项目编号：T030PRP38103



**本科生研究计划（PRP）研究论文**

(第 38 期)

**论文题目：\_\_基于B/S架构的实时图像标记系统\_\_\_**

**项目负责人：盛斌 学院（系）： 电子信息与电气工程学院**

**指导教师：盛斌 学院（系）： 电子信息与电气工程学院**

**参与学生：苏勇文、沈君陶、陈展、康家铭、孙鑫**

**项目执行时间：2020年 4 月 至 2020 年 10月**

**基于B/S架构的实时图像标记系统**

电子信息与电气工程学院

F1803301

苏勇文

盛斌，电子信息与电气工程学院

摘要

垃圾分类因其环保与经济效益，正日益受到国际社会的关注，近年来中国也实施了一系列的相关政策。然而在现有的深度学习方法中，需要使用大量带有标记的垃圾图片进行端到端的训练，数据集的采集成为难题。本论文提出了一种可以用于图片标注的实时图像标记系统，从而方便地对采集到的图片进行实时便捷的标记。本系统轻量级部署简单，使用了服务器-浏览器架构，通过HTTP协议传输标记数据并存储在服务器，适合多场景的图像标注。

关键字: 图片标注、B/S架构、垃圾分类

ABSTRACT

Because of its environmental protection and economic benefits, garbage sorting is receiving increasing attention from the international community. In recent years, China has also implemented a series of related policies. However, in the existing deep learning methods, a large number of marked garbage images need to be used for end-to-end training, and the collection of data sets becomes a problem. This paper proposes a real-time image labeling system that can be used for image labeling, so as to conveniently label the collected images in real time. This system is lightweight and simple to deploy, uses a server-browser architecture, transmits the marked data through the HTTP protocol and stores it on the server, which is suitable for image annotation in multiple scenarios.

**KEY WORDS:** Image annotation, B/S architecture, Trash classification

1. 绪论

1.1 国内外概况

近年来，垃圾分类逐渐受到国际社会的重视，垃圾的分类投放不但有利于整理市容，而且是城市环境治理不可缺少的一环。部分发达国家已经非常重视垃圾分类并率先建立好垃圾分类体系，并且用先进技术进行自动化分类。在垃圾分类这一问题上，中国目前还较为落后，首先政策上倡导的垃圾分类并没有深入到基层，再者民众的垃圾分类意识还未普及，同时用于垃圾分类的技术还不够成熟。随着近几年深度学习的发展，使用神经网络预测的垃圾分类技术进入社会视野，然而伴随而来的主要问题是数据集的缺失，现如今较为丰富精美的城市垃圾标记图像数据集还非常稀少，因此寻找能够快速制作数据集的方法至关重要。

1.2 选题意义和目的

目前，实施废物分类有两个重要障碍。首先，训练垃圾分类的神经网络模型需要大量图像标记数据集，以上海的废物分类政策为例。从互联网上的许多讨论中可以看出，许多市民无法正确地区分干垃圾和湿垃圾。这导致这样一个事实，即使公民愿意分类和标记垃圾，也可能由于缺乏相关知识而进行错误标记，因此我们需要雇佣专人进行标记。第二是现有的图像标记软件成本较为昂贵且使用复杂。一些公民不了解如何使用复杂的标记软件，一些公司和组织难以负担昂贵的软件费用。这样的政策显然需要大量的人力和物力，并且难以实施。为了节省人力和物力，我们正在尝试开发一个在线图像标记系统，该系统可以从下载获取需要标记的图片，并可以让使用者通过简单的操作进行标记垃圾的类型和位置，以帮助公众对垃圾进行分类。另外，在完成标记软件的开发之后，我们还可以考虑将此软件与开发的深度学习目标检测软件相结合，实现自动训练集的制作，这将进一步减少软件成本，造福于社会。

1.3 研究范围

本项目涵盖了这样一个实时图像标记软件的全部流程。包括前端标记界面的整体设计，和后端服务器的研发（HTTP服务器、数据存储）。本论文主要阐述其中软件数据协议设计的相关内容。具体而言，我们将HTTP协议与图像标记相结合，用于图像标记任务。

1.4 解决的主要问题

本项目是专注于解决标记软件的成本上的实际应用，解决当前国内高质量数据集缺失的缺口，同时也能大大节省垃圾分类实施上的成本，为推进我国垃圾分类政策的实施助力，丰富我国大城市的市容市貌。

1. 研究内容

B/S架构即浏览器和服务器架构模式，是随着Internet技术的兴起，对C/S架构的一种变化或者改进的架构。在这种架构下，用户工作界面是通过WWW浏览器来实现，极少部分事务逻辑在前端（Browser）实现，但是主要事务逻辑在服务器端(Server)实现。由于图像标记软件对便携性和跨平台性有着较高的要求，如果使用传统的C/S架构进行开发，便会影响用户的使用体验。因此本项目使用B/S架构进行开发，使得用户能够通过简单的网络登入便能使用高效便捷的图像标记功能。

本项目基于html5, JavaScript, Css实现，基于http协议实现前后端数据通信和权限控制，基于 java实现画板网页的传输，最终实现了实时图像标记软件。最终软件由标记界面、后端服务器、标记数据组成。

图示

描述已自动生成

图1：标记软件

①

标记界面：用来执行移动、标记、保存、更新等各种操作

数据：用来保存标记图片和标记数据

后端服务器：负责控制标记数据同步和任务管理

为了进行高效的开发，本组将开发任务划分为标记界面实现、数据通信实现、后端数据管理三部分。

2.1标记界面

随着浏览器版本的更新换代，功能丰富，应用广泛的Html5技术已经被大多数主流浏览器所支持。Html5技术是html, css, javasript多项技术的组合，这些技术使得其拥有快速处理不同多媒体信息的能力。由于其功能强大，设计简便，特别是其支持多设备，跨平台的性能，本小组选用html5技术来设计标记界面，以满足不同型号设备的需要。

canvas是html5中新增的一个用于定义图形的标签，作为Html5中已经定义好的图形容器，它本身没有行为，但提供了绘图API给予客户端JavaScript，只需对其增加相关脚本即可满足网页画板的对应功能。通过canvas提供的API编写脚本，可以实现在网页实时生成图像。

在标记界面的html代码中引用了MarkImage.js来调用函数以及实现各种功能，MarkImage.js中使用了canvas的API来实现种种功能，本项目采用了较为成熟的开源项目代码（<https://github.com/heylight/markImage>）。

2.2前后端通信

为了实现标记界面和后端服务器之间的数据传输，本项目决定使用http协议作为数据传输的协议。本项目是依赖html文件接入后端而实现的在线图像标记软件，为了实现客户端和后端服务器之间的数据传输，决定搭建一个http服务器负责传输用户界面的html文件同时接受标记数据，http服务器主要与客户端使用socket通信，解析请求头然后发送答复报文。该http服务器启动后开始监听8080端口，当同一个局域网的浏览器访问本主机的8080端口时将会启动连接，然后返回所需要的文件。

本项目主要使用了http协议中的GET和POST请求，GET请求负责从服务器索取原图像，POST请求的目的则是传输标记数据给服务器，从而存储到后台数据文件夹。

2.3数据存储

在标记界面向服务器发出POST请求后，服务器接受请求并解析来获得标记数据，接着需要以文本的形式存储到服务器本地，以供接下来使用。POST请求为标记数据矩阵所组成的数据，所以需要服务器进行相应的解析处理。当标记界面向服务器发出GET请求时，服务器接受请求后需要发送一张尚未标记的图片，因此需要根据标记数据来排除已被标记的图片，用到了一定的查找算法。

1. 研究方法与结果

3.1数据通信的设计与实现

本项目需要在标记界面和后端服务器之间进行图片、标记数据、指令的共享，因此设计通信流程是至关重要的，以下将通信流程拆分为连接服务器、获取图片、标记数据上传等步骤分别进行阐述。

3.1.1 连接服务器

由于本项目使用Socket进行数据的传输，因此首先需要启动后端的服务器，在网页登录之后需要与后端进行连接，之后才能进行接下来的各种操作

图示

描述已自动生成

图2：连接服务器示意图

由于标记界面需要与服务器进行数据传输，因此需要首先进行标记界面的连接。图中步骤①为标记界面进行HTTP连接。标记界面连接之后服务器便可以进行步骤②，传输一张当前的未标记图片给标记界面。

3.1.2 申请未标记图片

标记界面获得初始图片后可以进行标记并传回服务器，也可以申请新的未标记图片，每当一个对图片的请求传来，后端便会发送一个尚未标记的图片，将数据实时传输到标记界面。**图示

描述已自动生成** **图示

描述已自动生成**

图3：申请图片示意图 图4：标记数据上传示意图

图中步骤①为后端服务器取到初始图片，步骤②为后端将初始图片发送给标记界面。这时需要标记者进行标记并保存之后点击获取新的图片，之后通过步骤③将新图片的请求反馈给后端。最后经过步骤④后端将所要取出的未标记图片反馈回后端数据，再由后端服务器取出下一个需要传送给标记界面的图片。

3.1.3 标记数据上传

标记数据的上传主要在标记界面和后端服务器之间进行传输，步骤①为后端服务器向标记界面传输图片。之后在标记界面进行标记处理，将数据经由步骤②对数据进行编码，并上传给后端服务器。

3.1.4 数据协议设计

由于HTTP传输的是字节流，因此本项目自行设计数据协议，主要包括标记界面和服务器、服务器和后端数据之间的数据协议设计。

3.1.4.1 标记界面发往服务器

|  |  |  |  |
| --- | --- | --- | --- |
|  | **指令** | **data** | **when to send** |
| **标记界面** | 请求新图片 | img | 点击“更新”时 |
| 上传标记数据 | list | 点击“保存”时 |

表1：标记界面发送数据设计

3.1.4.2 服务器发往标记界面

|  |  |  |  |
| --- | --- | --- | --- |
|  | **指令** | **data** | **when to send** |
| **服务器** | 发送新图片 | img | 连接以及收到请求时 |
| 发送js插件 | js | 连接时 |

表2：服务器发送数据设计

3.2标记界面设计

3.2.1标记界面功能介绍

本组利用html5编写网页画板，除了满足基本的对图片的放大缩小平移的功能外，还增加了标记功能，更新图片和保存标记数据功能，这些功能都能通过在canvas和MarkImage.js提供的API中编写来实现。标记功能允许用户通过鼠标选取一定区域作为标记结果，同时将产生相应的标记数据list；更新图片功能允许用户点击之后获取一个新的图片；保存标记数据功能允许用户点击之后上传标记数据到服务器进行存储。

3.2.2界面主体设计

canvas标签提供了可供js编辑的API，从而进行2D图像的绘制。在JavaScript中通过调用canvas提供的getElementById函数即可使用canvas功能，然后通过getContent函数获得ctx对象，用于标注图像与产生标注数据等。

当用户使用鼠标点击或者触摸屏幕时将触发Html5的一个“事件”(event)，在html5中这些事件被分为了不同类型，以鼠标触碰事件为例，其有如下类型：

|  |  |
| --- | --- |
| 事件类型 | 说明 |
| 点击 | 执行放大、缩小、更新、保存、删除等操作 |
| 文本框 | 用于输入选区索引 |
| 拖移 | 在图像上确定选区 |

**表3：事件属性**

这些事件被触发后可以将其与JavaScript函数相关联，通过对这些事件的定义，即可完成对标记功能的实现。

3.2.3后端交互功能

Java提供较为完善的Socket传输API，本项目调用实现了标记界面和服务器的Socket通讯，使得前端点击功能之后可以使用其函数与后端进行交互，其中newimage()可以接收到服务器发来的数据，而sendmessage()则可以将数据发送到后端。因此通过这两个函数即可实现对后端数据的交互。

当现在标记数据更新时将点位信息发向后端，并接受后端信息并将其展现在界面上，即实现了前后端数据的交互。

3.2.3.1 传输数据的设计

通过自定义的sendmessage()函数向后端传输数据，当用户在标记界面点击保存按钮时传输。通过MarkImage.js获取到的标记数据是一串数字，因此需要进行一些编码，使得服务器方便对此进行解读。因此设计传输数据结构为:data.message; 其中data为一串数字，以逗号分隔开，.message表示数据为标记数据，是为了让服务器分辨信息类型。

3.2.3.2数据传输类型

根据传输数据的type值，将其进行编码，由下表所示，不同类型的数据记录不同的数据内容并且在接收这些数据时根据type类型判断处理。

|  |  |  |  |
| --- | --- | --- | --- |
| Data segment | Type | 属性 | |
| 1 | Array[4] | 标记框一的数据 | |
| 2 | Array[4] | 标记框二的数据 | |
| 3 | Array[4] | 标记框三的数据 | |
| 4 | Array[4] | 标记框四的数据 | |
| 5 | .message | 类型后缀，用来表示为标记数据 |

表4：传输数据类型

3.2.3.3接受数据并存储方式

由上述对通讯数据的介绍可知，可获得前端标记界面传输而来的数据，在判断数据类型后即可对其进行处理。如图所示，获得后端发送的数据后，首先对数据本身的类型进行判断，如果为画板数据则选择画板数据类再进行处理。

图示

描述已自动生成图表, 瀑布图

描述已自动生成

图5：接收流程 图6：http服务器流程

3.3 组件连接设计

3.3.1 http服务器

超文本传输​​协议（HTTP）是一个用于传输超媒体文档（例如 HTML）的应用层协议。它是为 Web 浏览器与 Web 服务器之间的通信而设计的，但也可以用于其他目的。HTTP 遵循经典的客户端-服务端模型，客户端打开一个连接以发出请求，然后等待直到收到服务器端响应。在本项目中，浏览器作为客户端发出http请求，需要我们自行搭建一个http服务器响应客户端的请求，传输画板所需要的html文件。

3.3.1.1 http服务器搭建环境

在java环境中搭建http服务器，使用eclipse IDE，创建项目后导入需要的依赖包，本项目采用较为基础的socket API，将从头实现一个服务器。

套接字使用TCP提供了两台计算机之间的通信机制。 客户端程序创建一个套接字，并尝试连接服务器的套接字。

当连接建立时，服务器会创建一个 Socket 对象。客户端和服务器现在可以通过对 Socket 对象的写入和读取来进行通信。

java.net.Socket 类代表一个套接字，并且 java.net.ServerSocket 类为服务器程序提供了一种来监听客户端，并与他们建立连接的机制。

以下步骤在两台计算机之间使用套接字建立TCP连接时会出现：

服务器实例化一个 ServerSocket 对象，表示通过服务器上的端口通信。

服务器调用 ServerSocket 类的 accept() 方法，该方法将一直等待，直到客户端连接到服务器上给定的端口。

服务器正在等待时，一个客户端实例化一个 Socket 对象，指定服务器名称和端口号来请求连接。

Socket 类的构造函数试图将客户端连接到指定的服务器和端口号。如果通信被建立，则在客户端创建一个 Socket 对象能够与服务器进行通信。

在服务器端，accept() 方法返回服务器上一个新的 socket 引用，该 socket 连接到客户端的 socket。

3.3.1.2 实现Runnable接口

使用Runnable接口

**public class App implements Runnable**

在**public void run()**这个函数中实现整个服务器的主线程

1. 首先创建客户端socket并监听客户端的请求调用accept()会自动开始监听服务器端口，如果收到请求将会自动连接。
2. 客户端和服务器建立连接之后，需要读取客户端发出的http请求内容来确定传输的文件
3. http服务器根据客户端请求的方式和需要的文件传输给浏览器。

在本项目的设计中只需要实现对GET请求的回应即可。如果浏览器用GET请求空则返回用户界面，如果请求不为空则截取请求内容，然后将所请求的内容发送回浏览器。设计了处理html文件和js文件的逻辑。

手机屏幕截图



描述已自动生成 图示



描述已自动生成

图7：http请求格式 图8：http应答报文

其中调用了传输文件的函数 fileService()，这个函数负责将参数名对应的文件根据ConTentType发送到client，需要注意的是http应答报文的格式。照正确的应答格式发送http应答报文，浏览器就可以正确处理，进而解析html文件。  
 3.3.2 标记数据保存

本项目需要将前端标记界面传来的数据保存起来供接下来使用，保存时需要对传来数据进行解析才可以。标记界面使用GET将数据上传给服务器，因此需要进行相应的解码。首先判断http请求类型，如果是GET则接受数据流，然后判断后缀是否是.message，是则进行下一步数据的解析，最后将数据以txt文本的形式存储到后台服务器。

图示

描述已自动生成

图9：接收数据并保存

3.4结果展示

3.4.1后端服务器运行

由于后端服务器使用Java自带API进行实现，因此需要提前安装好Java SDK并配置好环境变量，这里直接展示运行步骤。

进入Eclipse IDE中，启动服务器。

图形用户界面, 文本, 应用程序

描述已自动生成

图10：后端服务器启动

3.4.2标记界面

连接只需要使用浏览器登录服务器端口即可，本项目中为localhost:8080，运行效果如下（图11）：

碗里放着各种食物

描述已自动生成文本

描述已自动生成

图11：标记界面启动 图12：标记数据存储

3.4.3数据存储

点击保存即可存储数据，到主文件夹即可找到。（见图12）

1. 讨论

本项目是图像标记在垃圾分类领域一次基础的尝试，实现了一个实时图像标记软件，减少了标记的成本并使得其更便捷，为实现垃圾分类贡献一份力量。不可否认的是，本项目也存在一些缺陷。在软件架构上对于高并发不够理想。项目组尝试了使用Java多线程编程，但是实际实施中，因为架构问题等原因，多并发的表现并不理想。此外，本标记软件对于之前提到的使用不规范行为（比如不正确的操作顺序）鲁棒性不佳，这类行为的避免需要一些辅助的监督手段（比如指定相应的规章制度）。

1. 结论

本论文研究图像标记在垃圾分类这一时下热点问题上的解决办法，通过结合html与http协议我们实现了具有一定实用性的图像标记软件。不可否认的是，本项目仍存在诸如大规模并行表现不佳，对使用中的不规范行为鲁棒性差等问题。将来可以针对这些问题，改进软件架构，进行多线程高并发编程以获得更好的体验效果，让该实时图像标记软件能够得到实际部署。此外，将来还可以考虑在现有图像标记软件的基础上，结合神经网络进行自动化检测，开发全自动化的垃圾分拣系统，有利于构建和谐的市容市貌。

1. 参考文献

[1]薛坤,赵国玲.基于Web的前端开发技术探究[J].数码世界,2021(02):5-6.

[2]王红军,胡宇航,汪洪祥,刘玮,高巍,荣雯皓.Web前端开发课程体系建设与实践探索[J].文化创新比较研究,2021,5(02):64-66.

[3]刘淑丽.基于Java的图书馆人脸识别系统设计与实现[J].科技创新与应用,2021(07):86-88.

[4]MarkImage.js (<https://github.com/heylight/markImage>)

谢辞

经过几个月的研究,我最终完成了项目相关的工作。本论文的顺利完成，离不开各位老师、学长和同学的关心和帮助。本项目是由盛斌老师精心挑选提供的，由李佳佳学长全程指导的。盛老师和李佳佳学长为了项目有条不紊地实施，详细安排了项目的时间表并在过程中给予帮助和督促，也在我遇到困难时主动指导，帮助我开拓研究思路，充分地指导我。在此感谢盛老师和李佳佳学长对我的指导。此外，还要感谢项目小组的其他几位同学：沈君陶、陈展、康家铭、孙鑫。项目过程中我们互相交流想法和思路、互帮互助。最后，衷心感谢在百忙之中抽出时间审阅本论文的导师！