## 半自动标注系统详细设计方案

邹安平

### 目标

本文描述将半自动标注系统详细设计方案，为后续的系统实现提供工作指导。

###### 1.1 背景

半自动化标注系统是用来帮助使用者完成标注任务，并获得最终人工智能模型。。

###### 1.2 用户需求

###### 1.2.1 交通组需求

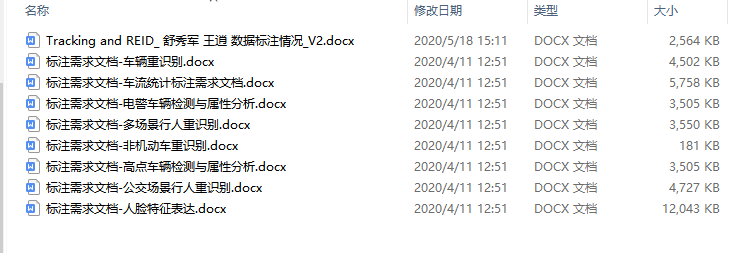


图1-1 交通组需求文档截图

交通课题组对标注系统提出上述需求，具体的需求文档比较多，不在本文中提供了。总结如下：

标注系统需要支持人、车、人脸的自动检测。

标注系统需要支持人、车的标注，包括车辆类型、车辆颜色等。

标注系统需要支持人脸标注。

标注系统需要支持人、车的重识别标注。

标注系统需要支持Tracking（视频追踪）。

标注系统需要支持标注数据导出。

标注系统需要支持视频标注。

标注系统需要支持车流统计标注。

这些需求在标注系统实现之后，还需要人工来进行标注，从而引申出如下非功能性能需求：

标注系统需要支持用户管理

标注系统需要支持标注任务管理。

标注系统需要支持标注人员工作度量，即报表管理。

###### 1.2.2 医疗组需求

标注系统不光要能标注交通类数据，还要标注医疗类数据，但医疗类数据与交通类数据不一样，交通类数据都是正常的视频、图片，而医疗类都是CT扫描的dicom格式数据及病理类超大图片svs等格式数据。

CT扫描类数据标注需求：

|  |  |
| --- | --- |
| **需求类型** | **需求描述** |
| CT类 | 1、支持点标注/bbox标注/多边形标注  2、语义分割、实例分割标注支持  3、曲线拟合  4、交互式标注补全（点击目标内部，生成外围轮廓；多边形点的位置优化）  5、支持标注格式：普通图片、CT序列、MRI序列、病理切片  6、数据集标注情况统计  7、其它标注需求（参考常用工具ITK-SNAP）  - 支持三视图展示(右边2个视图由横断面视图合成)  - 三维标注  - 对比度调整（CT标注中必备功能）  - 图像缩放  - 标注结果3D合成(左下) |
| 病理类 | 1、支持在线浏览，在线浏览部分参考openslide/openseaDragon。(<https://openslide.org/demo/>，<https://microdraw.pasteur.fr/data?source=/bigbrain/bigbrain.json>)  2、超大病理切片图像在线浏览支持，>5G，10w×10w像素  3、点标注，多边形标注，固定大小ROI区域选定  4、导航预览(左上角)  5、颜色/对比度归一化  6、不同任务标注导入 |

###### 1.3 可扩展性

标注系统在设计时需要考虑可扩展性：一个是功能上可扩展，另外一个部署上可扩展。

### 功能列表

根据用户需求列功能列表：

###### 2.1用户管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 增加用户 | 增加一个用户，让用户可以使用标注 系统 | 用户名  密码  电话  电子邮箱  公司  地址  呢称 | 用户ID  用户信息保存在数据库中 |
| 删除用户 | 删除用户 | 用户ID | 数据库中删除该用户记录 |
| 修改用户 | 修改用户信息 | 密码  电话  电子邮箱  公司  地址  呢称 | 修改是否成功 |

###### 2.2用户登录

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 用户登录 | 对用户进行鉴权 | 用户名  密码 | 登录成功后，  返回用户Token |
| 用户注销 | 当前登录用户退出登录 | 用户名 | Token失效 |

###### 2.3数据集管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 新建数据集 | 新建一个图片或者CT影像格式的数据集。  支持带标注数据导入。 | 1. 数据集名称 2. 数据集类型 3. 上传图片或者CT影像（压缩包） 4. 其它 | 1. 创建数据集保存在数据库中。 2. 创建完成后，可以进行数据集预览。 |
| 新建视频数据集 | 1. 新建视频数据集，可以选择多个视频上传。 2. 创建完成后，可以进行抽帧、合并 | 1. 数据集名称 2. 数据集类型 3. 上传图片或者CT影像（压缩包）   其它 | 1、创建数据集保存在数据库中。  2、创建完成后，可以进行数据集预览。 |
| 导出数据集 | 对已经创建的数据集可以再进行导出。 | 1、选择数据集 | 1、压缩文件 |
| 删除数据集 | 对已经创建的数据集进行删除 | 1. 选择数据集 | 无 |
| 修改数据集 | 对已经创建的数据集进行修改 | 1. 选择数据集 2. 修改指派用户 3. 修改数据集描述 |  |
| 查询数据集 | 对已经创建的数据集进行分布查询 | 无 | 分页显示数据集列表 |

###### 2.4自动标注

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 新建自动标注任务 | 新建一个自动标注任务，选择一个数据集，选择指定的模型，自动进行一次标注 | 1、选择数据集  2、选择算法模型  3、输入标注名称 | 1、创建任务保存在数据库中， |
| 查询自动标注任务 | 默认查询当前用户已经创建过的自动标注任务，分页显示，每页10条，按照创建时间倒序排列。 | 1. 当前用户名 2. 起始页 3. 每页数量 | 1、自动标注任务名称  2、用户  3、状态  4、任务开始时间 |
| 删除自动标注任务 | 删除标注任务：   1. 删除数据库中记录 | 1、自动标注任务名称 | 无 |

###### 2.5人工标注任务管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 新建自动标注结果人工标注任务 | 对自动标注的结果进行人工调整 | 1、标注任务名称  2、选择自动标注任务 | 1、创建任务保存在数据库中 |
| 新建数据集人工标注 | 对数据集中原始数据进行人工标注 | 1、标注任务名称  2、选择数据集 | 1、创建任务保存在数据库中 |
| 查询数据集人工标注 | 默认查询当前用户已经创建过的标注任务，分页显示，每页20条，按照创建时间倒序排列。 | 1、当前用户名  2、起始页  3、每页数量 | 1、标注任务名称  2、用户  3、任务状态（进度）  4、更新时间 |
| 删除人工标注任务 | 删除标注任务：  1、删除数据库中记录   1. 删除图片 | 1、标注任务名称 | 无 |
| 人工标注工具 | 1. 可以使用矩形框、多边形、点对图片进行标注。 2. 可以自定义设置属性。 3. 可以放大、缩小，拖动。 | 无 | 无 |
| 导出标注数据 | 导出Json及xml格式标注数据。 | 标注任务 | Zip格式压缩文件 |
| 转审核 | 转给审核人员对每个框进行审核，可以设置合格或者不合格。 | 标注任务 | 无 |

###### 2.6车流统计任务管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 新建车流统计任务 | 对自动标注的结果进行人工调整 | 1、标注任务名称  2、视频数据集 | 1、创建任务保存在数据库中 |
| 删除车流统计任务 | 删除车流统计任务 | 1、标注任务名称 |  |
| 人工标注工具 | 1. 可以播放视频 2. 可以下拉选择车辆类型 3. 可以选择车辆行驶方向 | 无 | 无 |
| 导出标注数据 | 导出Json及xml格式标注数据。 | 标注任务 | Zip格式压缩文件 |
|  |  |  |  |

###### 2.7ReID标注任务管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 新建自动标注结果ReID标注任务 | 对自动标注的结果进行ReID标注 | 1、标注任务名称  2、选择源自动标注任务  3、选择对照自动标注任务 | 1、创建任务保存在数据库中 |
| 新建数据集ReID标注 | 对数据集中原始数据进行人工标注 | 1、标注任务名称  2、选择源数据集  3、选择对照数据集 | 1、创建任务保存在数据库中 |
| 删除ReID标注任务 | 删除标注任务：  1、删除数据库中记录 | 1、标注任务名称 | 无 |
| ReID标注工具 | 1、可以使用矩形框、多边形、点对图片进行标注。  2、自动出现ReID属性。还可自定义设置属性。  3、可以放大、缩小，拖动。  4、可直观查看ReID标注结果  5、可以复制、删除指定标注，  6、可对目标进行目标跟踪。  7、如果抽帧图片带有时间格式，还可以根据时间列出最近的ReID图片。 | 无 | 无 |
| 导出标注数据 | 导出Json及xml格式标注数据。 | 标注任务 | Zip格式压缩文件 |

###### 2.8视频标注任务管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 新建视频标注任务 | 对视频进行标注 | 1、标注任务名称  2、选择视频数据集 | 1、创建任务保存在数据库中 |
| 查询视频标注 | 默认查询当前用户已经创建过的标注任务，分页显示，每页20条，按照创建时间倒序排列。 | 1、当前用户名  2、起始页  3、每页数量 | 1、标注任务名称  2、用户  3、任务状态  4、更新时间 |
| 删除视频标注任务 | 删除标注任务：  1、删除数据库中记录 | 1、标注任务名称 | 无 |
| 视频标注工具 | 1、可以使用矩形框、多边形、点对视频进行标注。  2、可以自定义设置属性。  3、可以放大、缩小，拖动帧。  4、可以自动对某些帧进行标注 | 无 | 无 |
| 导出标注数据 | 导出Json及xml格式标注数据。 | 标注任务 | Zip格式压缩文件 |

###### 2.9模型管理及重训

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 模型增加 | 对训练模型进行增加，可以基于mmdetection增加其支持的模型，或者根据标准接口支持任意的模型。 | 模型名称  模型位置  模型输入参数 | 无 |
| 模型删除 | 删除模型 | 模型名称 | 无 |
| 模型重训 | 对模型进行重训 | 模型名称 | 新的模型 |

###### 2.10报表管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 报表数据 | 对每个操作的用户产生标注图片、标注框、标注属性、标注不合格的数据 | 无 | 无 |
| 报表度量 | 选择不同的用户可以对标注框等属性进行横向对比。 | 用户名称  时间 | 无 |

###### 2.11超大图像标注

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 新建超大图像标注 | 新建超大图像标注任务 | 选择数据集 | 无 |
| 超大图像标注 | 在标注页面进行标注 | 选择标注任务 | 无 |
| 导出标注数据 | 导出标注数据 | 选择标注任务 | 标注数据文件 |

###### 2.12操作日志管理

|  |  |  |  |
| --- | --- | --- | --- |
| 功能名称 | 功能说明 | 输入参数 | 输出参数 |
| 操作日志管理 | 记录用户的安全相关操作 | 无 | 无 |

### 系统功能详细设计

### 3.1系统架构

根据用户相关需求及列出的功能列表，综合考虑，半自动标注最终系统架构应该如下图所示。

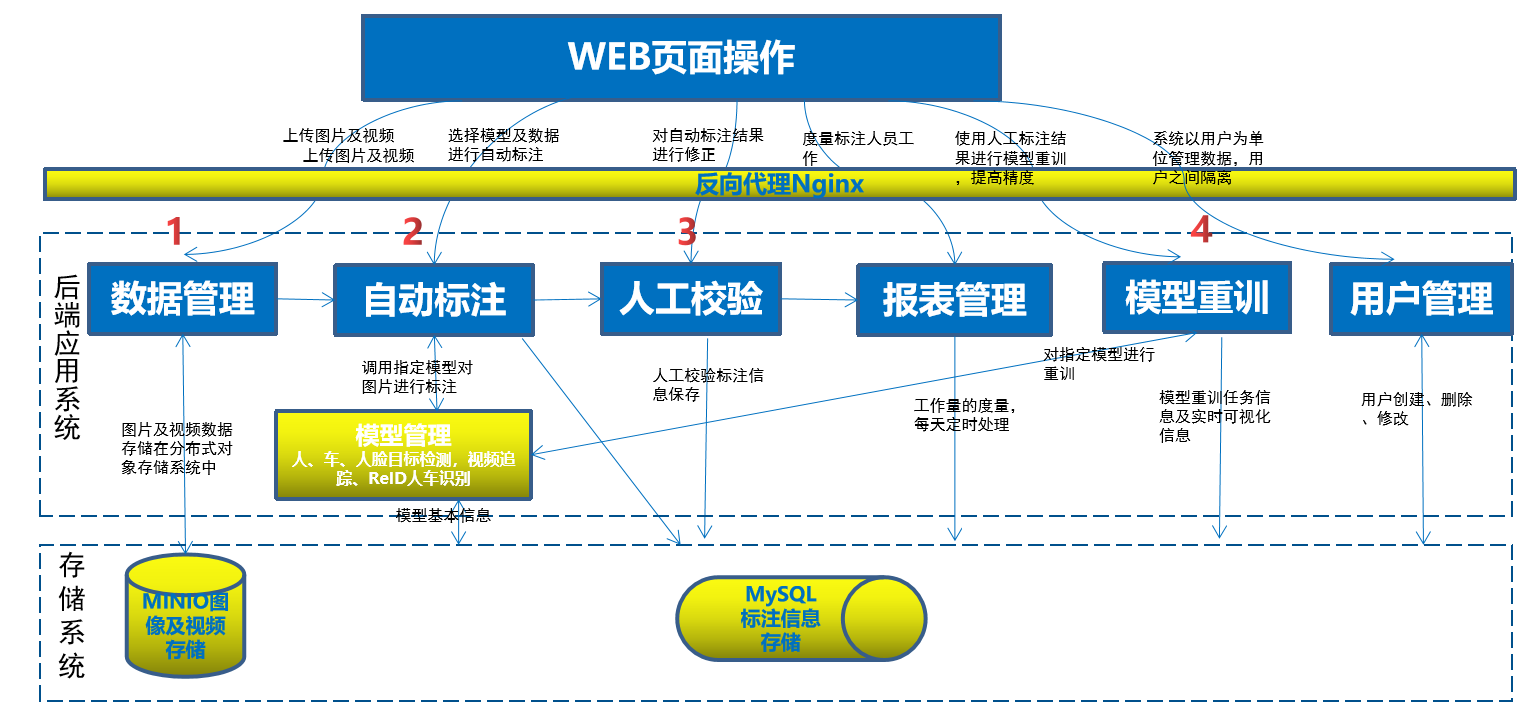


图1-2 标注系统总体架构

本系统采用BS架构，方便对数据在服务端进行集中安全管控，因此用户在WEB页面进行操作。另外由于标注系统更多是操作图片，并进行渲染，因此不需要使用很多的框架，采用原生的JS及Jquery进行开发。

后端应用系统采用目前最流行的Spring Boot进行开发，方便快捷，响应速度快。

在存储系统的设计中，标注任务相关信息及数据最终采用MySQL进行存储，用户上传的文件、视频、图片，由于系统需要支持海量的小文件及很多大文件，最终采用minio文件存储系统进行存储。MySQL及minio都是开源免费的，并且使用简单，开发者多及活跃，这就是为什么最终采用这两者的原因。

另外存储系统也可以还支持华为云上的OBS存储替换minio，可以参见：OBSFileService的实现。

### 3.2功能详细设计

###### 3.2.1 用户登录、管理功能

标注系统支持多用户操作，那么用户管理必然要支持增加用户，删除用户，并且区分用户类型，比如管理员，标注人员，审核人员等，同时每个用户的操作应该是独立的，所以其它功能的操作过程中，需要将用户信息全程携带，由于是在Web端保存用户信息，因此遵照惯例，Web端在用户登录之后，只保存用户的Token信息，客户端与服务端通过Token来识别对应的用户。

用户管理设计架构及流程如下图所示：



图1-3 用户登录及管理架构图

标注系统初始化会有一个管理员帐号LabelSystem01，密码为pcl123456，安装好标注系统后，管理员使用此帐号进行登录操作。

* 用户登录流程步骤如下：

1. 在登录页面输入用户名、密码登录。
2. 后端应用系统接到登录请求后，对用户名及密码校验，如果通过则转3，否则返回错误的用户名、密码，流程结束。
3. 检查系统缓存中是否有该用户的Token，如果有则使用存在的Token返回到Web面面，流程结束。否则转4
4. 生成一个新的Token，加入到缓存及数据库表中，将此Token返回到Web面面，流程结束。

* 用户登录后操作系统流程如下：

1. 创建数据集。
2. 服务端获取操作携带的Token，校验此Token是否是合法的用户，如果不是，则直接返回异常信息，流程结束。
3. 服务端正常创建数据集。

* 管理员登录后增加用户：

1. 管理员登录成功。
2. 在用户管理中添加用户，输入用户名称、密码、呢称、用户类型、联系方式等信息进行用户增加。
3. 在用户管理中删除用户，前提此用户没有创建过任务或者数据集。
4. 在用户管理中修改用户的密码。

###### 3.2.1.1 数据库表设计

表名：users\_userprofile

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 整数 | 是 | 用户id，主键，自动递增。 |
| password | 字符串(128) | 是 | 密码，使用SHA-256算法加密。 |
| last\_login | 时间 | 否 | 上次登录 |
| is\_superuser | 整数 | 是 | 0为管理员，1是标注人员，2是审核人员 |
| username | 字符串(150) | 是 | 唯一，不能重复 |
| email | 字符串(254) | 否 | 电子邮箱 |
| nick\_name | 呢称(50) | 是 | 呢称 |
| address | 地址(100) | 否 | 用户地址 |
| mobile | 电话(11) | 是 | 移动电话 |
| company | 公司(128) | 否 | 用户所在公司 |

###### 3.2.1.2 接口说明

登录入口类：LoginController

用户管理类：UserController

具体接口，略，可以参见系统的swagger-ui页面。

###### 3.2.1.3 前端原型

用户管理前端原型设计如下图1-4所示：

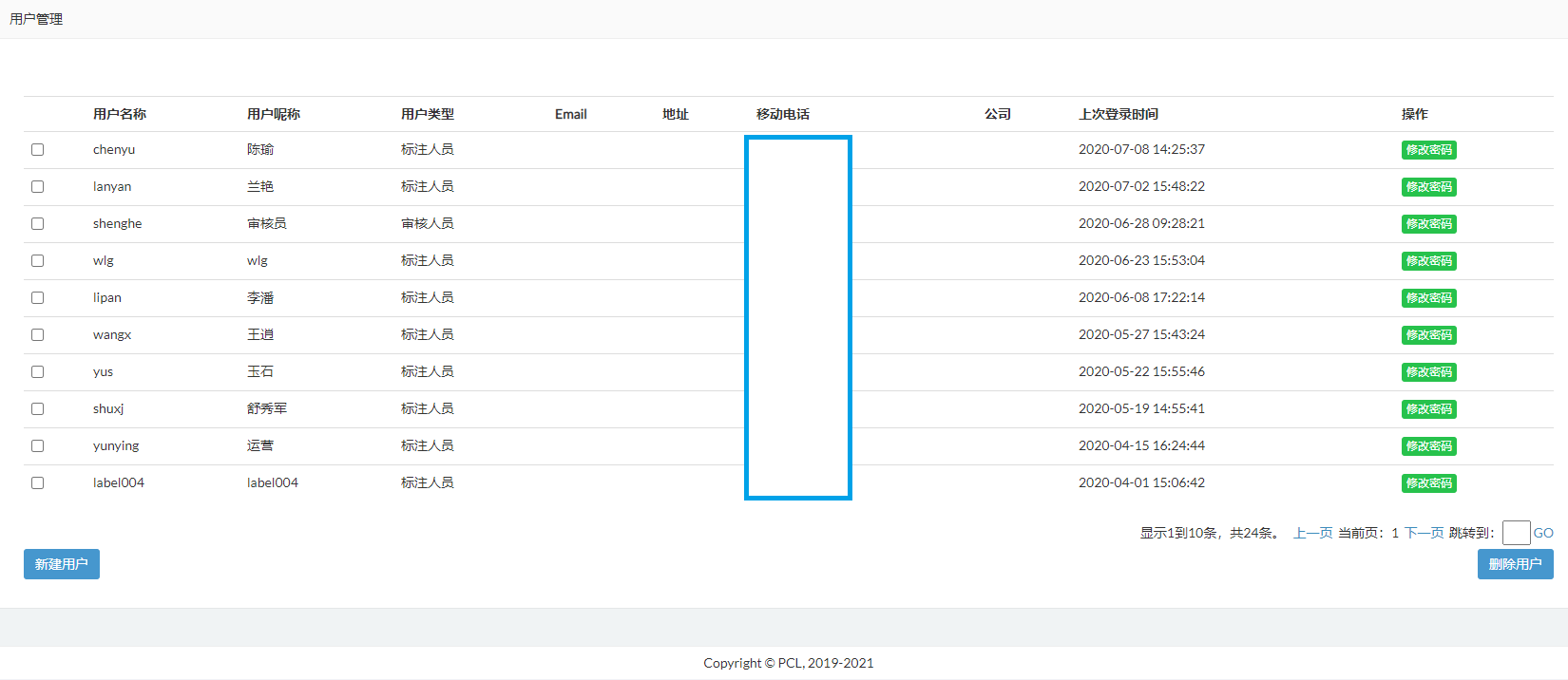


图1-4 用户管理前端原型

###### 3.2.2 数据集管理功能

标注系统是对交通、医疗数据进行标注，而数据集管理是整个标注系统的核心，它管理整个标注系统的数据，而自动标注、人工标注、视频标注、视频流统计、模型重训、ReID标注这些功能都需要用到这些数据来呈现给用户。因此不管这些任务创建多少次，数据据集中的原始数据始终只有一份，从而节约存储空间及简化数据管理。从系统架构中可以看到数据集中标注数据存储在minio系统中，MySQL中只存储数据集任务相关信息。

整个数据数据集管理功能架构及流程图如图1-4所示。

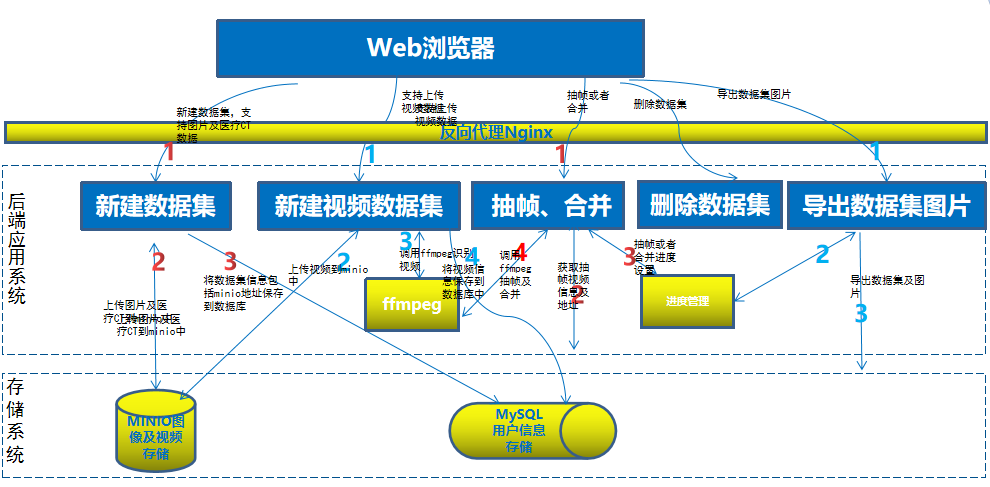


图1-4 数据集管理功能架构及操作流程

用户操作流程描述如下：

* 新建数据集

1. 新建数据集，填写数据集名称及其它信息。
2. 上传图片及医疗数据的压缩包到minio中。
3. 后端应用系统接收到新建数据集请求后，将信息存储到MySQL中。
4. 同时从minio中下载压缩文件，并解压，将每个图片或者每个医疗文件地址存储到MySQL中。如果压缩文件中还有标注信息，则将标注信息与对应的图片关联起来，也存储到MySQL中。最后将压缩包中所有文件数量也存储到MySQL中。

* 新建视频数据集

1. 新建视频数据集，填写数据集名称及其它信息。
2. 上传一个或者多个视频到minio中
3. 后端应用接收到新建视频数据集请求后，将信息存储到MySQL中。
4. 同时从minio中下载视频文件，调用ffmpeg识别视频的信息，并将视频信息存储到MySQL中。

* 对视频抽帧

1. 选择数据集，并设置抽帧参数。
2. 后端应用接收到请求后，获取该数据集的视频信息，并预估所需要的时间，设置任务进度信息。
3. 调用ffmpeg进行抽帧。
4. 抽帧完成后，并图片保存到minio中。并将所有图片在minio中的地址保存到MySQL中。

* 对视频合并

1. 选择数据集，在界面对合并的视频进行排序，并设置好序号。
2. 后端应用接收到请求后，获取所有的视频信息，并预估所需要的时间，设置任务进度信息。
3. 调用ffmpeg按照指定的顺序进行合并。
4. 合并完成之后，删除原来的视频，只保留合并的视频，并将合并的视频重新上传到minio中。并针minio的地址保存到MySQL中。

* 删除数据集

1. 选择数据集。
2. 删除minio中的数据集保存的数据。
3. 删除MySQL中保存的数据集。

* 导出数据集图片

1. 选择数据集。
2. 读取数据集在MySQL中保存的信息，获取其所有图片的minio地址。
3. 根据图片数量设置进度信息。
4. 根据minio地址获取图片数据，并写到下载文件中。

###### 3.2.2.1 数据库表设计

表名：tasks\_dataset， 数据集任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| task\_name | 字符串(400) | 是 | 任务名称 |
| task\_desc | 字符串(1000) | 否 | 任务描述 |
| task\_add\_time | 时间 | 是 | 任务增加时间 |
| dataset\_type | 整数 | 是 | 数据集类型，图片，医疗、视频 |
| total | 整数 | 否 | 数据集的文件数量 |
| zip\_object\_name | 字符串(255) |  | 数据集中数据在minio中的地址 |
| user\_id | 整数 | 是 | 创建的用户id，外键，关联users\_userprofile中id。 |
| assign\_user\_id | 整数 | 否 | 数据集共享的用户id |
| task\_status | 整数 | 否 | 任务状态 |
| file\_bucket\_name | 字符串(255) |  | 抽帧后图片所在minio的目录 |
| camera\_number | 字符串(256) |  | 摄像头编号，只对图片、视频有效。 |
| camera\_gps | 字符串(64) |  | 摄像头GPS，只对图片、视频有效。 |
| camera\_date | 字符串(64) |  | 拍摄日期，只对图片、视频有效。 |
| videoSet | 字符串(4000) |  | 视频集合，用于存储多个视频信息 |
| mainVideoInfo | 字符串(2000) |  | 主视频信息，用于存储该视频的详细信息，包括时长，编码等信息。 |

###### 3.2.2.2 接口说明

数据集管理入口类：DataSetController

文件上传入口类：FileController

具体接口，略，或者参见系统的swagger-ui页面。

###### 3.2.2.3 前端原型

数据集功能前端原型设计如下图1-5所示：

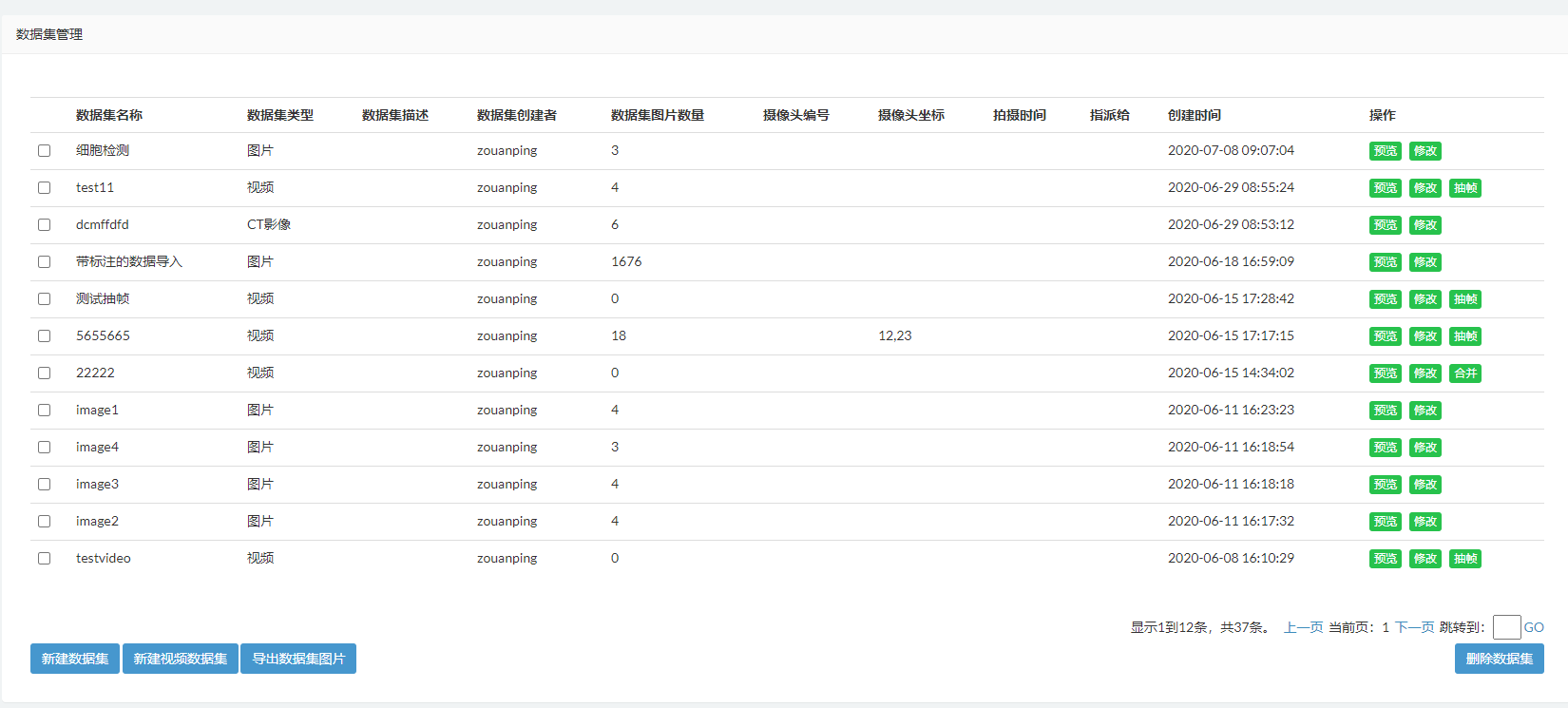


图1-5 数据集管理前端原型

###### 3.2.3 自动标注功能

自动标注是对数据集的数据调用标注系统集成的算法模型进行自动标注，比如集成人、车、人脸、视频追踪、ReID分类、细胞检测等算法模型。

由于人工智能的算法模型是一般是Python编写，因此需要考虑Java与Python之间调用的数据交互问题，同时还要考虑算法模型增加扩展的机制。

Java与Python之间调用的数据交互问题，采用HTTP消息进行交互。在Java程序调用Python时，将HTTP地址作为参数传递给Python脚本，Pyhon程序处理完之后，通过HTTP地址发送回处理消息。

对于算法模型增加的问题，在MySQL数据库中存储所有的算法模型，每个模型有调用的脚本及参数信息，Java程序不关心具体的算法模型，两者约定好数据交互的格式。

处理流程：

1. 输入自动标注任务名称，选择自动标注任务使用的模型。选择需要标注的数据集。
2. 后端应用接收到自动标注任务后，在tasks\_prepredicttasks创建一个任务，并将任务设置为创建状态。
3. 首先从选择的数据集中读取所有需要标注的图片，将图片处理成指定算法模型需要数据集。
4. 调用指定算法模型的脚本进行标注。
5. 读取标注结果，将其存入到数据库中tasks\_prepredictresult。
6. 自动标注任务创建完成，用户可以查看并预览结果。

自动标注功能，本系统内置集成mmdetection 2.3.0版本框架，同时搜集网上开源的人、车训练出两个模型，一个是freeachor，专门检测人，一个是retinaet，检测人、车。同时新增了一个自动分割的模型。

*另外，自动标注功能也方便扩展，即使算法模型与Java程序不在同一个系统中，那么只需要写好双方交互的restful接口，那么也可以接入本系统，可以参见AutoLabelService类中的WISE\_MEDICAL\_MODEL\_ID模型的调用，就是使用restful接口来实现两个系统的交互，并在标注系统中呈现自动标注的结果。*

###### 3.2.3.1 数据库表设计

表名：tasks\_prepredicttasks， 自动标注任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | 主键，UUID |
| task\_name | 字符串(400) | 是 | 任务名称 |
| zip\_object\_name | 字符串(128) | 否 | 暂未使用 |
| task\_start\_time | 时间 | 是 | 任务开始时间 |
| task\_finish\_time | 时间 | 否 | 任务结束时间 |
| task\_status | 整数 | 否 | 任务状态 |
| alg\_model\_id | 整数 | 是 | 使用的算法模型id |
| user\_id | 整数 | 是 | 创建者ID，外键，关联users\_userprofile中id。 |
| zip\_bucket\_name | 字符串(64) | 否 | 暂未使用 |
| task\_status\_desc | 字符串(2000) | 否 | 任务状态描述 |
| dataset\_id | 字符串(64) | 是 | 自动标注的数据集ID |
| delete\_no\_label\_picture | 整数 | 否 | 在自动标注完成后，是否需要自动删除未检测到目标的记录 |

表名：tasks\_prepredictresult， 自动标注结果存储表

属性说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | 主键，UUID |
| pic\_url | 字符串(256) | 否 | 未使用 |
| pic\_object\_name | 字符串(128) | 否 | 未使用 |
| label\_info | TEXT(65535) | 否 | 标注信息，Json格式。 |
| item\_add\_time | 时间 | 是 | 创建时间 |
| pre\_predict\_task\_id | 字符串(32) | 是 | 任务id,外键关联tasks\_prepredicttasks中的id |
| pic\_image\_field | 字符串(400) | 否 | 图片或者文件在minio中的地址 |

###### 3.2.3.2 接口说明

批量的自动标注入口类：PrePredictTaskController

API接口，参见系统的swagger-ui页面。

如下，是标注系统中自动标注之后对目标标注的存储格式：

[

{

"class\_name": "person",

"score": 1,

"box": [

318,

301,

433,

706

],

"id": "0"

},

{

"class\_name": "person",

"score": 1,

"mask": [

202,

301,

250,

456,

308,

683

],

"id": "1"

}

]

class\_name：目标类别

score：得分，用于自动标注中，取值为0---1

box: 标注框的坐标，矩形坐标框为xmin=box[0],ymin=box[1],xmax=box[2],ymax=box[3]

mask:多边形的坐标列表，x1=mask[0],y1=mask[1],x2=mask[2],y2=mask[3]......,用于分割的自动标注。

id：标注框的id，每张图片中的每个标注框的id要唯一。

###### 3.2.3.3 前端原型

自动标注功能前端原型设计如下图1-6所示：

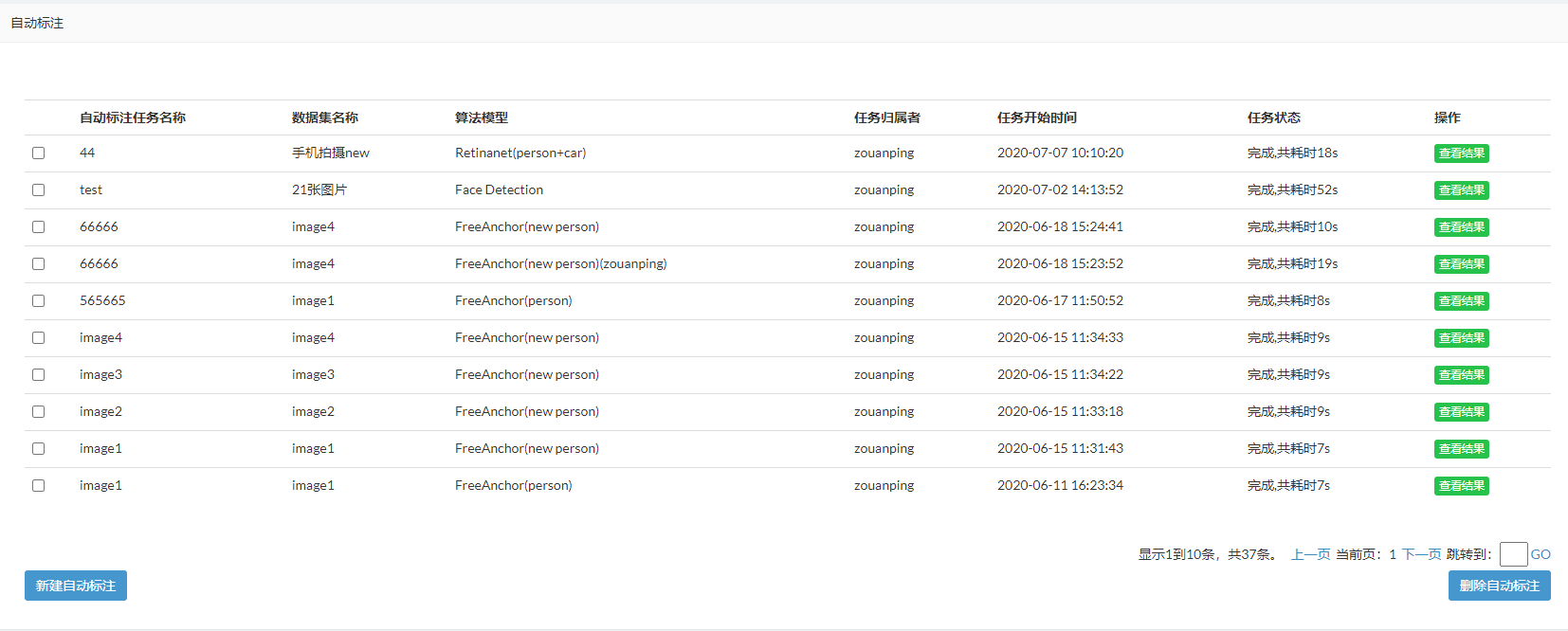


图1-6 自动标注前端原型

###### 3.2.4 人工标注功能

人工标注是对自动标注的一个补充，同时也是对一些没有算法模型作自动检测的数据进行纯人工标注，从而获得标注数据。因此人工标注需要两种模式，一种是基于自动标注的调整，一种是基于数据集的纯人工标注，同时人工标注需要支持图片、医疗各种数据的人工标注，另外还支持对单个图片的自动标注，包括目标检测，目标跟踪等功能。另外人工标注任务应该有进度可查看，另外完成之后，还需要有人进行审核，因此需要有审核流程。

人工标注不光是有任务管理，还需要提供界面可供用户进行标注，标注形状可以是矩形，多边形，圆点，而标注对象的属性可以任意设置，还能支持标注对象的属性导入、导出。同时标注图形可以支持放大、缩小、拖动等操作。

功能处理流程：

* 新建自动标注结果人工标注

选择自动标注任务，输入人工标注任务名称。

1. 后端应用接到请求后，将自动标注的结果拷贝一份，存储到人工标注的数据库表中。
2. 用户在界面对标注结果进行调整，并存储。
3. 标注完成后，可以转审核人员进行审核。
4. 审核人员审核完成后，可以正常进行标注数据导出。

* 新建数据集人工标注

1. 选择数据集对象，输入人工标注任务名称，同时可以直接输入一个标注属性Json。
2. 后端应用接到请求后，将数据集所有图片对象拷贝一份，存储到人工标注的数据库表中。

3、用户在界面对标注结果进行调整，并存储。

4、标注完成后，可以转审核人员进行审核。

5、审核人员审核完成后，可以正常进行标注数据导出。

新建数据集人工标注功能中，新增了一个主动学习分类的功能，即一个大数据集的人工标注，在标注了100张图片左右的时候，标注系统会定时在晚上自动学习，并对剩下未标注的图片进行排序，将权重较大的图片排列在前面供标注人员进行标注，其实现原理参照：<https://github.com/yuantn/MIAL/raw/master/paper.pdf> （ CVPR 2021）

###### 3.2.4.1 数据库表设计

表名：tasks\_labeltask， 数据集任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| task\_name | 字符串(400) | 是 | 任务名称 |
| task\_add\_time | 时间 | 是 |  |
| relate\_task\_id | 字符串(32) | 否 | 关联的任务ID，根据task\_type决定是关联的自动标注任务，或者是人工标注任务。 |
| relate\_task\_name | 字符串(64) | 否 |  |
| finished\_picture | 整数 | 否 | 标注完成图片数量 |
| total\_picture | 整数 | 否 | 任务需要标注总图片数量 |
| task\_type | 整数 | 否 | 任务类型 |
| zip\_object\_name | 字符串(64) | 否 | 暂时未使用 |
| zip\_bucket\_name | 字符串(64) | 否 | 暂时未使用 |
| user\_id | 整数 | 是 | 创建者ID，外键，关联users\_userprofile中id。 |
| assign\_user\_id | 整数 | 否 | 指派给某个用户的id |
| relate\_other\_label\_task | 字符串(400) | 否 | 用于医疗审核模式下关联的其它tasks\_labeltask记录的id |
| task\_flow\_type | 整数 | 否 | 用于医疗审核模式下任务工作流类型 |
| task\_label\_type\_info | 字符串(4000) | 否 | 人工标注的用户自定义设置的属性，Json格式。 |

表名：tasks\_labeltaskitem， 人工标注结果存储表

属性说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | 主键，UUID |
| pic\_url | 字符串(256) | 否 | 未使用 |
| pic\_object\_name | 字符串(128) | 否 | 用作图片长宽width,height |
| label\_info | TEXT(65535) | 否 | 标注信息，Json格式。 |
| label\_task\_id | 字符串(32) | 是 | 任务id,外键关联tasks\_labeltask中的id |
| item\_add\_time | 时间 | 是 | 创建时间 |
| pic\_image\_field | 字符串(400) | 否 | 图片或者文件在minio中的地址 |
| label\_status | 整数 | 否 | 任务状态，0，完成，1未完成 |
| display\_order1 | 整数 | 否 | 扩展属性，用来排序显示 |
| display\_order2 | 整数 | 否 | 扩展属性，用来排序显示，暂未使用 |
| verify\_status | 整数 | 否 | 审核状态 |
| verify\_desc | 字符串(400) | 否 | 审核描述 |

###### 3.2.4.2 接口说明

人工标注入口类：LabelTaskController

标注属性的导入导出：LabelPropertyController

人工标注页面单个图片的自动标注或者目标跟踪：AutoLabelController

具体接口，参见系统的swagger-ui页面。

标注属性格式说明：

{

"id":{

"type":"text",

"description":"",

"default\_value":""

},

"type":{

"type":"dropdown",

"description":"",

"options":{

"car":"car",

"person":"person",

"non-motor":"non-motor"

},

"default\_options":{

}

},

"color":{

"type":"checkbox",

"description":"",

"options":{

"upblack":"",

"upwhite":"",

"upred":"",

"uppurple":"",

"upyellow":"",

"upgray":"",

"upblue":"",

"upgreen":"",

"downblack":"",

"downwhite":"",

"downpink":"",

"downpurple":"",

"downyellow":"",

"downgray":"",

"downblue":"",

"downgreen":"",

"downbrown":""

},

"default\_options":{

}

}

}

第一层的id、type、color、otherprop代表标注属性的名称。

第二层的type代表标注属性呈现类型，目前支持四种：

text：文本输入

附属属性：default\_value，文本输入的默认值。

dropdown：下拉选择

附属属性：options，下拉框的选项

附属属性：default\_options，默认的选项

checkbox：多选checkbox

附属属性：options，多选框的选项

附属属性：default\_options，默认的选项

radio:单选框

附属属性：options，多选框的选项

附属属性：default\_options，默认的选项

第二层的description代表属性的名称的描述。

标注系统中人工标注的存储格式（后续的标注功能，如果没有特别说明，都是这个格式）：

[

{

"class\_name": "person",

"box": [

318,

301,

433,

706

],

"id": "0",

"blurred": false,

"goodIllumination": true,

"frontview": true,

"other": {

"region\_attributes": {

"type": "person",

"id": "0"

}

}

},

{

"class\_name": "person",

"mask": [

202,

301,

250,

456,

308,

683

],

"id": "1",

"blurred": false,

"goodIllumination": true,

"frontview": true,

"other": {

"region\_attributes": {

"type": "person",

"id": "1"

}

}

}

]

class\_name：目标类别

box: 标注框的坐标，矩形坐标框为xmin=box[0],ymin=box[1],xmax=box[2],ymax=box[3]

mask:多边形的坐标列表，x1=mask[0],y1=mask[1],x2=mask[2],y2=mask[3]......

keypoints:点标注的坐标，x1=keypoints[0],y1=keypoints[1]

id：标注框的id，每张图片中的每个标注框的id要唯一

other--->region\_attributes-->中存储为标注工具界面自定义的其它属性。其中type与class\_name一样，id与外层id一样。

###### 3.2.4.3 前端原型

人工标注管理功能前端原型设计如下图1-7所示，人工标注工具界面前端原型如下图1-8所示。

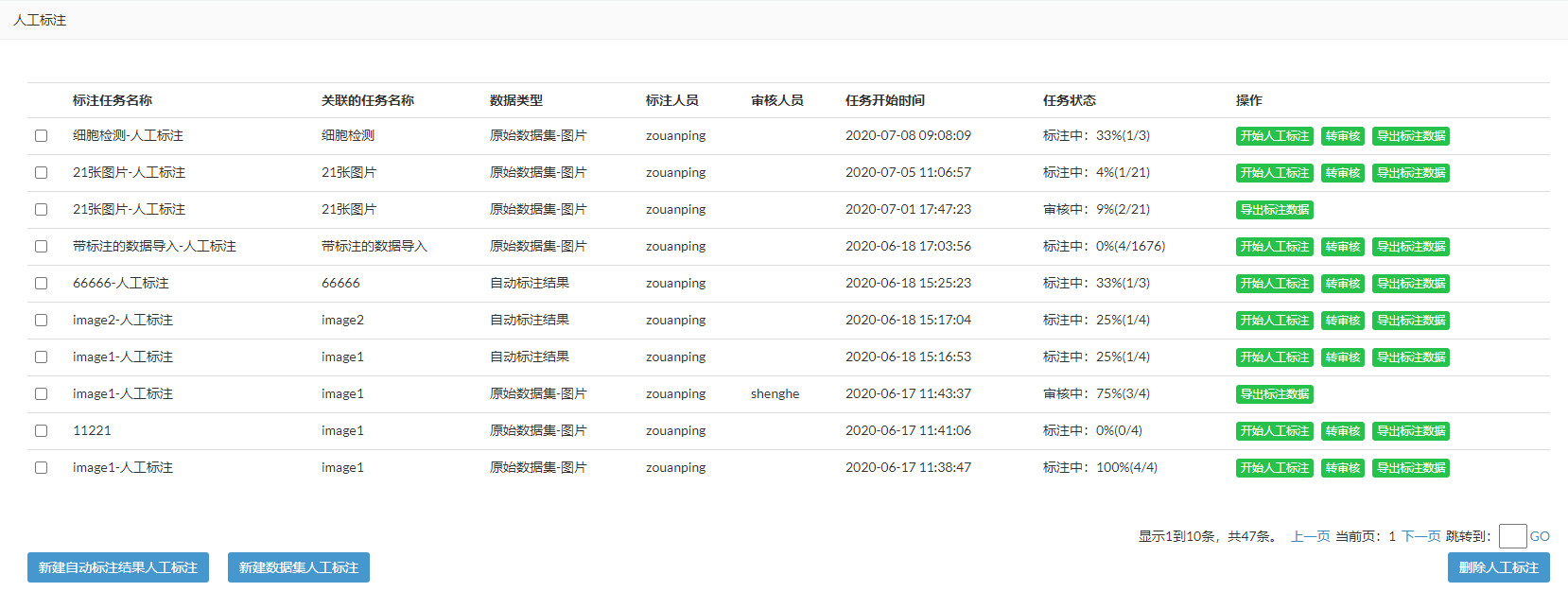


图1-7 人工标注管理界面前端原型



图1-8 人工标注工具界面前端原型

###### 3.2.5 ReID标注功能

目前人工智能的检测识别这一块，ReID的精度还没有做到很高，因此这一块需要大量的数据，所以标注系统必需要支持ReID的标注，由于ReID的标注跟普通的标注不一样，它是很多目标是相关联的，因此需要设计一个易用较好的标注方式，因此需要有一个界面直观显示当前用户ReID标注的结果集，并且在这个结果集中也可以标注进行微调。在标注完成之后，可以对ReID的结果进行导出，结果导出时可以将原图进行抠图导出，也可以只导出原图。

功能处理流程：

* 新建自动标注结果ReID标注（使用京东的开源fast-reid算法）

1. 选择源自动标注结果，选择一个或者多个对照自动标注结果，输入ReID标注任务名称，选择标注对象（人，车），选择是否进行自动标注ReID。
2. 后端接到请求后，将任务设置为创建状态。
3. 将源自动标注结果中已经标注的对象进行抠图导出，生成ReID分类的查询数据集。
4. 遍历对照自动标注结果，进行以下5--7循环处理：
5. 将对照自动标注结果已经标注的对象抠图导出，生成ReID分类的测试数据集。
6. 调用ReID自动分类算法，对查询数据集每个抠图，找出最相近测试数据集中欧式距离最近的5张图，将其存储到数据库中。
7. 如果选择了自动标注ReID，对查询数据集每个抠图，找出最相近测试数据集中欧式距离最近的第1张图，对他们进行同一ReID标注，从1开始。

8、新建自动标注结果ReID标注完成，将任务状态设置完成，用户可以开始进行人工标注。

* 新建数据集ReID标注

1. 选择源数据集，选择一个或者多个对照数据集，输入ReID标注任务名称，选择标注对象（人，车）。
2. 后端接到请求后，将源数据集及所有的对照数据集的图片进行存储到ReID数据库表中。
3. 创建完成，用户可以进行人工标注。
4. 也可以对某个目标进行手动画框，之后使用单目标跟踪对目标进行自动跟踪。

ReID标注结果页面可以对标注结果进行重新分类，也可以进行调整，包括对单个图片修改、删除或者整个ReID下所有图片的操作。

###### 3.2.5.1 数据库表设计

表名：tasks\_reidtask， ReID任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| task\_name | 字符串(400) | 是 | 任务名称 |
| src\_predict\_taskid | 字符串(128) | 否 | 源任务id（自动标注任务或者数据集） |
| dest\_predict\_taskid | 字符串(4000) | 否 | 对照任务id列表（自动标注任务或者数据集），Json格式。 |
| task\_start\_time | 时间 | 是 | 任务开始时间 |
| task\_finish\_time | 时间 | 否 | 任务完成时间 |
| task\_status | 整数 | 是 | 任务状态 |
| alg\_model\_id | 整数 | 否 | 使用的算法id |
| user\_id | 整数 | 是 | 创建者ID，外键，关联users\_userprofile中id。 |
| src\_bucket\_name | 字符串(64) | 否 | 暂时未使用 |
| dest\_bucket\_name | 字符串(64) | 否 | 暂时未使用 |
| task\_status\_desc | 字符串(1000) | 否 | 任务状态描述 |
| assign\_user\_id | 整数 | 否 | 指派给某个用户操作 |
| relate\_other\_label\_task | 字符串(400) | 否 | 暂时未使用 |
| task\_flow\_type | 整数 | 否 | 暂时未使用 |
| task\_type | 整数 | 否 | 用来标识是自动标注任务还是数据集任务 |
| task\_label\_type\_info | 字符串(4000) | 否 | 标注的用户自定义设置的属性，Json格式。 |
| finished\_picture | 整数 | 否 | 暂未使用 |
| total\_picture | 整数 | 否 | 暂未使用 |
| reid\_obj\_type | 整数 | 否 | 标注对象是人还是车 |
| verify\_user\_id | 整数 | 否 | 审核用户的id |

表名：tasks\_reidtaskitem， ReID标注信息存储表

属性说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | 主键，UUID |
| pic\_url | 字符串(256) | 否 | 未使用 |
| pic\_object\_name | 字符串(128) | 否 | 用作图片长宽width,height |
| label\_info | TEXT(65535) | 否 | 标注信息，Json格式。 |
| label\_task\_id | 字符串(32) | 是 | 任务id,外键关联tasks\_labeltask中的id |
| item\_add\_time | 时间 | 是 | 创建时间 |
| pic\_image\_field | 字符串(400) | 否 | 图片或者文件在minio中的地址 |
| label\_status | 整数 | 否 | 任务状态，0，完成，1未完成 |
| display\_order1 | 整数 | 否 | 扩展属性，用来排序显示 |
| display\_order2 | 整数 | 否 | 扩展属性，用来排序显示，暂未使用 |
| verify\_status | 整数 | 否 | 审核状态 |
| verify\_desc | 字符串(400) | 否 | 审核描述 |

###### 3.2.5.2 接口说明

ReID标注入口类：ReIDTaskController

略，参见系统的swagger-ui页面

###### 3.2.5.3 前端原型

ReID标注管理功能前端原型设计如下图1-9所示，ReID标注工具界面前端原型如下图1-10所示。

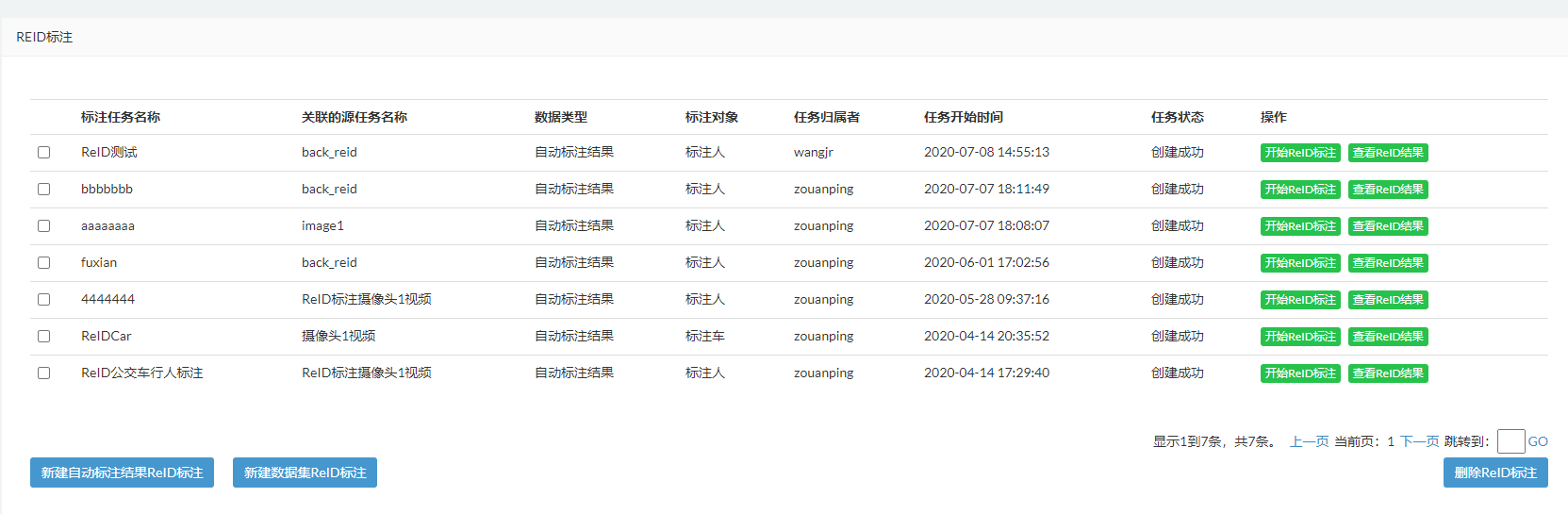


图1-9 ReID标注管理界面前端原型



图1-10 ReID标注管理界面前端原型

###### 3.2.6 车流统计功能

车流统计功能是一个比较简单的标注功能，用户需要观看视频，统计某个时间段在某个十字路口有多少车辆经过，并将此车类型及行驶方向记录下来。

功能处理流程：

* 新建车流统计标注

1. 选择视频数据集对象，输入车流统计任务名称。
2. 后端应用接收到请求后，在tasks\_videocounttask插入一条任务记录。任务创建完成。
3. 用户开始在界面边播放视频边进行车流统计标注，车流统计标注存储到tasks\_videocounttaskitem中。
4. 用户可以拖动视频，加快进度，后端接收到请求后，返回指定时间的视频数据流。
5. 标注完成后，可以导出车流统计标注信息。

###### 3.2.6.1 数据库表设计

表名：tasks\_videocounttask， 车流统计任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| task\_name | 字符串(400) | 是 | 任务名称 |
| dataset\_id | 字符串(32) | 是 | 视频数据集id |
| task\_add\_time | 时间 | 是 | 任务增加时间 |
| task\_finish\_time | 时间 | 否 | 任务完成时间 |
| task\_status | 整数 | 是 | 任务状态 |
| zip\_object\_name | 字符串(255) | 否 | 视频在minio中的路径 |
| zip\_bucket\_name | 字符串(255) | 否 | 暂时未使用 |
| user\_id | 整数 | 是 | 创建者ID，外键，关联users\_userprofile中id。 |
| task\_status\_desc | 字符串(1000) | 否 | 任务状态描述 |
| assign\_user\_id | 整数 | 否 | 指派给某个用户操作 |
| verify\_user\_id | 整数 | 否 | 审核用户的id |

表名：tasks\_videocounttaskitem， 车流标注信息存储表

属性说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | 主键，UUID |
| pic\_url | 字符串(256) | 否 | 未使用 |
| pic\_object\_name | 字符串(128) | 否 | 用作图片长宽width,height |
| label\_info | TEXT(65535) | 否 | 标注信息，Json格式。与其它的标注信息有点差别。 |
| label\_task\_id | 字符串(32) | 是 | 任务id,外键关联tasks\_labeltask中的id |
| item\_add\_time | 时间 | 是 | 创建时间 |
| pic\_image\_field | 字符串(400) | 否 | 图片或者文件在minio中的地址 |
| label\_status | 整数 | 否 | 任务状态，0，完成，1未完成 |
| display\_order1 | 整数 | 否 | 扩展属性，用来排序显示 |
| display\_order2 | 整数 | 否 | 扩展属性，用来排序显示，暂未使用 |
| verify\_status | 整数 | 否 | 审核状态 |
| verify\_desc | 字符串(400) | 否 | 审核描述 |

###### 3.2.6.2 接口说明

车流统计入口类：VideoCountController

车流标注格式说明：

{

"time":"00:00:24.727",

"traj\_mode":"DU",

"road\_id":"",

"car\_category":"1",

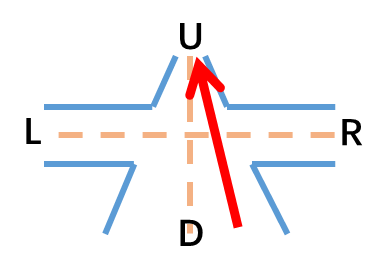
"color":"2",

"loc":"X:219;Y:217"

}

time:视频帧所在时间

traj\_mode:车辆行进方向，如下图所示，共12种



road\_id：车道id

car\_category:车辆类型

color：车辆颜色

loc：车辆在帧上的坐标

略，参见系统的swagger-ui页面

###### 3.2.6.3 前端原型

车流统计任务管理功能前端原型设计如下图1-11所示，车流统计标注工具界面前端原型如下图1-12所示。



图1-11 车流统计任务管理界面前端原型

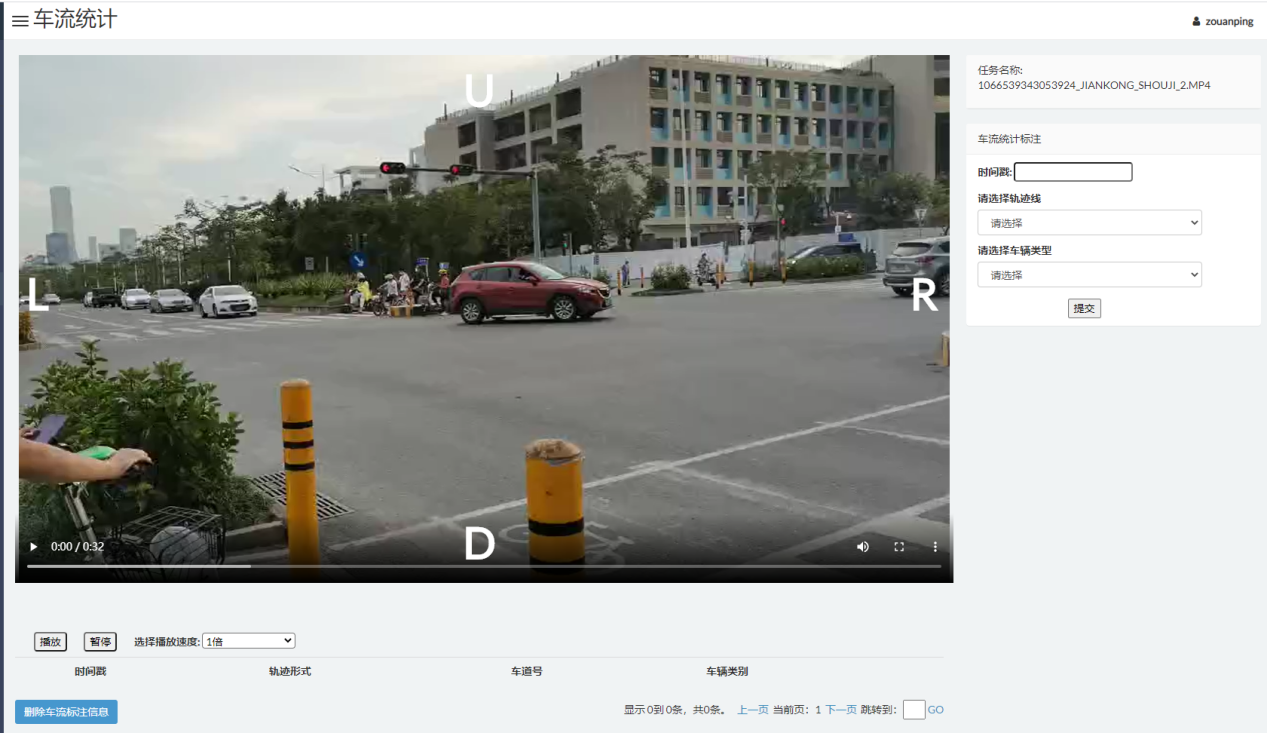


图1-12 车流统计标注工具界面前端原型

###### 3.2.7 视频标注功能

视频标注功能是针对视频进行标注，方便用户只需要视频的部分目标进行标注，用户可以播放视频，然后点击暂停，在暂停界面进行标注。标注的过程与方式与人工标注基本一样。

功能处理流程：

* 新建视频标注

1. 选择视频数据集对象，输入视频标注任务名称。

2、后端应用接收到请求后，在tasks\_videolabeltask插入一条任务记录。任务创建完成。

3、用户开始在界面边播放视频边进行视频标注，当暂停时对视频中某帧图片进行人工标注或者自动。

4、后端应用接收到指定时间的某帧图片时，先调用ffmpeg对指定时间进行抽帧，并将其上传到minio中，并存储到tasks\_videolabeltaskitem中。

5、如果用户选择了自动对某帧图片进行标注，那么会调用指定算法对其进行自动标注，标注结果存入tasks\_videolabeltaskitem中。

6、前端界面显示更新标注信息。

###### 3.2.7.1 数据库表设计

表名：tasks\_videolabeltask， 视频标注任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| task\_name | 字符串(400) | 是 | 任务名称 |
| dataset\_id | 字符串(32) | 是 | 视频数据集id |
| task\_add\_time | 时间 | 是 | 任务增加时间 |
| task\_finish\_time | 时间 | 否 | 任务完成时间 |
| task\_status | 整数 | 是 | 任务状态 |
| zip\_object\_name | 字符串(255) | 否 | 视频在minio中的路径 |
| zip\_bucket\_name | 字符串(255) | 否 | 暂时未使用 |
| user\_id | 整数 | 是 | 创建者ID，外键，关联users\_userprofile中id。 |
| task\_status\_desc | 字符串(1000) | 否 | 任务状态描述 |
| assign\_user\_id | 整数 | 否 | 指派给某个用户操作 |
| mainVideoInfo | 字符串(2000) | 否 | 视频标注的视频详细信息存储 |
| verify\_user\_id | 整数 | 否 | 审核用户的id |

表名：tasks\_videolabeltaskitem， 视频标注信息存储表

属性说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | 主键，UUID |
| pic\_url | 字符串(256) | 否 | 未使用 |
| pic\_object\_name | 字符串(128) | 否 | 用作图片长宽width,height |
| label\_info | TEXT(65535) | 否 | 标注信息，Json格式。 |
| label\_task\_id | 字符串(32) | 是 | 任务id,外键关联tasks\_labeltask中的id |
| item\_add\_time | 时间 | 是 | 创建时间 |
| pic\_image\_field | 字符串(400) | 否 | 图片或者文件在minio中的地址 |
| label\_status | 整数 | 否 | 任务状态，0，完成，1未完成 |
| display\_order1 | 整数 | 否 | 扩展属性，用来排序显示 |
| display\_order2 | 整数 | 否 | 扩展属性，用来排序显示，暂未使用 |
| verify\_status | 整数 | 否 | 审核状态 |
| verify\_desc | 字符串(400) | 否 | 审核描述 |

###### 3.2.7.2 接口说明

视频标注入口类：VideoLabelController

具体接口，参见系统的swagger-ui页面

###### 3.2.7.3 前端原型

视频标注任务管理功能前端原型设计如下图1-13所示，视频标注工具界面前端原型如下图1-14所示。



图1-13 视频标注任务管理界面前端原型

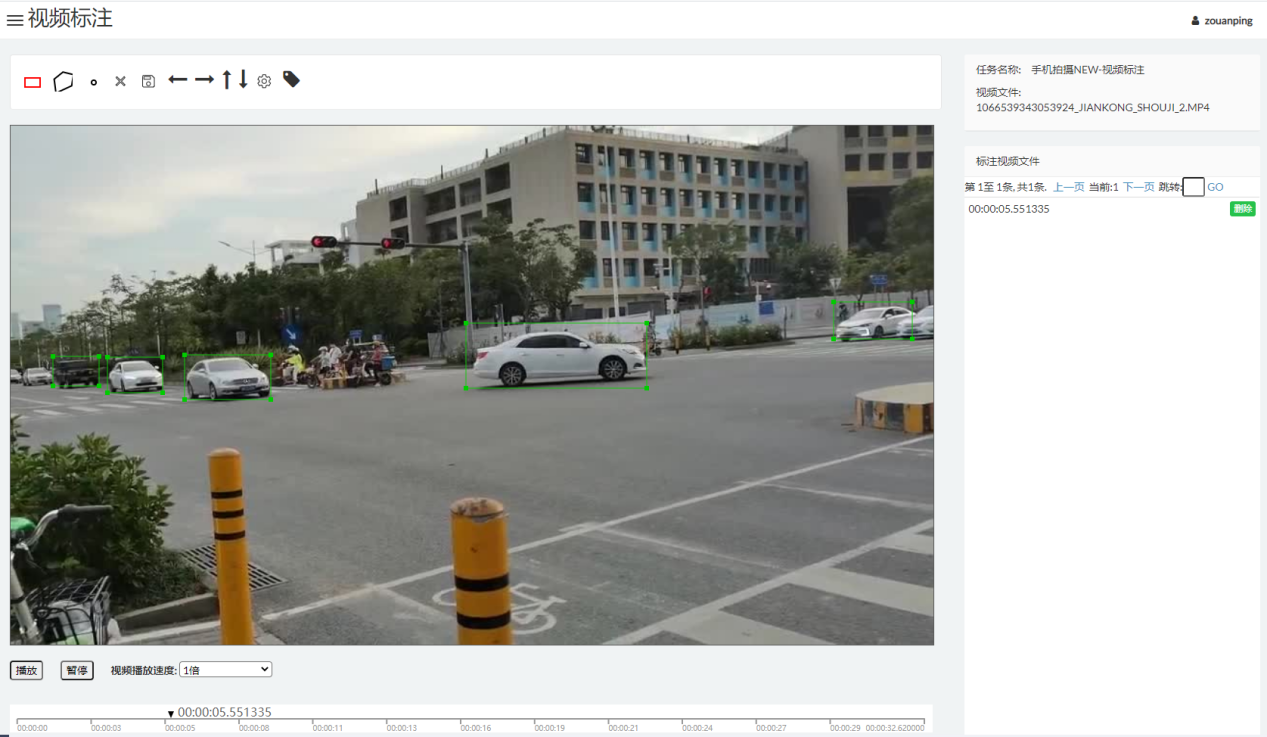


图1-14 视频标注工具前端原型

###### 3.2.8 模型管理及重训功能

半自动标注系统要能自动检测对象，那么一些好的模型算法必不可少，同时考虑到模型可能随时会换，并且更新，因此后端应用与模型算法应该要脱勾。同时我们系统是用来标注数据的，因此对模型进行重训也是一个必不可少的功能，可以用来提升已有算法模型的检测或者识别的精度。

功能处理流程：

* 模型管理

目前需要调用API接口创建、删除、查询模型。

* 模型重训或者初始训练

1. 前端用户输入重训名称，并选择基准模型并同时选择训练方式、输入训练后的模型名称，选择要训练的数据及输入训练与测试划分比例。
2. 后端接到请求后，加入到重训队列中。下面步骤3--5为循环处理。
3. 从队列中获取到请求后，开始组织重训的增量数据，即将用户选择的人工标注数据集中的数据下载到训练数据集中，并根据测试及训练集比例进行划分。
4. 调用重训脚本，并进行阻塞2天。
5. 重训完成之后，调用API接口创建新的模型，此时可以在自动标注选择模型的时候看到重训的模型。
6. 在重训的过程中，如果界面上要显示重训结果，则后端应用读取重训的过程结果并返回给界面进行显示。

###### 3.2.8.1 数据库表设计

表名：alg\_warehouse\_alginstance， 算法模型目录

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 整数 | 是 | 算法id，主键自增 |
| alg\_name | 字符串(400) | 是 | 算法名称 |
| add\_time | 时间 | 是 | 增加时间 |
| alg\_type\_name | 字符串(400) | 是 | 算法大类名称 |
| alg\_root\_dir | 字符串(400) | 否 | 算法根目录 |

表名：alg\_warehouse\_algmodel， 模型实例表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 整数 | 是 | 模型id，主键自增 |
| conf\_path | 字符串(400) | 是 | 模型配置文件路径 |
| model\_name | 字符串(32) | 是 | 模型实例名称 |
| local\_path | 时间 | 是 | 暂时未使用 |
| model\_url | 时间 | 否 | 暂时未使用 |
| alg\_instance\_id | 整数 | 是 | 模型归属的算法id |
| exec\_script | 字符串(500) | 否 | 模型执行脚本 |
| train\_script | 字符串(500) | 否 | 模型训练脚本 |

表名：tasks\_retraintasks， 重训任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| task\_name | 字符串(400) | 是 | 重训任务名称 |
| task\_start\_time | 时间 | 否 | 任务开始时间 |
| task\_finish\_time | 时间 | 否 | 任务结束时间 |
| task\_status | 整数 | 是 | 任务状态 |
| task\_status\_desc | 字符串(2000) | 否 | 任务状态描述 |
| alg\_model\_id | 整数 | 是 | 使用算法模型id，外键，关联alg\_warehouse\_algmodel中的id |
| pre\_predict\_task\_id | 字符串(32) | 是 | 重训完成之后，重新运行的自动标注任务id |
| user\_id | 整数 | 是 | 创建者ID，外键，关联users\_userprofile中id。 |
| pid | 整数 | 是 | 重训进程id |

###### 3.2.8.2 接口说明

模型重训的入口类：ReTrainController

具体接口，参见系统的swagger-ui页面

###### 3.2.8.3 前端原型

重训任务管理功能前端原型设计如下图1-15所示，重训过程界面前端原型如下图1-16所示。

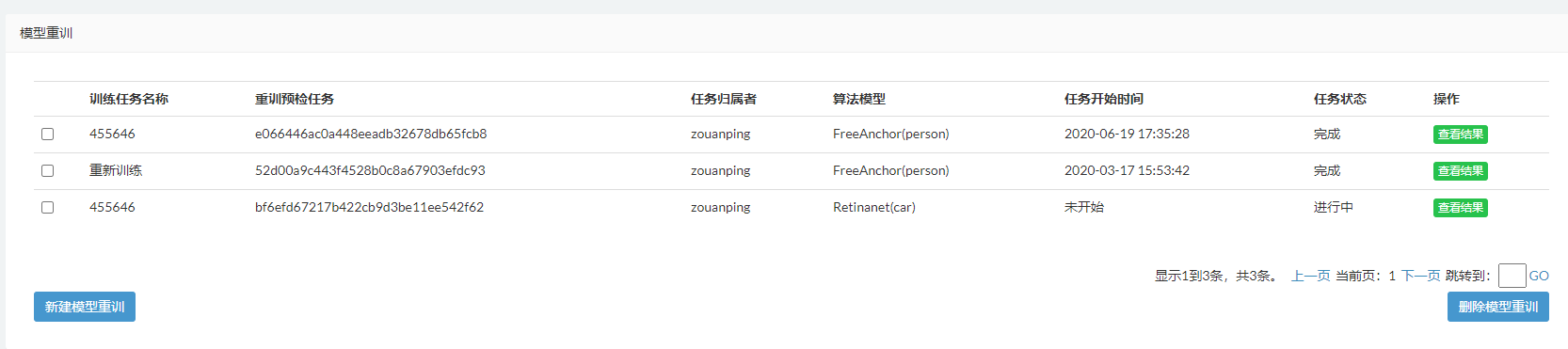


图1-15 重训任务管理前端原型

图1-16 重训任务过程前端原型

###### 3.2.9 报表管理

报表管理，统计标注人员的每天标注情况，可以作各个维度的对比。目前的报表管理是统计各个用户的操作日志来确定每个用户的操作情况。

对于人工标注任务、ReID标注任务、视频标注任务、车流统计来说，标注人员标注一张图片，保存一次，那么系统就会对比这一次的差异，如果有差异，那么“标注图片数量”会加1，同时如果有新建标注，那么“新建标注数量”加1，如果有修改标注，那么“修改标注数量”加1，同时统计新建标注和修改标注的设置属性数量。

对于人工标注任务，审核人员审核一张图片时，每保存一次，那么系统也会比较这一次的差异，如果有差异，那么“标注图片数量”会加1，如果有修改标注，那么“修改标注数量”加1，同时统计设置不合格的属性数量，将其加到对应的标注人员用户的不合格数量上去。

报表统计每隔4个小时统计一次日志信息，用户如果有产生日志，则就会产生该用户本天的数据。

###### 3.2.9.1 数据库表设计

表名：log\_info， 日志表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| oper\_type | 整数 | 否 | 操作类型 |
| user\_id | 整数 | 是 | 用户id |
| oper\_name | 字符串(200) | 否 | 操作名称 |
| oper\_id | 字符串(200) | 否 | 操作名称对应的id |
| oper\_json\_content\_old | TEXT(65535) | 否 | 上一次操作内容 |
| oper\_json\_content\_new | TEXT(65535) | 否 | 新的操作内容 |
| oper\_time\_start | 时间 | 是 | 开始操作时间 |
| oper\_time\_end | 时间 | 是 | 结束操作时间 |
| record\_id | 字符串(64) | 否 | 操作记录的id |
| extend1 | 字符串(400) | 否 | 扩展属性，暂时未用 |
| extend2 | 字符串(400) | 否 | 扩展属性，暂时未用 |

表名：report\_label\_task， 用户报表数据存储表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| user\_id | 整数 | 是 | 用户id |
| oper\_time | 时间 | 是 | 操作时间 |
| rectUpdate | 整数 | 否 | 更新的标注数量 |
| rectAdd | 整数 | 否 | 增加的标注数量 |
| properties | 整数 | 否 | 设置的属性数量 |
| pictureUpdate | 整数 | 否 | 图片更新数量 |

###### 3.2.9.2 接口说明

略，参见系统的swagger-ui页面

###### 3.2.9.3 前端原型

报表管理功能前端原型设计如下图1-17所示，报表度量界面前端原型如下图1-18所示。

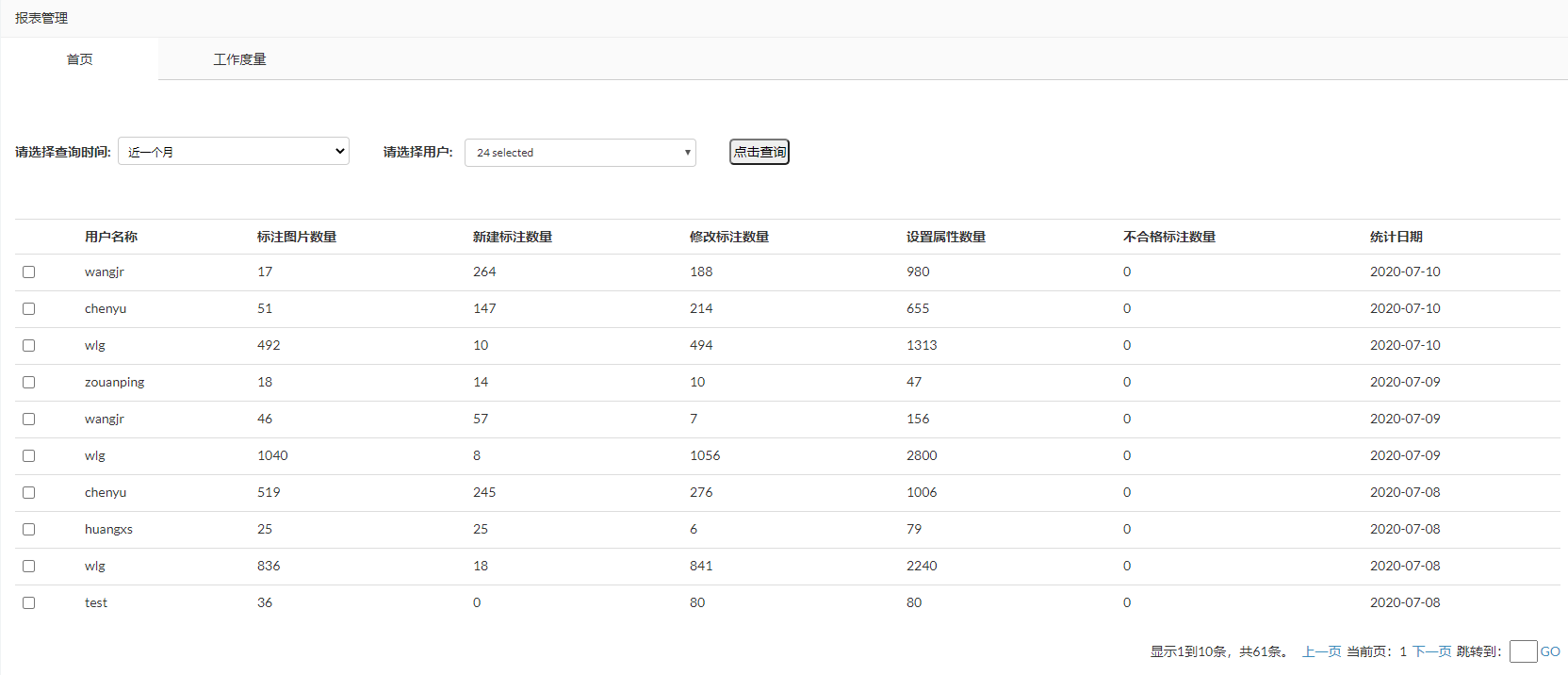


图1-17 报表数据前端原型



图1-18 报表度量前端原型

###### 3.2.10 超大图像标注

超大图像标注功能，是针对单个图像大小在20M以上的图像进行标注，目前设计是针对病理图像，当然可以非常容易扩展到任意超大图像。

超大图像标注原理是在新建数据集的时候，如果用户选择超大图像格式，那么在创建数据集时，就会将此图像在后端进行dzi格式分层，采用的是python组件pyvips来进行分层处理，将超大图像分层后的所有图片进行上传到minio中，分层的动作仅仅在创建数据集的过程中处理一次，后续标注的时候，前端页面读取的是分层的图像，前端读取及呈现图像的组件采用的是openseadragon组件，而要在其上进行标注，我们进行重新开发，可以对超大图像进行标注，存储的标注坐标是针对的原始图像的像素坐标。

###### 3.2.10.1 数据库表设计

表名：tasks\_largepicturelabeltask， 超大图像标注任务表

属性设置：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | UUID |
| task\_name | 字符串(400) | 是 | 任务名称 |
| dataset\_id | 字符串(32) | 是 | 超大图像数据集id |
| task\_add\_time | 时间 | 是 | 任务增加时间 |
| task\_finish\_time | 时间 | 否 | 任务完成时间 |
| task\_status | 整数 | 是 | 任务状态 |
| zip\_object\_name | 字符串(255) | 否 | 原始超大图像在minio中的路径 |
| zip\_bucket\_name | 字符串(255) | 否 | 暂时未使用 |
| user\_id | 整数 | 是 | 创建者ID，外键，关联users\_userprofile中id。 |
| task\_status\_desc | 字符串(1000) | 否 | 任务状态描述 |
| assign\_user\_id | 整数 | 否 | 指派给某个用户操作 |
| mainVideoInfo | 字符串(2000) | 否 | 超大图像标注的dzi信息描述Json信息 |
| appid | 字符串(200) | 否 | 扩展属性 |

表名：tasks\_largepicturetaskitem， 超大图像标注信息存储表

属性说明：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名称 | 类型及长度 | 是否必填 | 说明 |
| id | 字符串(32) | 是 | 主键，UUID |
| pic\_url | 字符串(256) | 否 | 未使用 |
| pic\_object\_name | 字符串(128) | 否 | 未使用 |
| label\_info | TEXT(65535) | 否 | 标注信息，Json格式。 |
| label\_task\_id | 字符串(32) | 是 | 任务id,外键关联tasks\_labeltask中的id |
| item\_add\_time | 时间 | 是 | 创建时间 |
| pic\_image\_field | 字符串(400) | 否 | 未使用 |
| label\_status | 整数 | 否 | 任务状态，0，完成，1未完成 |
| display\_order1 | 整数 | 否 | 扩展属性，用来排序显示 |
| display\_order2 | 整数 | 否 | 扩展属性，用来排序显示，暂未使用 |
| verify\_status | 整数 | 否 | 审核状态 |
| verify\_desc | 字符串(400) | 否 | 审核描述 |

###### 3.2.10.2 接口说明

超大图像标注入口类：LargePictureController

略，参见系统的swagger-ui页面

###### 3.2.10.3 前端原型

如下图1-19所示：

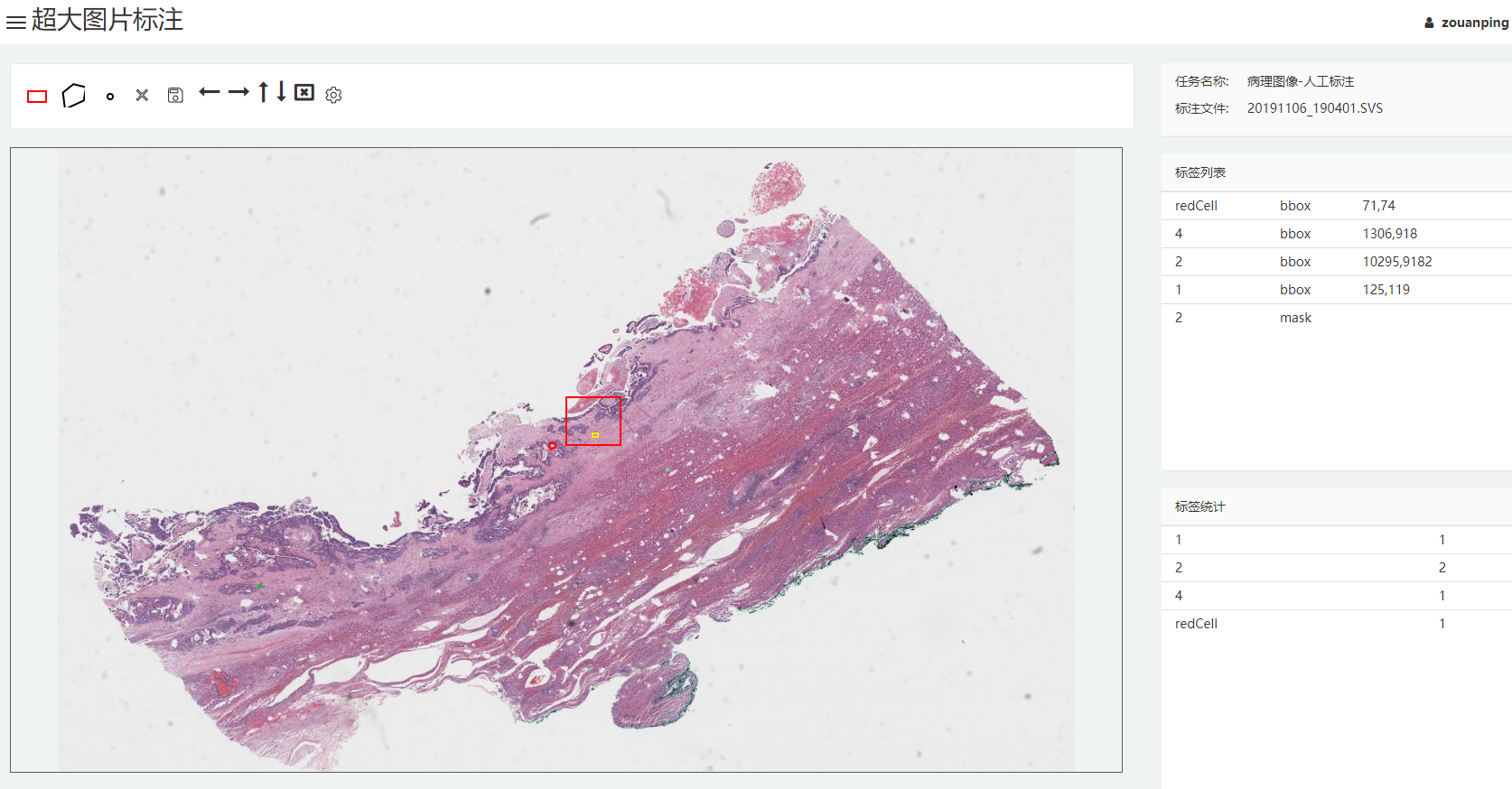


图1-19 超大图标标注前端原型