



**北京航空航天大学**  
B E I H A N G U N I V E R S I T Y

## 医用视网膜血管分割辅助诊断系统

2022 年 4 月

## 摘要

为充分发挥深度学习算法在眼底视网膜分割方面的优势，推动人工智能在医疗领域的落地转化应用，本项目开发一款医用眼底视网膜分割辅助诊断应用系统。主要采用基于双编码器的动态图卷积神经网络，大幅提高了视网膜分割的效率和精度，在 DRIVE 数据集上的分割准确率达到了 96.87%，并经过了北京友谊医院眼科临床医生所采用的五十张眼底视网膜图片的验证。搭建了一套眼底视网膜血管图像分割及临床辅助诊断系统，在校内服务器部署网页访问端，临床医生在校外可通过 VPN 访问校内服务器登录辅助诊断系统，以低廉的成本支持大规模医疗设备集群运行，为临床辅助诊断提供定量的分析工具。针对糖尿病、青光眼等眼底疑难杂症，提供了眼底视网膜金标准和人工智能算法分割结果的对比展示和标注结果，支持对患者点对点的跟踪治疗和会诊功能，具有重要的临床应用价值。

**关键词** 人工智能辅助诊断，实用型临床医用系统开发，视网膜分割算法

# 目录

摘要.....	i
图目录.....	ii
引言.....	1
一、 项目简介.....	2
（一） 项目特色.....	2
（二） 技术实现思路.....	3
二、 视网膜分割算法.....	3
（一） 相关工作.....	3
（二） 图卷积神经网络.....	3
（三） DE-DCGCN-EE 框架.....	4
（四） 模块方法.....	5
（五） 实验结果.....	5
三、 网页设计.....	5
（一） 前端设计.....	5
（二） 后端设计.....	8
四、 功能展示.....	11
(一)注册、登录页面.....	11
(二)主页面.....	12
（一） 上传图片页面.....	13
（二） 图片编辑页面.....	15
五、 应用前景.....	16
（一） 市场需求.....	16
（二） 推广模式.....	17
结论.....	18
参考文献.....	18

## 图目录

图 1	动态图卷积神经网络结构图.....	4
图 2	Vue.js 运行机制图.....	5
图 3	MVVM 示意图.....	6
图 4	组件化示意图.....	6
图 5	插件引用和重定义代码.....	8
图 6	Django 后端示意图.....	9
图 7	MySQL 数据库示意图.....	10
图 8	数据库关系示意图.....	10
图 9	数据库表示意图.....	11
图 10	注册卡片.....	12
图 11	登录卡片.....	12
图 12	上传图片页面.....	13
图 13	上传图片样例.....	13
图 14	分页操作区.....	15
图 15	图片编辑页面.....	15
图 16	图片标注示例.....	16

## 引言

眼底疾病种类繁多，发病率高，且一旦发病容易致盲，危害严重。同时，眼底也是人体中唯一一处能以直接成像的手段观察血管的部位，经验丰富的医生可通过对视网膜血管观察来诊断眼底以及身体其它部位的疾病。但人工分割视网膜血管存在耗时长、易漏诊、易误诊的现象，延误了某些疾病的治疗时机。因此，采用人工智能辅助进行视网膜血管自动分割具有不可忽视的临床医学意义。

目前人工智能算法发展迅速，且在实际生活中应用较为广泛，但在医学领域尚缺乏实用性强、可靠性高的临床医用系统。中国的医学仍处于发展阶段，现有的临床诊断过程存在耗时长、效果差、形式单一等问题。目前中国只拥有 4 万余名眼科医生，而能熟练诊断眼底疾病的医生只有大概一千位，供不应求。基于人工智能算法的视网膜分割技术，采用监督学习的策略，以专家级眼科医生标注的数据集为训练样本，通过模型优化在测试集中能获得较高的准确率，使眼底疾病诊断技术触达基层成为可能。

普通计算机由于硬件性能的限制，无法运行深度学习算法，且医院出于成本方面的考虑，一般不会出资租赁高性能的计算服务器，这也间接导致了人工智能算法在医疗领域的落地困难。

为缓解国内医疗资源紧张、眼科疾病看病难的问题，笔者致力于开发一款集准确性、快速性、智能性于一体的医用视网膜分割辅助诊断系统。为保障功能的完善和使用的流畅，本项目与北京友谊医院、北京同仁医院和北京儿童医院眼科等单位展开合作，在实际考察与调研中进行辅助诊断功能的迭代开发，旨在为视网膜图像的血管形态分析提供智能化工具，减轻眼科医师的工作压力，并为眼底疾病致病机理的临床研究提供理论依据。

## 一、项目简介

### （一）项目特色

#### 1. 高性能计算服务器支持大规模医疗设备集群运行

基于双编码器的动态图神经网络的视网膜分割算法具有处于领先地位的性能，但仍对处理器的性能有较高的要求。试验和调研表明，普通医用办公电脑无法支持深度学习相关算法的运行，且一般医院没有配备高性能服务器，导致该方法不能在临床上获得广泛应用。

本项目所依托的校内实验室服务器具备运行该算法的能力，且本校提供了可用于校外远程访问的 VPN 服务器。本项目将眼底视网膜分割平台的网页前端和后端，以及视网膜分割算法部署在校内实验室服务器上，能够支持多个医院不同医生的远程访问。通过少量的计算资源，以低廉的成本实现了大规模、高效率和高精度的辅助诊断。

#### 2. 诊、疗、研一体化功能

视网膜血管错综复杂，毛细血管肉眼分辨较为困难，造成眼底疾病诊断任务重，误诊率高。本项目中所采用的视网膜分割算法能实现眼底视网膜图像端到端的分割，并提供给医生准确程度较高的分割结果，有利于医生在早期发现眼底疾病以及身体其它部位的病变，以及及时采取治疗措施。

为追踪患者的治疗过程，及时更正用药方案等治疗措施，本项目为视网膜分割云平台部署数据库，并记录医患的对应关系、医生的治疗方案和患者的病情发展等信息。可通过云平台上的患者档案或图片列表中的搜索栏对信息进行检索和查询。

对于眼科疾病中的疑难杂症，医生可通过上传人工视网膜血管分割的黄金标准，和已有的算法分割结果进行比对，并通过在线的图像编辑器对病变区域等进行标注和强调。该方法的应用，不但能为医生提供眼底疾病致病机理分析、治疗效果评估的理论依据，还有利于与医院开展合作的实验室人员在后台收集临床黄金标准，用于更新模型训练中所采用的数据集，为算法优化提供一手资料。

## （二）技术实现思路

首先，在实验室服务器上进行视网膜分割模型的训练。下载 DRIVE，STARE，CHASE\_DB1，HRF 和 IOSTAR 五个公开眼底视网膜图像分割数据集，采用基于双编码器的动态图神经网络算法在实验室服务器上训练，并保存模型参数和测试程序。

其次，在本地搭建网页的前后端及数据库。网页前端开发采用 Vue 框架，使用 Element-UI 组件库搭建页面，使用 tui-image-editor 图片编辑组件开发图像在线编辑功能；网页后端开发采用 Django 框架，以及 MySQL 搭建数据库。

再者，在校园网环境下，将网页前后端与视网膜分割算法的测试程序进行联调。在网页后端使用 subprocess.Popen 模块创建子进程，以执行视网膜分割算法。

最后，研究校外通过 VPN 访问校内服务器的方式，在非校园网环境下进行测试，并对系统中出现的 bug 进行修复，根据眼科医师的需求进行功能的完善和迭代。

## 二、视网膜分割算法

### （一）相关工作

眼底视网膜血管的形状、尺度、分叉、分支角度、延伸程度等结构特征可以通过仪器直接观察，医生通过观测血管形态以及渗出物、出血点等主要病变特征来判断病变程度，因此采用计算机辅助进行眼底视网膜的分割一直是重要的研究方向。

U-Net 作为医学图像处理领域最具代表性的框架，相比于无监督学习提高了眼底视网膜分割的精度，但仍然难以处理血管复杂的形态结构。

为解决连续卷积和池化操作造成的空间信息丢失，CE-Net 通过密集卷积块和多核残差池化块，捕获了图像中的高级语义信息及上下文信息。

为解决血管曲线结构连续性的问题，多项式拟合、Sobel 及 Canny 等无监督边缘检测算子也被引入到深度学习框架中，提高了分割结果中感兴趣区域的边缘强度。

### （二）图卷积神经网络

由于传统的卷积神经网络 CNN 依赖于方形结构的卷积核，其对于图的处理能力

有限. 不同于 CNN, 图神经网络 GCN 可以整合节点之间的拓扑关联并融合相关特征. 基于血管所具有的类似于图结构的树状形态, GCN 和 CNN 的结合有效地捕捉了血管像素之间的连接关系.

### (三) DE-DCGCN-EE 框架

本项目采用基于图卷积神经网络的 DE-DCGCN-EE 框架, 实现了眼底视网膜血管端到端的分割. 其框架结构图如图 x 所示.

采用先验知识获取大小为  $512 \times 512$  的眼底视网膜图像的边缘特征图. 对原始图像和边缘特征图分别进行编码, 获取大小为  $256 \times 64 \times 64$  的高级特征. 在对边缘特征图进行编码的过程中, 将其引入边缘增强模块进行处理, 为解码器提供补充信息.

将双编码器获得的结果进行像素级别的相加, 采用动态图卷积神经网络捕获其通道内的拓扑关联、聚合通道间的有效特征.

采用双解码器对经过图卷积的特征图进行解码, 得到大小为  $512 \times 512$  的分割结果. 解码器所解码的特征来自边缘增强模块处理后的结果, 以及动态图卷积神经网络处理后的结果.

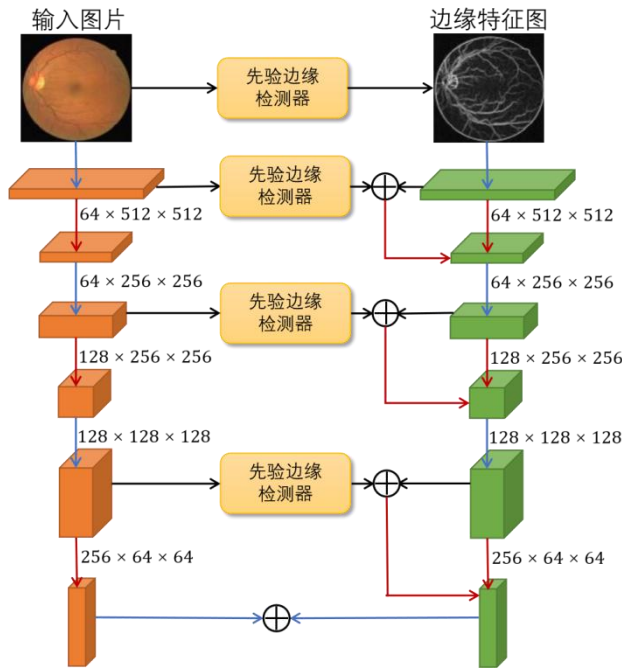


图 1 动态图卷积神经网络结构图



#### （四）模块方法

#### （五）实验结果

### 三、网页设计

#### （一）前端设计

##### 1. Vue 架构

前端使用了类 MVVM 模式的 Vue.js 架构。

视图层内部可以分为三个层次：模型层，视图层和视图模型层。模型层存储数据及对数据的处理，如增删改查；视图层实现 UI 用户界面，视图模型层实现模型层和视图层的数据绑定。

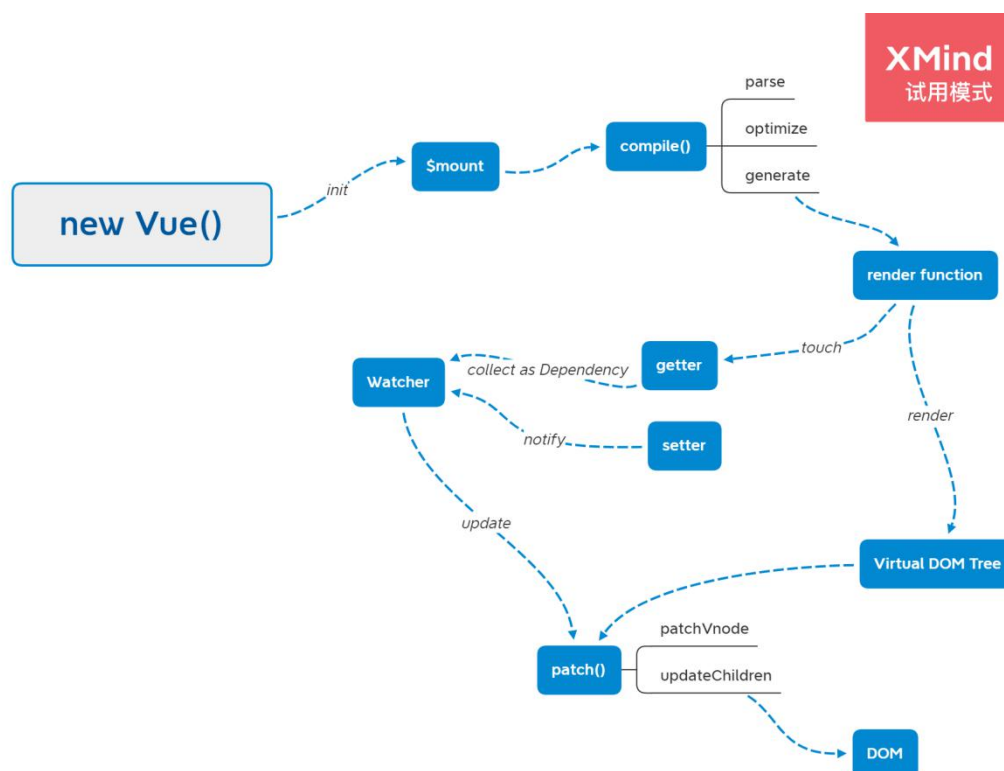


图 2 Vue.js 运行机制图

Vue.js 本身提供了类 MVVM 的数据管理方式，以创建的 Vue 实例作为视图模型层，负责联系模型层的数据与视图层。

模型层采用模块化方式打包，内部使用 Axios 与后端进行数据传输，仅向外部暴露一个名为 Request 的请求方法.封装的模型层与视图模型层进一步解耦，可以随着后端的改动而随时升级.

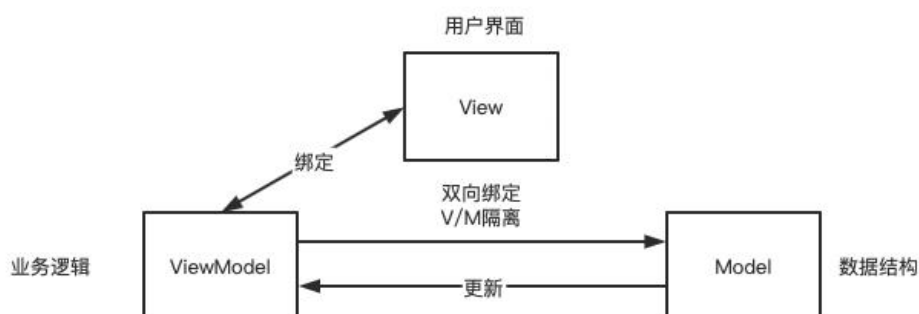


图 3 MVVM 示意图

视图模型层为每一个 data 中的数据建立了内置的 getter 和 setter 方法，并在创建时自动构建了每个对象依赖的数据.通过监听依赖的数据，视图模型层实现了对数据的响应式更新.

视图层由 HTML5、JavaScript、CSS 组成，用于生成用户界面.

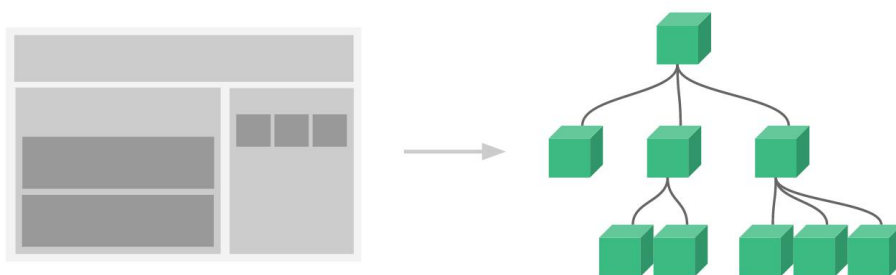


图 4 组件化示意图

在构建应用过程中，我们选择使用组件化开发模式. 抽离包括按钮、列表、下拉栏等组件作为全局组件. 全局组件的开发应遵循结构清晰完整，数据高自由度的标准，并应设置自定义事件实现通信.

同时在开发具体页面时，也应抽离子组件来实现代码复用.每个组件之间进行样式隔离，只在局部进行注册，最大限度实现代码复用和页面解耦.

父子组件的通信使用自定义事件实现，对于其他无关联的组件，抽离其共同状态

放入 Vuex 全局状态管理系统.避免组件之间由于通信产生的耦合度过高现象.

## 2. Element-UI 组件库

Element-UI 是一个基于 Vue 2.0 的桌面端组件库,是本项目大部分插件的引用来源.对该组件库的引用,有利于快速搭建和修饰网站前端.

## 3. Tui-image-editor 插件

Tui-image-editor 插件是一个基于 html5 的图像编辑组件,可以完成图像的放大、缩小、裁剪、旋转、标注等基本的图像编辑操作.本项目引入该插件辅助进行图片标注功能的开发,对插件的属性进行重新定义,在原界面的基础上修改了布局.通过调研了解医生的使用需求,保留图像的放大、缩小、裁剪、旋转、涂鸦标注、形状标注、图标标注、文本框和滤镜等功能,在简化页面的同时满足医生对图片编辑和标注的需求.该插件引用和重定义的部分代码如图 x 所示.

```
import "tui-image-editor/dist/tui-image-editor.css";
import "tui-color-picker/dist/tui-color-picker.css";
this.instance = new ImageEditor(
  document.querySelector("#tui-image-editor"+this.singleImage.photo_id+this.index),
  {
    includeUI: {
      loadImage: {
        path: this.src,
        name: "SampleImage",
      },
      menu: [
        "resize",
        "crop",
        "rotate",
        "draw",
        "shape",
        "icon",
        "text",
        "filter",
      ],
      initMenu: "filter",
      uiSize: {
        width: "1000px",
        height: "700px",
      },
      menuBarPosition: "bottom",
      theme: customTheme,
      locale: locale_zh,
    },
    cssMaxWidth: 700,
    cssMaxHeight: 500,
    selectionStyle: {
      cornerSize: 20,
      rotatingPointOffset: 70,
    },
  },
);
```

图 5 插件引用和重定义代码

## （二）后端设计

### 1. Django 框架

本着简洁度高、可读性强的原则，采用 Django 框架、MySQL 数据库进行网页后端设计。Django 框架作为使用 Python 语言编写的开源 Web 应用框架，具有相对完备的在线文档及相对完善的开发社区，可以在开发过程中参考资料，以提高开发效率。

Django 后端主要完成数据的增、删、改、查等数据库操作，前后端分离的网页设计使得网页具有更好的可扩展性。利用 Django 后端的 `make_password` 和 `check_password` 函数，还可以对数据库中存储的用户密码进行加密，以加强用户隐私的安全性。

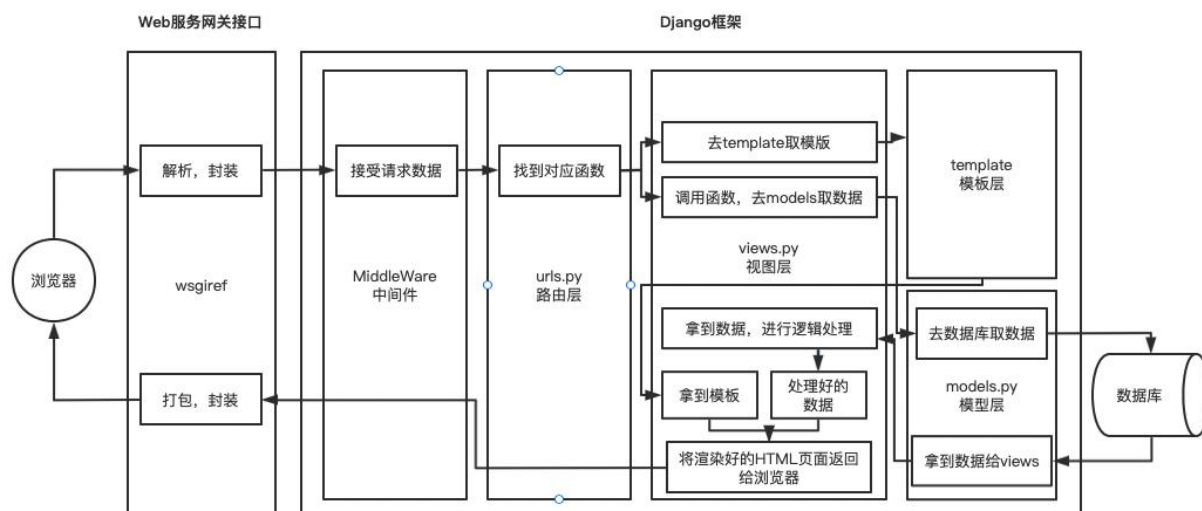


图 6 Django 后端示意图

## 2. MySQL 数据库

### (1) 数据库结构

MySQL 是当下最流行的关系型数据库管理系统, 在 WEB 应用方面是最好的关系数据库管理系统之一. MySQL 使用标准的 SQL 数据语言形式, 可以运行于多个系统上, 并且支持多种语言. 考虑到提升项目可移植性, 我们采用 MySQL 进行网页架构. 为加快项目开发进度, 减少项目人力成本, 本项目使用 Navicat 对 MySQL 数据库中的表进行可视化管理, 使得数据库的设计和管理更加方便, 更具可视化.

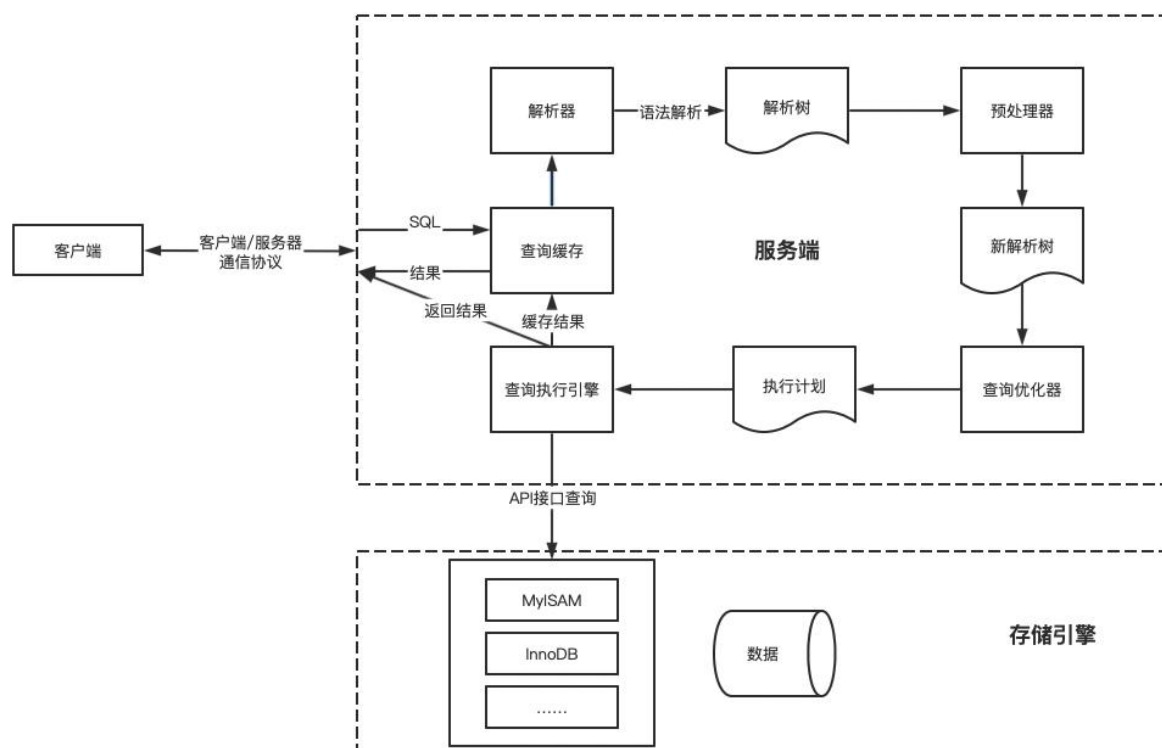


图 7 MySQL 数据库示意图

## (2) 数据库关系

考虑到数据开发成本，本平台使用 MySQL 数据库部署在服务器上。本平台使用者只有医务人员一种角色，因此数据交互只涉及医务人员和服务器两方。

本项目数据库结构涉及三张表，分别为记录医生信息的 `user_doctor` 表，记录上传图片信息的 `user_photo` 表和记录患者信息的 `user_patient` 表。数据库关系示意图如图 x 所示。



图 8 数据库关系示意图

user\_patient 表中，主键设置为患者编号(patient\_ID)，另一字段设置为患者姓名(patient\_name)，剩余字段为自选属性，可以展现更多患者信息.user\_doctor 表中，主键设置为医生编号(doctor\_ID)，另一字段为密码(doctor\_password)，医生登录时需要对密码进行验证，剩余字段为自选属性，可以展现更多医生信息.user\_photo 表中，主键为图片(组)编号(photo\_ID)，外键为医生编号(doctor\_ID)，表示该图片所属医生，另一个外键为患者编号(patient\_ID)，表示该图片对应的患者.处理前图片(photo\_origin)和处理后图片(photo\_promap)分别表示处理前后的照片，上传图片(photo\_upload)表示医生上传的黄金标准，以 Image 形式存储.

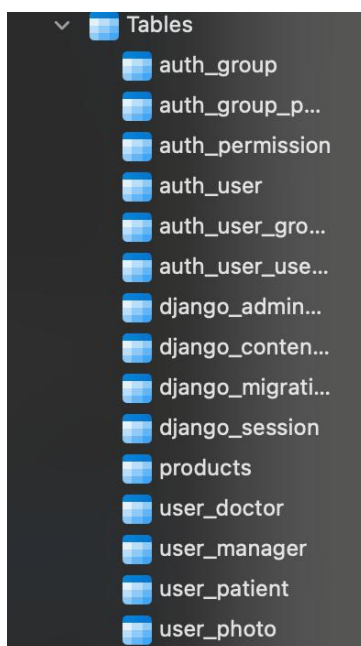


图 9 数据库表示意图

## 四、功能展示

### (一)注册、登录页面

#### 1. 注册卡片

如图 x 所示，输入用户名、真实姓名和密码等个人信息进行注册，其中用户名、密码、真实姓名为必填项，邮箱、手机号、性别信息为选填项。

#### 2. 登录卡片

如图 x 所示，输入用户名和密码进行登录.如还未注册账号，点击下方的“前往注



册”，即可跳转到注册卡片进行注册.注册成功后自动跳转到登录卡片，如果注册失败会有相应的信息提示。

图 10 注册卡片

图 11 登录卡片

## (二)主页面



## 1. 导航栏

可以跳转到不同的子页面，包含个人中心、手术查询、药品信息、最新通知、上传图片界面。

## 2. 登录状态

若处于已登录状态则显示用户名，可以注销登录；未登录状态显示“未登录”。

### （一）上传图片页面

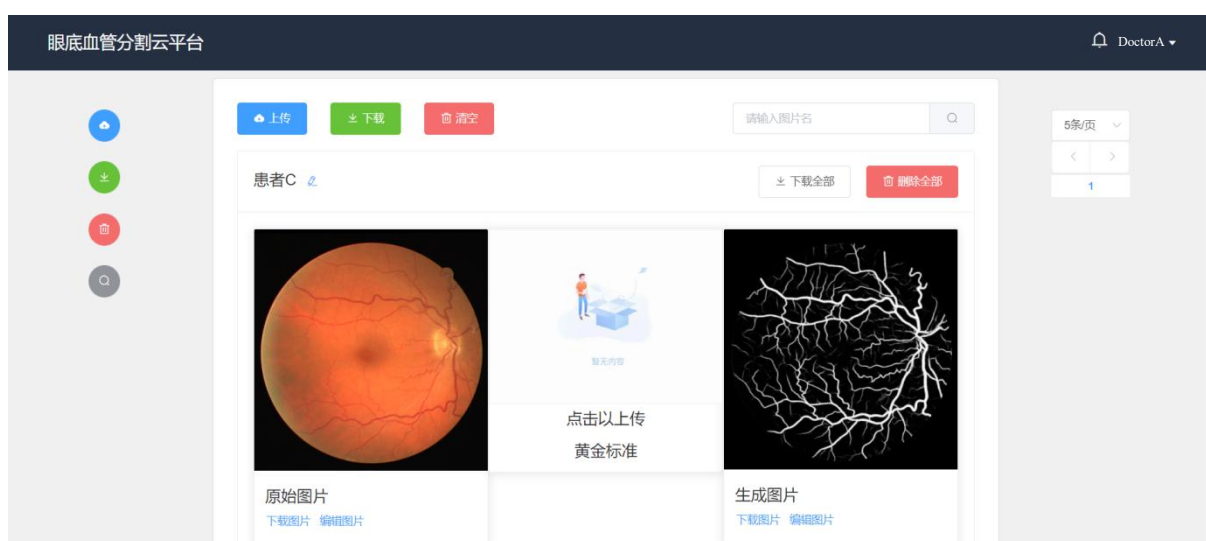
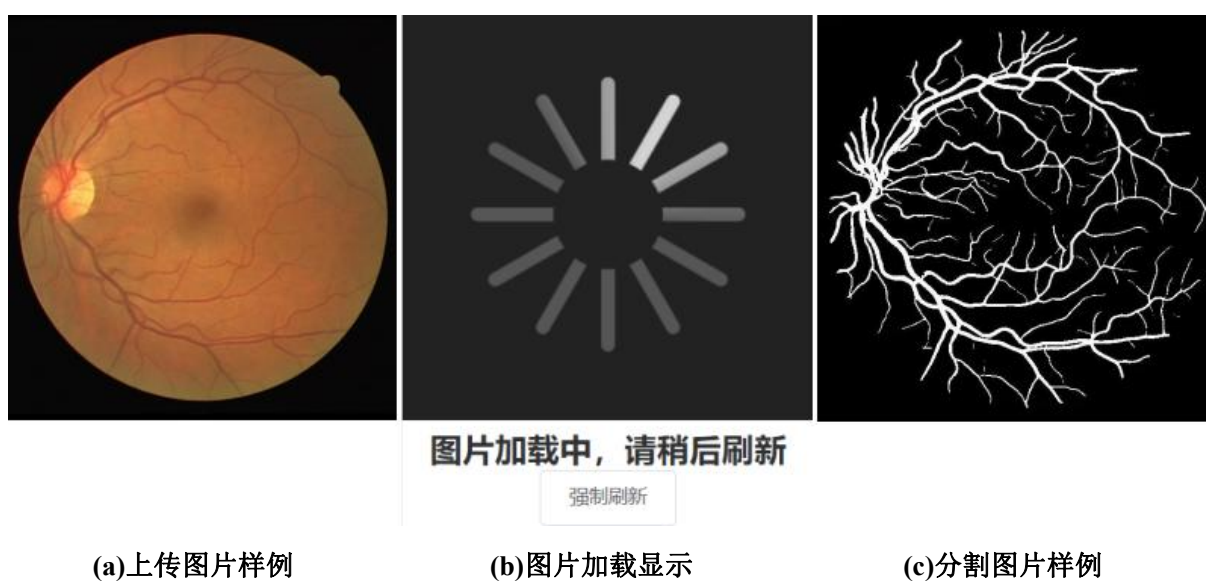


图 12 上传图片页面

### 1. 图片卡片区

位于页面的中部,包含上传、下载、删除、重命名和搜索栏等功能.



(a)上传图片样例

(b)图片加载显示

(c)分割图片样例

图 13 上传图片样例

- **上传**

点击页面最上方的“上传”按钮，用户可选择需要上传的原始图片，选择完毕后点击“确认上传”，系统提示上传结果，上传图片样例如图 x(a)所示.上传成功后系统自动运行视网膜分割算法，显示“图片加载中”，可点击“强制刷新”按钮，如图 x(b)所示.图像分割完毕后在 5s 内自动显示在页面上，生成图片样例如图 x(c)所示.

- **下载**

点击页面最上方的“下载”按钮，用户可以下载本页中的所有生成图片.点击原始图片、黄金标准、生成图片下方的“下载图片”可以下载对应的图片，或点击图片名称右侧的“下载全部”可以下载单条图片卡片的全部图片.

- **删除**

点击页面最上方的“清空”按钮，用户可以清空本页中的所有图片卡片.点击图片名称右侧的“删除全部”，可以删除本条图片卡片.

- **重命名**

点击图片名称旁的编辑按钮，可以修改该条图片卡片的检索名称.

- **搜索栏**

在搜索栏中可以按图片名称、患者信息等检索与其相关的图片卡片.

## **2. 按钮操作区**

位于页面的左侧.按钮操作区的功能与图片卡片区有重叠，用户滚动中间的主操作区到页面下端时，可以通过点按快速操作区的按钮来实现上传图片、下载本页全部图片、删除本页全部图片、按照图片名字搜索图片的功能.

## **3. 分页操作区**

位于页面右侧.

- **调整按钮**

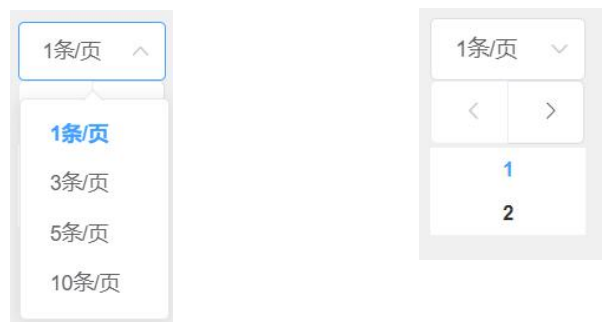
用户点按页面大小调整按钮，系统页面显示的图像卡片数目改变.

- **翻页按钮**

用户点按翻页按钮，系统页面向前或向后翻动.

### • 跳转按钮

用户点按跳页按钮，系统跳转到对应的页面。



(a) 调整每页条数

(b) 页面翻动和跳转

图 14 分页操作区

## (二) 图片编辑页面

在线图片编辑器提供了对图片进行多种操作的功能，包括放大、缩小，长和宽的自由缩放或等比例缩放，图片的裁剪、旋转，自由标注、图形标注和文字标注，灰度、锐化和滤镜等。

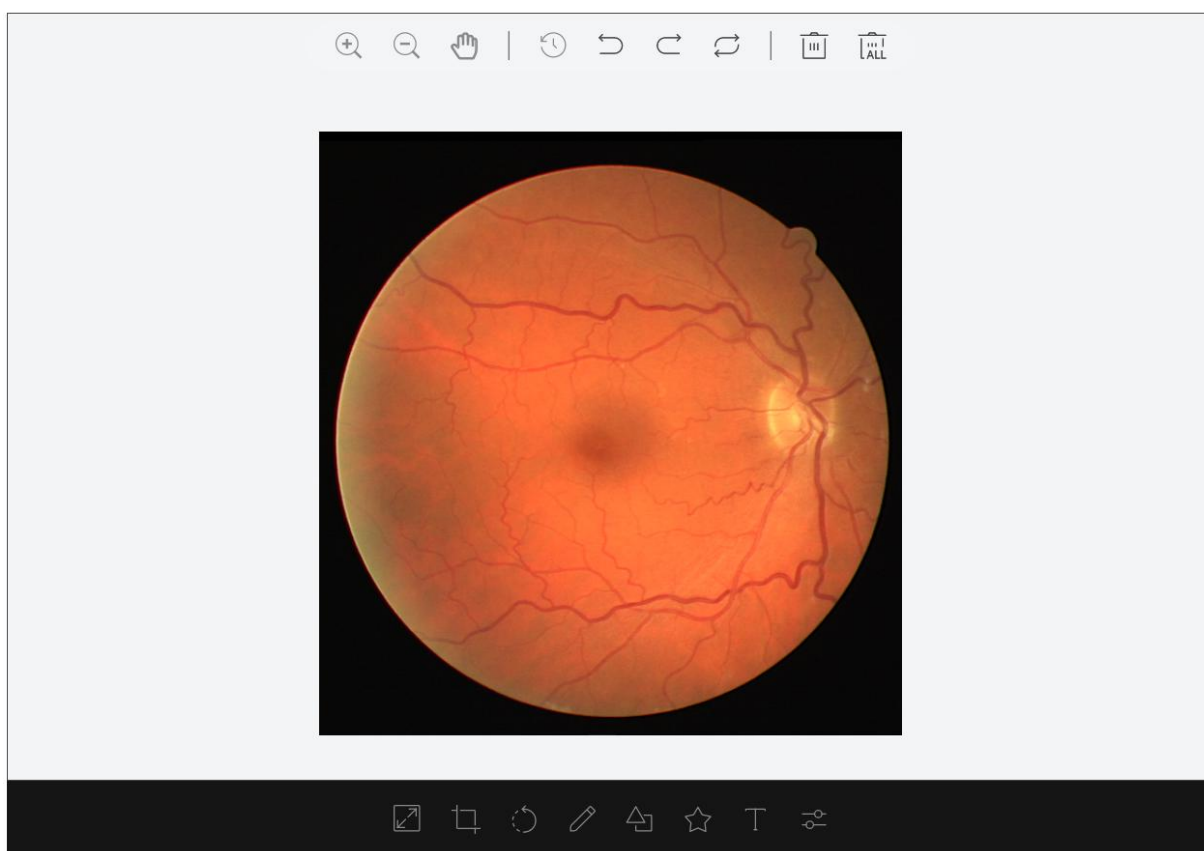


图 15 图片编辑页面

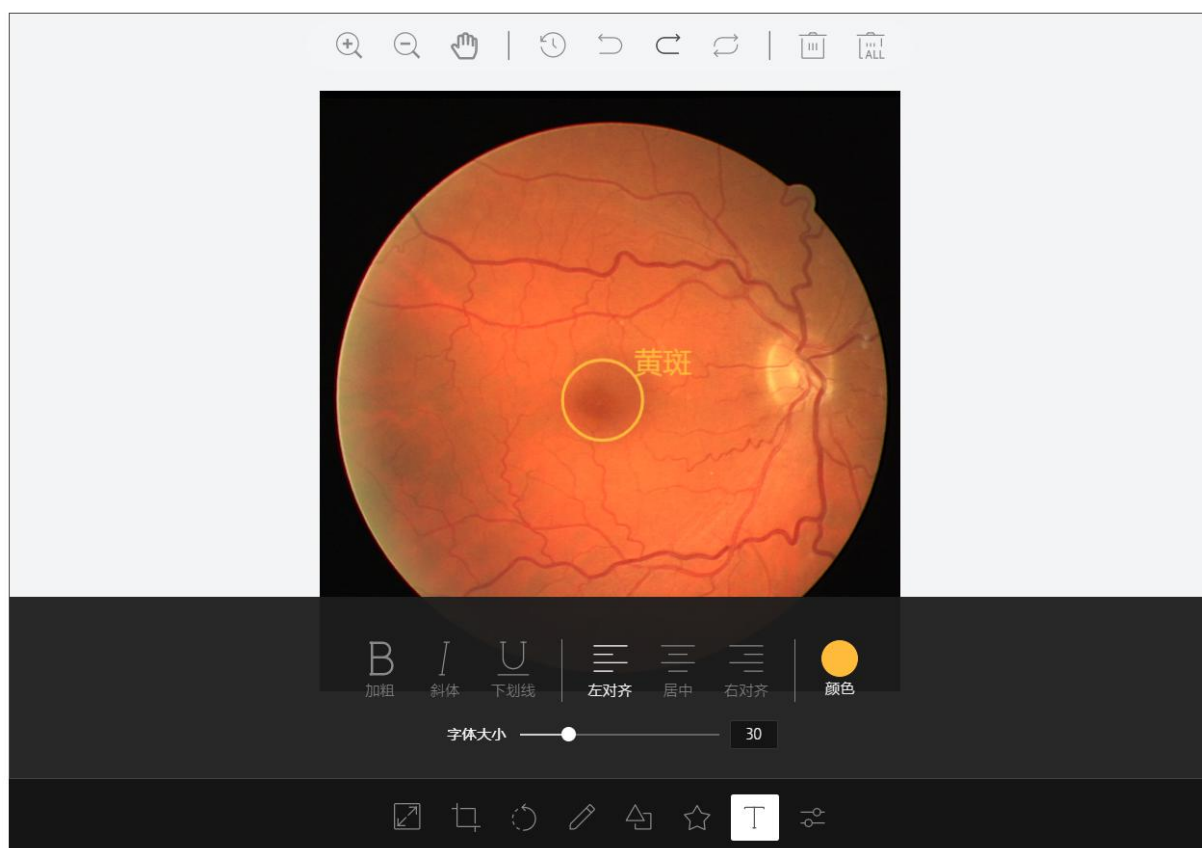


图 16 图片标注示例

在线图片编辑器的主页面如图x所示，使用图片标注的功能在原始图片上标注病灶的示例如图x所示.图片标注功能有利于医生对分割前后的图片和黄金标准进行比对，丰富了图片所承载的有效信息.

## 五、应用前景

### （一）市场需求

目前国内能熟练诊断眼底疾病的医生只有大概一千位，很多患者的疾病无法在早期被确诊，严重延误了最佳的治疗时期.缓解国内医疗资源紧张、“看病难”的问题迫在眉睫.

眼底视网膜血管分割辅助诊断系统充分利用深度学习算法在医学图像处理领域所取得的成果，降低对硬件设备的要求，节约了成本，突破了时间和空间的限制，提高了效率.

预计系统开发的技术手段的推广不仅能推进医疗信息数字化，还可以应用于其他依

赖大规模计算的领域，推动了以机器学习为代表的人工智能算法在医疗领域的落地。

## **（二）推广模式**

拟采用走访、调研各眼科医院的方式，针对医院的实际需求开放该系统一段时间的试运行，并根据医生的反馈为每家医院定制个性化的应用，使得该系统能为规模较小的医院提供技术支持，为规模较大的医院提供数据支持。

也可更加广泛地与医院开展合作，通过研究其他疾病的特性和诊断、研究需求，研制出合适的算法后封装入系统，即可实现该辅助诊断系统在多个医疗领域的广泛应用，并能将不透露隐私信息的数据作为科研机构进一步研究疾病致病机理与医疗手段的临床资料，推动诊、疗、研一体化进程。

## 结论

该辅助诊断医用系统拟采用先进的眼底视网膜血管分割技术，有助于人工智能算法的落地，预计该系统的成功开发能高效辅助诊断眼底疾病，有效缓解国内眼科医生人力短缺，就诊供不应求的现状.充分利用深度学习、网页开发、数据库设计等技术，创新性地提出以简洁度高、可读性强、易于调用、便于维护的 Python 语言为主体的算法架构、与 HTML、CSS 技术相结合，以及 AI 计算服务器支持大规模医疗设备集群运行的理念.将多项技术综合运用于系统的设计，力图增进医生与患者的联系，并为数据集的采集和修正提供可靠的临床样本，推动诊、疗、研一体化进程.本项目作者致力于开发集准确性、快速性、智能性于一体的眼底疾病辅助诊断系统.

## 参考文献

- [1]Quelleg G, Charrière K, Boudi Y, et al. Deep image mining for diabetic retinopathy screening[J]. Medical image analysis, 2017, 39: 178-193.
- [2]Gu Zaiwang et al. CE-Net: Context Encoder Network for 2D Medical Image Segmentation.[J]. IEEE transactions on medical imaging, 2019, 38(10): 2281-2292.
- [3]杨知桥.改进型 U 型网络在视网膜病变检测中的应用研究[A].计算机科学与探索.2021 年.
- [4]W.Richard Stevens, Bill Finner, Andrew M.Rudoff. UNIX Network Programming. POSTS&TELECOM PRESS, 2010.
- [5]Anthony Gore. Full-Stack Vue.js 2 and Laravel 5. Packt Publishing. 2017.