个功能模块——打开摄像头、摄像头画面监测以及结果保存。打开摄像头功能模块能够打开摄像头开始捕获画面。摄像头画面监测功能模块会将视频画面送入检测器，检测器输出结果后再返回进行显示，进行实时监测。结果保存功能模块将会把摄像头监测得到的画面保存到特定文件夹中。

# 第4章 开发与功能实现

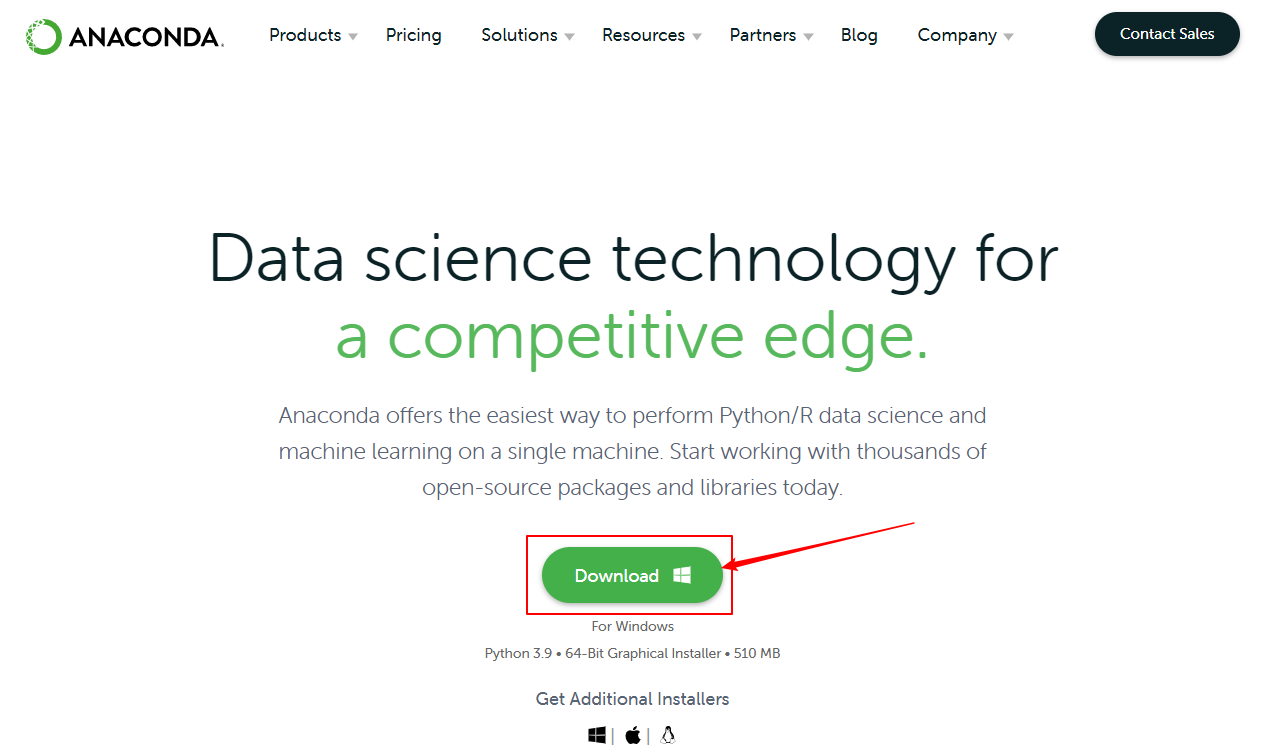
项目运行系统为Ubuntu 20.04，并使用4块GeForce GTX 1080 Ti显卡对模型进行训练。项目开发人员要求掌握深度学习相关知识，能够熟练应用python编程语言。

## 4.1 运行环境安装与搭建

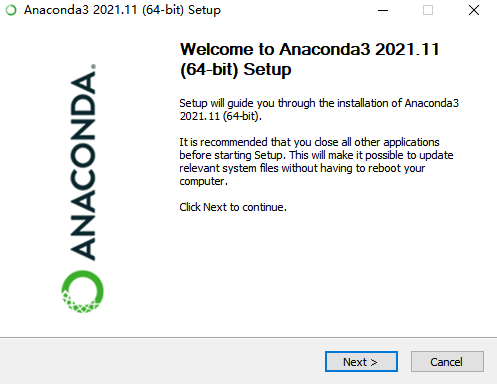
### 4.1.1 Anaconda下载及安装

由于在本机上之间安装python容易导致本机python环境变得十分混乱，本项目采用Anaconda软件配置python的虚拟环境并进行管理。

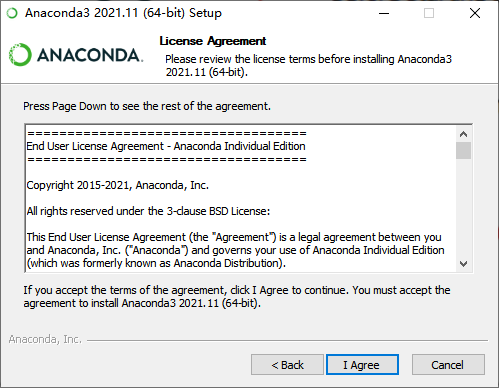
首先打开Anaconda官网（<https://www.anaconda.com/>），进入官网后如下图所示，点击Download即可开始下载。



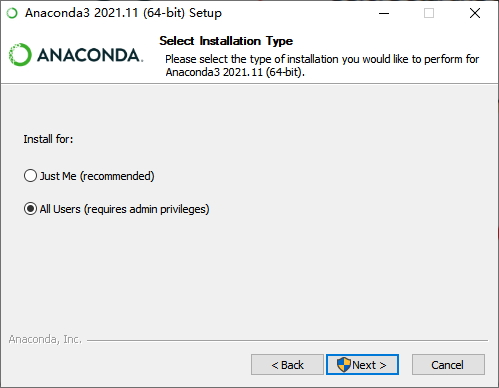
下载完成之后点击exe文件进行安装，如下图所示。



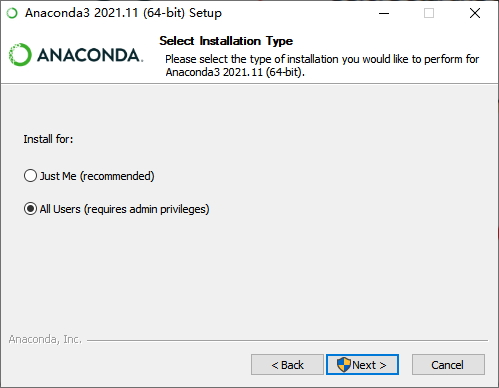
点击Next进入如下界面。



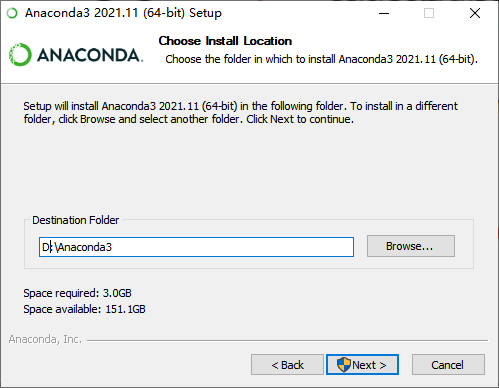
在此选择I Agree，之后点击Next。



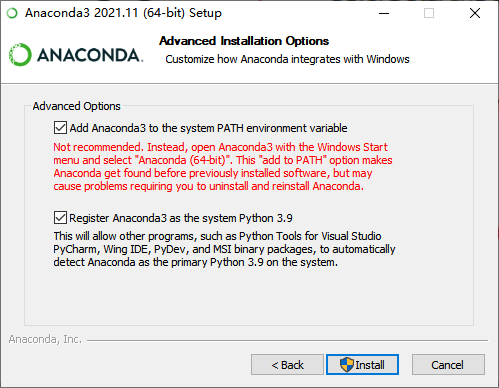
在此选择All-Users，之后点击Next。



自行选择安装位置（尽可能不要安装在C盘中，以节省系统盘的空间资源）。

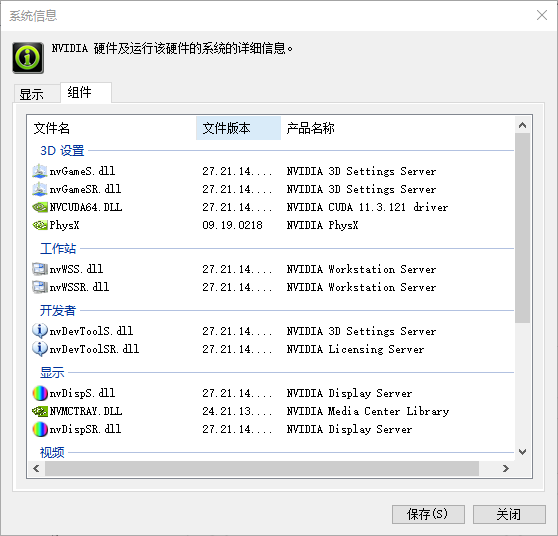


将两个框框都打上勾，能够节省手动添加环境变量的麻烦，再点击Install，接下来耐心等待，安装完成之后一直点Next就安装完成了。



### 4.1.2 安装CUDA（不支持AMD显卡，显卡是AMD请跳过）

① 查看显卡型号，支持的CUDA最高版本（非常重要） 桌面右键→NVIDIA控制面板→帮助→系统信息→组件。



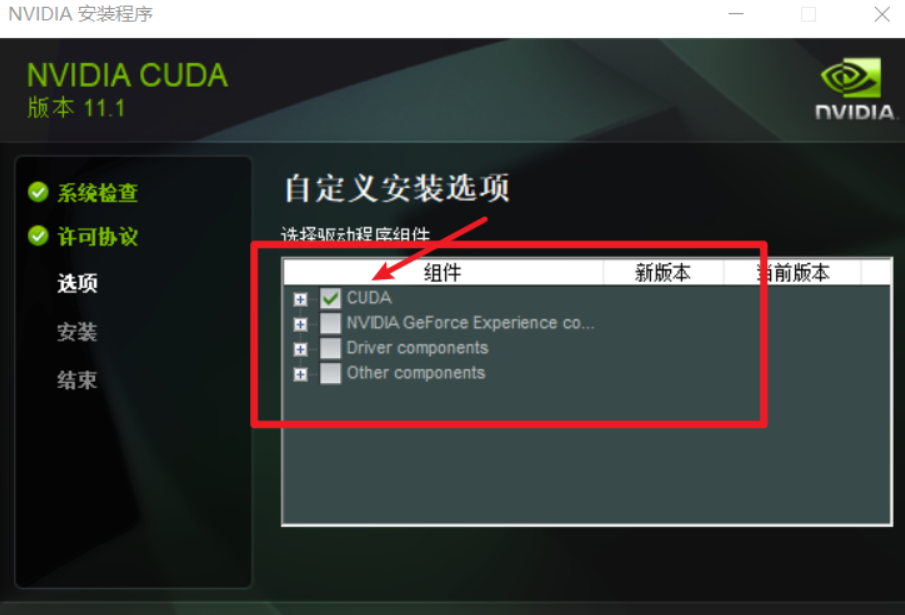
② 选择与英伟达驱动对应版本的CUDA，我的显卡驱动为466.77 > = 465.89，选择的是CUDA11.3。



③ 下载CUDA Toolkit，选择自定义，再点击下一步。

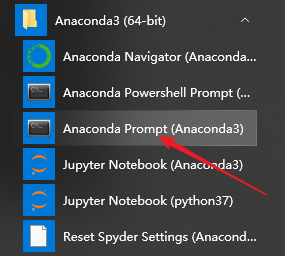


④ 选择CUDA后一直点击下一步即可。



### 4.1.3 安装Pytorch

① 打开Anaconda Prompt，在命令行格式下，输入以下四行代码，完成调用清华镜像。

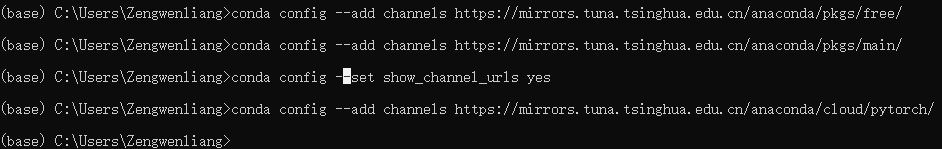


conda config --add channels <https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/free/>

conda config --add channels <https://mirrors.tuna.tsinghua.edu.cn/anaconda/pkgs/main/>

conda config --set show\_channel\_urls yes

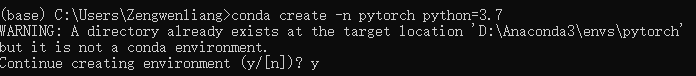
conda config --add channels <https://mirrors.tuna.tsinghua.edu.cn/anaconda/cloud/pytorch/>

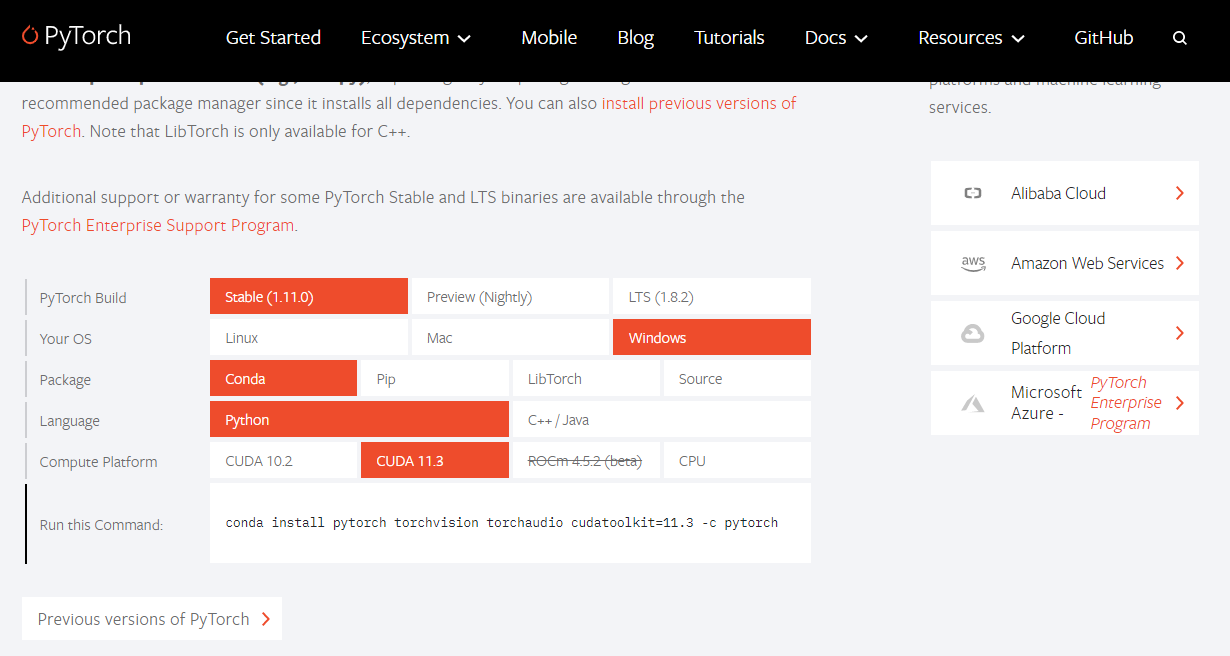


② 创建虚拟环境pytorch，使用的Python版本为3.7

conda create -n pytorch python=3.7

之后在加载过程中会弹出提示，输入y，即可安装。

③ 安装Pytorch，在Pytorch官网（https://pytorch.org/）上寻找安装命令。



如图所示版本则是复制（-c pytorch可以不要加，否则会换成国外源）：

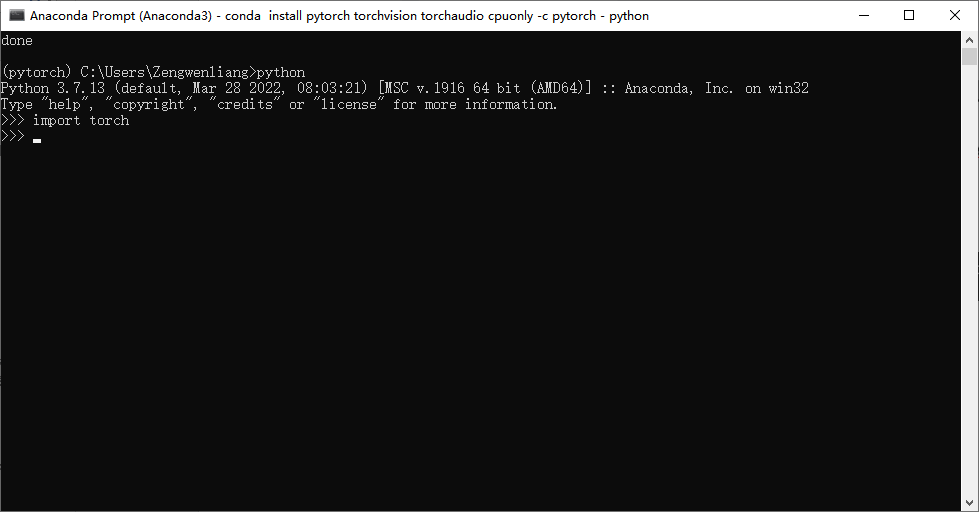
conda install pytorch torchvision torchaudio cpuonly -c pytorch

将复制的代码粘贴到命令行格式下，弹出提示，输入y，即可完成安装，显示“done”。



④ 测试pytorch

在命令行输入python进入python编译器之后，输入import torch，如下图所示则安装成功。



### 4.1.4 安装相关库

整个系统的代码需要安装相应的库，安装库的库名和版本号如下：scipy==1.2.1、numpy==1.17.0、opencv\_python==4.1.2.30、torch==1.2.0、torchvision==0.4.0、tqdm==4.60.0、Pillow==8.2.0、h5py==2.10.0、pyqt5、matplotlib、tqdm、copy、csv。打开Anaconda Prompt进入命令行，用activate激活环境后，使用“pip install 库名==版本号”的格式安装即可。

## 4.2 功能实现

### 4.2.1 导入模型

整个系统的所有功能都依赖于我们在服务器上使用YOLOX-CBAM所训练的模型，由于模型较大，为避免在每次调用功能是浪费时间再次调用，所以我们在系统的最开始导入模型。

### 4.2.2 图片检测

1) **单张图片检测**

用户点击运行gui.py后，在模型导入完成之后，系统将进入GUI界面。GUI界面主要分为了六大模块，分别为——标题栏，主页，视频检测功能区，图片检测前后对比区域，图片检测结果输出区域，图片识别功能区。如图6所示。



图 6 垃圾分类系统界面

1. 上传图片

系统使用QPushButton控件为每一个功能都设计了一个按钮，当用户点击上传图片按钮时，系统会跳转到self.change\_img方法让用户自由选择所需要检测的图片。系统使用了QFileDialog类中的getOpenFileName方法获取图片文件信息，接收到图片文件信息后，对其进行处理后得到图片文件路径。系统获得图片文件路径后，使用cv2.imread方法读取图片，再用cv2.resize方法对图像大小进行微调后保存到img文件夹中并将其显示在GUI界面上，如图7所示。



图7 图片上传效果图

1. 图片识别

当GUI界面中出现用户所选择图片后，用户点击开始识别按钮，系统将会将将上传图片时微调后保存的图片送入检测器进行检测。检测器得到图片后将图片进行调整之后送入网络中进行检测，然后会对网络的输出进行解码后得到预测框。通过对预测框进行堆叠并采用非极大值抑制之后，系统将会得到图片中的垃圾类别代码，类别得分，预测位置。随后，检测器将会根据这些值将其在图片中画出来，并将图片保存在img\_out目录下。

在图片检测结束后，GUI界面将会显示出检测后图片、垃圾种类、物品名称以及垃圾种类知识普及信息，如图8所示。



图8 识别效果图

GUI界面的中间部分显示用户选择图片的最终检测效果。系统界面右侧上方会在检测结束后显示“垃圾种类”和“物品名称”，即能够显示垃圾分类的四类垃圾（可回收垃圾、有害垃圾、厨余垃圾和其他垃圾），也能够显示具体的垃圾名称。由于训练时采用的标签为英文字母，因此系统将会根据将检测器得到的垃圾种类代码在预先设置的字典中找到相应的垃圾物品名称进行显示。系统根据垃圾种类的不同会选取不同的内容显示在界面右侧下方框中。

2) **多张图片检测**

对于多张图片检测功能，首先需要用户将所需要检测的图片放入到文件夹当中。系统要求文件夹中的图片格式为JPEG（.jpg、.jpeg）、BMP（.bmp）、PNG（.png）、DIB（.dib）、PBM（.pbm）、PGM（.pgm）、PPM（.ppm）、TIFF（.tif、.tiff），其他格式图片无法进行检测，如图9所示。

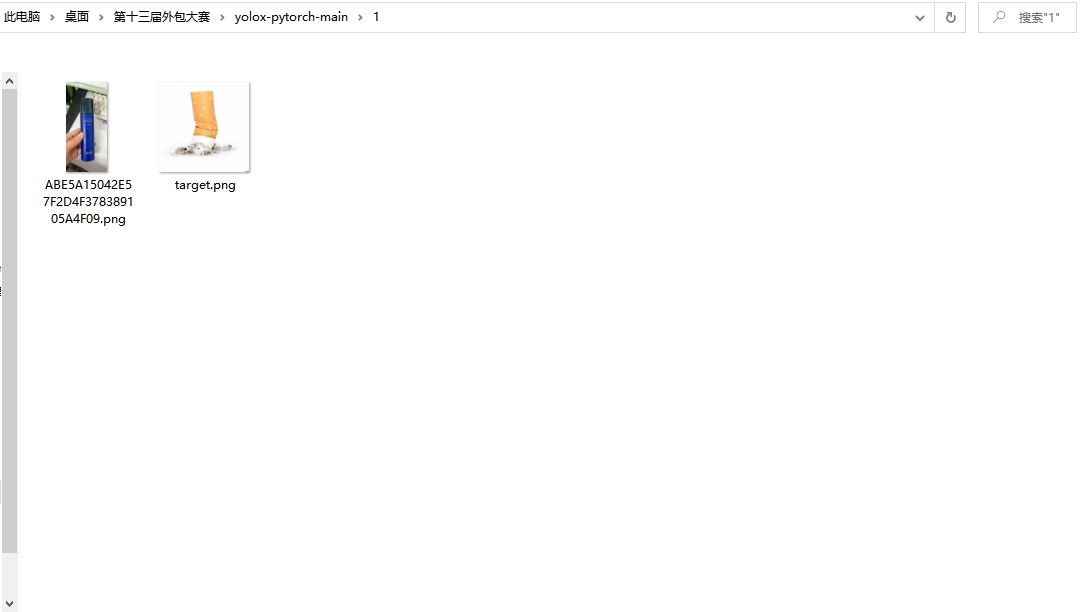


图9 待检测文件夹

将图片放入文件夹之后，用户首先点击批量识别图片按钮，然后选择放置待检测图片的文件夹。此时，系统将调用QFileDialog类中的getExistingDirectory方法去获取文件夹路径。系统得到文件夹路径后会遍历文件夹中的图片名称，并通过os.path.join方法将文件夹中的文件名和文件夹路径拼接在一起，从而得到图片的路径。在得到图片路径后，系统会将将图片路径传递给文件夹图片检测器进行检测。每当检测完一张图片后，检测器将会把检测图片的检测结果图、垃圾类别代码、是否存在垃圾（存在为1，不存在为0）、类别得分、预测框的四个坐标存入系统预先创建的列表中。当遍历完整个文件夹之后，系统会让用户选择一个文件夹作为保存位置，然后将所有结果保存到CSV文件中，如图10所示。

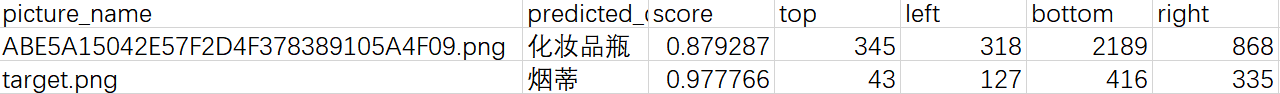


图10 CSV保存结果

同时，Pycharm运行界面会输出识别情况，如图11所示。

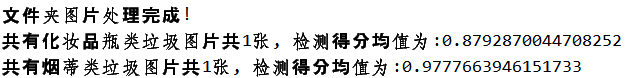


图11 识别结果分析

### 4.2.3 视频检测

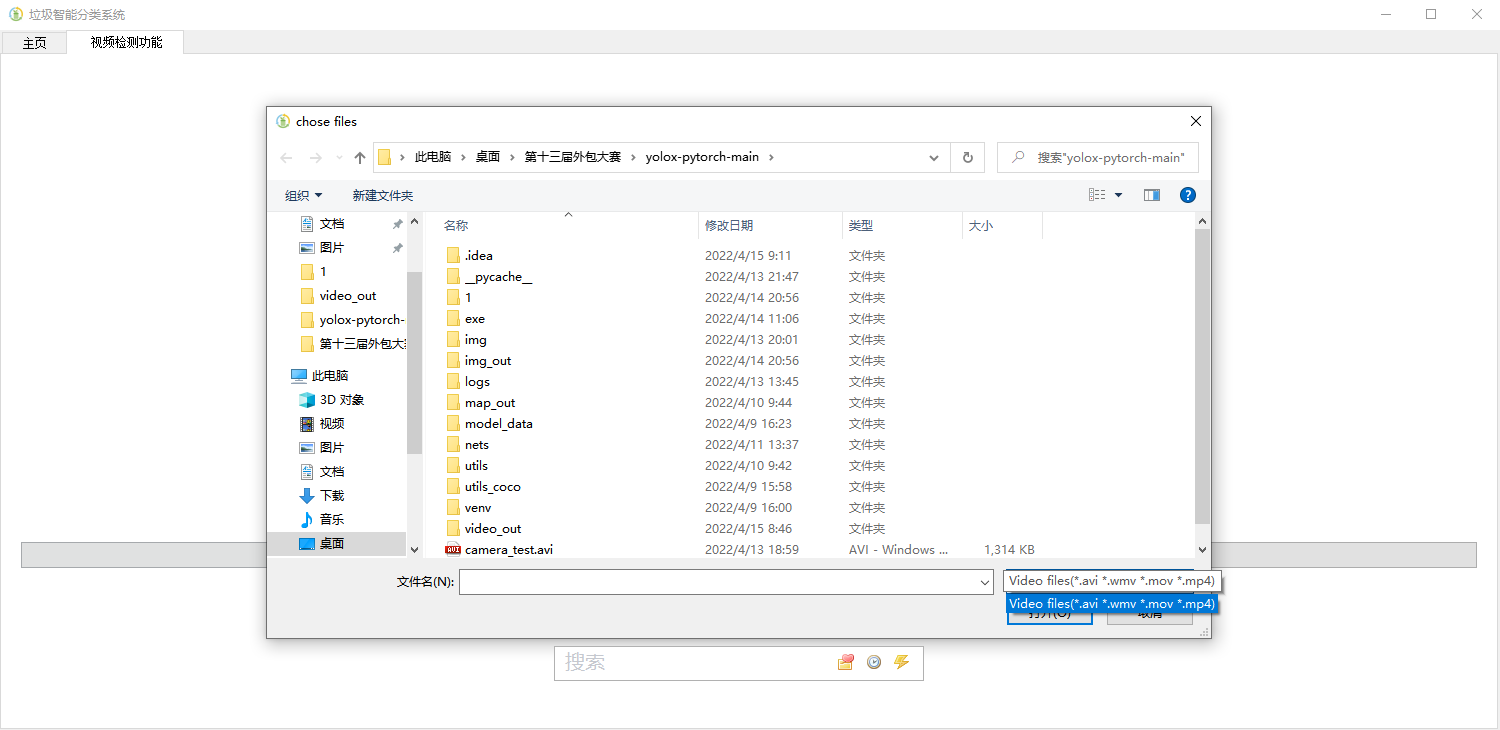
在视频检测功能模块区，为了满足用户应用在不同场景的需求，我们将其功能划分为了两个模块。模块一的功能为视频检测功能，模块二为摄像头监测功能，如图12所示。



图12 GUI界面图

1. 本地视频检测

用户在点击“选取视频”按钮之后将触发QPushButton控件的clicked.connect事件，系统将会调用视频检测方法来让用户选择需要进行检测的视频，如图13所示。与图片检测类似，视频检测部分对待视频的格式有着一定的要求，格式要求为AVI（.avi）、WMV（.wmv）、MOV（.mov）或MP4（.mp4）格式。

图13 本地视频选择

选取相对应的格式的视频之后，系统将会对视频中的画面进行逐帧处理。逐帧处理的原理是对视频中的画面通过Opencv中的方法获取到视频中的每一帧，随后将之送入视频图像检测器中，从而实现对视频的检测。

当读取到视频中的某一帧时，系统首先将其转换成为Image图像后再送入视频检测器。这里的视频检测器与之前的单张图片检测器以及多张图片检测器有所不同，视频检测器将只返回带有预测框的图片，不会返回其他值，视频检测的结果将以视频的方式进行保存。在视频检测的过程中，系统将视频进行同步播放，让用户能够看到其中的效果。同时，在视频播放界面，我们会显示出“Press Q to save and quit”和当前FPS值。“Press Q to save and quit”将提示用户能够自行决定检测视频时间范围，FPS值则与运行机器有关，但是不影响其检测精度，如图14所示。



图14 视频检测效果图

1. 摄像头监测

摄像头监测的原理与视频检测类似，摄像头监测与视频检测共用一个检测器，但是系统会调用不同的方法。在摄像头监测方法中，我们会将视频检测中的用户自行选取的视频路径设置为0，然后使用Opencv中的VideoCapture方法打开摄像头，对摄像头所捕获的画面进行逐帧处理。

在摄像头监测的过程中，界面上同样会显示出“Press Q to save and quit”和当前FPS值。“Press Q to save and quit”将提示用户能够自行决定监测时长，FPS值则与运行机器有关，但是不影响其检测精度，如图15所示。

图15 摄像头监测效果图

# 第5章 应用场景落地

经过三个多月的设计和开发，基于YOLOX的赋能垃圾分类检测系统在团队五人的努力下终于圆满完成，无论是界面还是功能都基本符合预期的需求与规划。

系统拟投入到社区、企业中使用，实现智能垃圾分类。随着人工智能的发展，智能机器人在我们生活中将系统程序安装在社区、企业的智能移动机器人中，可连接5G网络。

首先，用户可以根据设备上的垃圾分类提示自行将待投入的垃圾进行分类，其次，当用户不能准确分辨垃圾类别时，可以将垃圾放在设备的检测器置物台上，设备通过摄像头对垃圾类别进行实时检测，确定垃圾种类后会将结果显示在机器人的显示屏上，并对垃圾分类知识进行普及。

项目在社区、企业中使用在源头上减轻了垃圾分类的工作量，丢完垃圾后的垃圾回收、处理工作更容易展开。且能够更好地实现系统的部署与维护。