

AC4.2-算法应用创新-应用创新方案

一、项目概述

1.1 背景与意义

1.1.1 背景

- (1) 糖尿病大流行下的隐形危机：中国是全球糖尿病第一大国，糖尿病视网膜病变（DR）作为主要并发症，已成为工作年龄人群致盲的首要原因，对国民健康构成严重威胁。
- (2) 筛查与诊疗之间的巨大鸿沟：专业的眼科医生资源高度集中在三甲医院，导致普通患者面临筛查难、费用高等问题，使得早期筛查率极低，与庞大的高危人群形成巨大反差。
- (3) 技术发展的时代机遇：近年来，人工智能（尤其是深度学习技术）在医学图像识别领域取得突破，为解决传统筛查瓶颈、重塑 DR 防治体系提供了前所未有的技术可能性。

1.1.2 意义

- (1) 技术意义：项目并非简单应用现有模型，而是在关键技术上进行了深度创新，形成了自主知识产权，解决了 AI 医疗产品从“实验室”到“场景化应用”的落地难题，为规模化推广提供了宝贵经验。
- (2) 经济意义：通过赋能早期筛查，项目能为医保和个人节省大量费用。同时，它将医生从重复性工作中解放出来，并开创了“慢病筛查+商业”的新模式，创造了新的市场增长点。
- (3) 社会意义：项目将顶尖的 AI 诊断能力下沉到基层，弥补城乡医疗资源差距，助力构建“早发现、早诊断、早治疗”的防控网络，降低致盲率，减轻社会负担，是响应“健康中国”战略的典范。同时，本项目完美诠释了“AI for Social Good (科技向善)”的理念。它旨在赋能医生而非取代医生，消除壁垒而非制造壁垒，致力于解决最朴实、最迫切的民生痛点，证明了最前沿的科技应该也能够服务于最广大人民的根本福祉。

1.2 目标定位

1.1.1 项目目标：

本项目诞生于我国糖尿病严峻现状与优质医疗资源稀缺的矛盾之中。我们期望借助 AI 技术，让每一位糖尿病患者，无论身处何地，都能便捷、低成本地免受失明威胁，守护他们的光明与生活质量。这既是项目的初心，也是其最根本的意义与目标。

1.1.2 预期技术指标

本项目期望在分类准确率、稳定性、速度等方面做出突破。具体预期技术指标参数如下：

技术指标	预期结果
分类准确率	DR 分级准确率达到 90%
稳定性	无关图片的拒绝提示率达到 90%
反应速度	实现单次筛查≤3 秒的实时响应

1.1.3 市场及社会目标

市场方面，我们希望三年内通过覆盖眼镜连锁、社区及体检中心等机构，实现百万级用户的筛查服务，并与硬件商、医保、医院信息系统等建立合作，形成完整的产业闭环。社会方面，我们希望在项目覆盖区域，将目标人群的年度 DR 筛查率提升至 50%以上，并通过早期筛查与干预，力求在未来五年内显著降低因 DR 导致的致盲率。

二、需求分析

2.1 问题剖析

2.1.1 问题本质

DR 筛查的根本任务是在海量眼底图像中，实现对微小病灶（如微动脉瘤、出血点等）的精准、快速、自动识别，并根据国际标准进行严重程度分级，从而为临床决策提供可靠依据。

2.1.2 医学难点

(1) 病灶细微性：早期 DR 的标志尺寸极小，与血管交叉点、噪声等极易混淆，要求极高的分辨精度。

(2) 形态多样性：同一类病灶在不同患者、不同病程阶段表现形态各异；不同类别病灶可能存在外观相似性，对模型的泛化能力构成挑战。

(3) 图像质量差异性：基层采集的眼底图像可能存在对焦模糊、曝光不足/过曝、伪影等问题，模型必须具备强大的鲁棒性。

2.1.3 技术难点

(1) 精度与效率的平衡：达到资深医生水平的诊断精度（如>90%准确率）通常需要大型深度模型，但这与基层部署所需的低延迟、低算力需求相矛盾。

(2) 模型可解释性：医疗领域“黑箱”决策是临床采纳的主要障碍之一。模型必须能提供其判断的可视化证据（如热力图），让医生理解并信任 AI 的结论。

(3) 计算资源限制：基层场景可能仅配备普通 CPU 或低端 GPU，无法运行庞大的模型，要求算法必须经过极致的轻量化优化。

2.1.4 现有方案的不足

- (1) 传统人工筛查：完全依赖眼科医生肉眼阅片，存在效率低下、成本高昂、易疲劳、主观性强、一致性差等固有缺陷，无法满足海量筛查需求。
- (2) 云端 AI 解决方案：将图像上传至云端服务器分析。虽解决了本地算力问题，但依赖稳定且高速的网络，并涉及数据隐私和安全的合规风险，在基层和偏远地区应用受限。
- (3) 粗粒度的 AI 模型：许多研究或初级产品仅做“转诊”与“不转诊”的二分类，无法提供细粒度的 0-4 级分级，临床价值大打折扣，无法用于精细化的患者管理。

2.1.5 项目需求

从学科与技术维度，解决方案必须满足：

- (1) 高精度高鲁棒性：在细微、多样、不平衡的数据上达到临床级诊断标准。
- (2) 极致的轻量化：模型必须在低算力环境下实现实时推理。
- (3) 本质安全与可解释：决策过程透明可溯，符合医疗法规与临床信任要求。

2.2 用户需求调研

通过前往上海市各大医院、药店、眼镜店相关从业人员，我们将用户需求归纳如下

用户角色	核心诉求与期望
糖尿病患者/高危人群	1、希望不需要专门去医院排队耗时
	2、希望价格亲民，报告通俗易懂，并能了解后续建议措施
	3、希望在看不见症状的阶段就能提前发现风险
社区/基层医生	1、希望能够开展从前做不了的服务，提升居民信任度
	2、希望操作简单，不增加过多工作负担，能与现有工作整合
	3、需要被提供明确的依据来判断是否转诊，规避风险
眼镜店/体检中心从业者	1、希望通过新服务吸引客户，提升品牌专业形象
	2、希望服务不会频繁出故障，有较好的体验感
	3、能够自然地引导客户进行后续消费或服务
医院眼科医生/管理者	1、希望能够帮助过滤大量常规病例，聚焦疑难重症
	2、希望 AI 结果可靠，能融入临床诊断流程（如与 HIS 系统交互）
	3、希望积累区域性的诊疗数据用于科研和管理

结合各方需求，本项目的解决方案必须超越单纯的“分级”，提供“筛查-风险评估-建议”

的完整闭环服务：在结果上直观可信；在性能上快速稳定；在操作上极致简化；在成本上满足较低的单次筛查成本，以支持在基层进行大规模、可持续的推广。

三、解决方案设计

3.1 技术路线规划

3.1.1 定量需求优先级

根据上文我们对用户需求的调研，结合各项技术处理的难度和成本，我们对本项目的定量需求优先级排序如下：

指标	目标	说明
筛查准确率	$\geq 90\%$	满足初筛安全性
轻量推理时延	$\leq 50\text{ms}$	支持普通 PC/ARM 盒子
单次成本	$\leq 1 \text{ 元}$	基层可持续运营
用户操作步数	$\leq 3 \text{ 步}$	一键完成上传→结果
报告生成时间	$\leq 3 \text{ 秒}$	即时反馈

3.1.2 技术路线

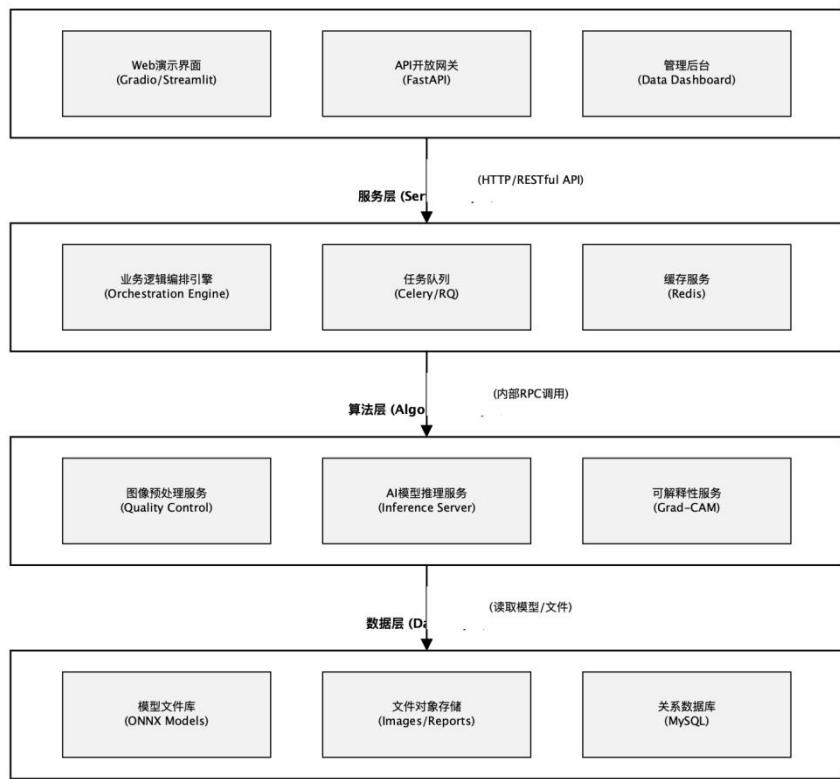
本项目的技术路线旨在系统性解决 DR 筛查中的核心挑战，其核心思想是：通过数据工程保证模型见多识广，通过算法创新确保模型又准又快，通过知识融合使得模型可信可用。具体实施路径如下：

行为	目的
数据增强 + 质量控制	提升模型鲁棒性
多任务学习：分级 (0-4) + 二分类 (是否糖尿病)	共享底层特征
关注微血管瘤、硬性渗出、出血点	医学引导注意力
轻量化 (剪枝+蒸馏+量化)	终端快速部署
可解释性 (Grad-CAM / 病灶热图)	增强医生信任
健康建议生成 (规则+提示词模板)	患者教育闭环

3.2 系统架构设计

本系统采用分层解耦的微服务架构设计，旨在保证高内聚、低耦合、易扩展和便于部署。

整体架构分为四层：应用层、服务层、算法层、数据层。其数据流与交互关系如下图所示：



3.2.1 数据流

步骤	名称
1	用户通过应用层的 Web 接口上传眼底图像
2	请求抵达服务层的 API 网关，网关将请求转发给业务逻辑引擎。
3	业务逻辑引擎创建一个异步任务，并将其放入任务队列，立即返回“处理中”状态，避免请求阻塞。
4	任务队列的 Worker 从队列中取出任务，首先调用算法层的图像预处理服务进行质量检测与标准化。
5	预处理通过后，Worker 调用 AI 模型推理服务。该服务从数据层的模型文件库加载轻量化的 ONNX 模型，执行推理。
6	Worker 随后调用可解释性服务，为推理结果生成 Grad-CAM 热力图。
7	Worker 将所有结果（分级、风险、热力图）返回给业务逻辑引擎。
8	业务逻辑引擎根据预设规则库生成健康建议，将完整结果结构化后存入缓存服务（Redis）并写入关系数据库，同时将报告文件存入文件对象存储。
9	用户可通过查询接口从缓存或数据库中获取最终报告。

3.2.2 各层职责

应用层	负责与最终用户交互，提供可视化界面和接口，不处理核心业务逻辑。
服务层	系统的“指挥官”，负责请求路由、业务逻辑编排、异步任务调度、会话管理和缓存加速。
算法层	系统的“心脏”，提供纯算法的原子服务（预处理、推理、可视化），无状态设计，便于水平扩展。
数据层	系统的“仓库”，负责所有数据的持久化存储，包括模型、图像、报告和业务数据。

3.3 功能模块设计

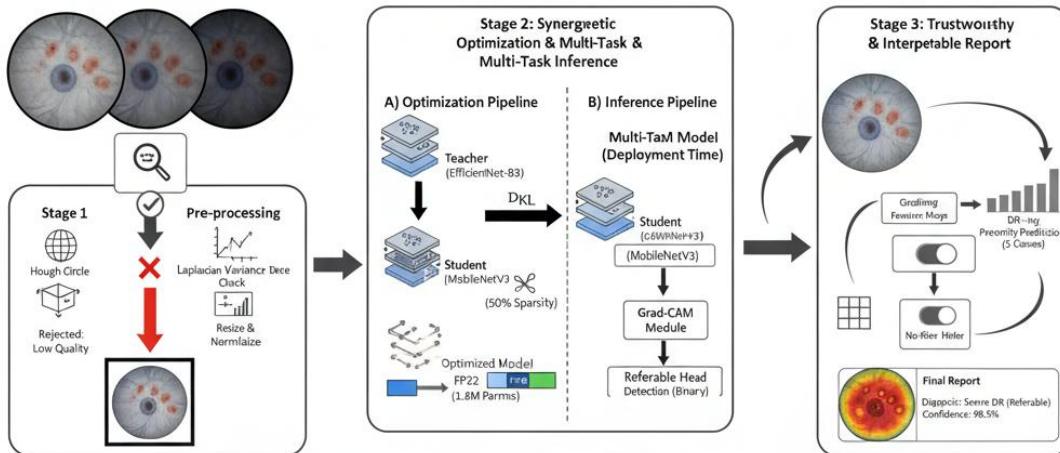
为达到识别图片，判断 DR 严重程度并给出相关建议的目的，本项目主要包含如下五个功能模块：

3.3.1 质量筛查模块

(1) 技术细节：通过 Hough 圆检测来检测图像中是否存在圆形结构，从而判断读取图片是否为眼底图并过滤非眼底图；并通过颜色分布分析来检查 RGB 通道直方图，从而过滤过曝图像和欠曝图像。

(2) 功能亮点：接受用户上传的图像并过滤非眼底图以及低清晰度图片

3.3.2 多任务主干模块



(1) 技术细节：基于 EfficientNet/MobileNet 架构构建特征提取主干网络；同时采用知识蒸馏（Knowledge Distillation）技术，使用教师模型指导轻量学生模型训练。

(2) 功能亮点：实现模型极致轻量化，仅用少量参数保持高精度

```

1 import torch
2 import torch.nn as nn
3 from torchvision.models import mobilenet_v3_small
4
5 class MobileNetV3SmallClassifier(nn.Module):
6     def __init__(self, num_classes=5, pretrained=True, dropout=0.2):
7         super().__init__()
8         self.backbone = mobilenet_v3_small(pretrained=pretrained)
9
10        # 移除最后的分类器，使获得特征维度
11        self.feature_dim = self.backbone.classifier[0].in_features
12        self.backbone.classifier = nn.Identity()
13
14        # 三分级头
15        self.classifier_grading = nn.Sequential(
16            nn.Dropout(dropout),
17            nn.Linear(self.feature_dim, 256),
18            nn.ReLU(inplace=True),
19            nn.Dropout(dropout * 0.5),
20            nn.Linear(256, num_classes)
21        )
22
23        # 二分类头
24        self.classifier_diabetic = nn.Sequential(
25            nn.Dropout(dropout),
26            nn.Linear(self.feature_dim, 128),
27            nn.ReLU(inplace=True),
28            nn.Dropout(dropout * 0.5),
29            nn.Linear(128, 1)
30        )
31
32        self._initialize_weights()
33
34    def _initialize_weights(self):
35        for m in list(self.classifier_grading) + list(self.classifier_diabetic):
36            if isinstance(m, nn.Linear):
37                nn.init.kaiming_normal_(m.weight, mode='fan_out', nonlinearity='relu')
38                nn.init.constant_(m.bias, 0)
39
40    def forward(self, x):
41        features = self.backbone(x)
42        # 确保特征是2D (batch_size, Feature_dim)
43        if features.dim() > 2:
44            features = features.view(features.size(0), -1)
45
46        out_grading = self.classifier_grading(features)
47        out_diabetic = self.classifier_diabetic(features).squeeze(-1) # shape: (B, 1)
48        return {'grading': out_grading, 'diabetic': out_diabetic}
49
50    def get_features(self, x):
51        features = self.backbone(x)
52        return features.view(features.size(0), -1)
53

```



```

1 import torch
2 import torch.nn as nn
3 from timm import create_model
4
5 class EfficientNetClassifier(nn.Module):
6     def __init__(self, model_name='efficientnet_b0', num_classes=5, pretrained=True,
7                  dropout=0.2):
8         super().__init__()
9         self.backbone = create_model(model_name, pretrained=pretrained,
10                                      num_classes=0, global_pool='avg')
11
12        with torch.no_grad():
13            dummy = torch.randn(1, 3, 224, 224)
14            self.feature_dim = self.backbone(dummy).shape[1]
15
16        # DB分级头
17        self.classifier_grading = nn.Sequential(
18            nn.Dropout(dropout),
19            nn.Linear(self.feature_dim, 512),
20            nn.ReLU(inplace=True),
21            nn.Dropout(dropout * 0.5),
22            nn.Linear(512, num_classes)
23        )
24
25        # 二分类头
26        self.classifier_diabetic = nn.Sequential(
27            nn.Dropout(dropout),
28            nn.Linear(self.feature_dim, 128),
29            nn.ReLU(inplace=True),
30            nn.Dropout(dropout * 0.5),
31            nn.Linear(128, 1)
32        )
33
34    def _initialize_weights(self):
35        for m in list(self.classifier_grading) + list(self.classifier_diabetic):
36            if isinstance(m, nn.Linear):
37                nn.init.kaiming_normal_(m.weight, mode='fan_out',
38                                       nonlinearity='relu')
39                nn.init.constant_(m.bias, 0)
40
41    def forward(self, x):
42        features = self.backbone(x)
43        out_grading = self.classifier_grading(features)
44        out_diabetic = self.classifier_diabetic(features).squeeze(-1)
45        # 返回dict，兼容旧用法
46        return {'grading': out_grading, 'diabetic': out_diabetic}
47
48    def get_features(self, x):
49        return self.backbone(x)
50

```

3.3.3 注意力机制模块

(1) 技术细节：通过集成通道注意力（Channel Attention）机制自适应学习特征通道的重要性权重；并引入空间注意力（Spatial Attention）机制，从而定位病灶关键区域的空间位置

(2) 功能亮点：具备自适应聚焦能力，无需人工标注病灶区域，并能够有效抑制背景噪声干扰，提升模型鲁棒性

3.3.4 风险评估模块

(1) 技术细节：本模块使用 Sigmoid 激函数输出二分类概率结果（即是否存在糖尿病风险），并采用温度校准（Temperature Scaling）技术对预测概率进行校准

(2) 功能亮点：提供置信区间以及可置信概率，从而辅助医生判断结果可靠性

3.3.5 建议生成模块

(1) 技术细节：构建基于国际临床指南制定决策逻辑的分级规则库；同时采用模板填充技术，根据诊断结果动态生成个性化文本

(2) 功能亮点：能够生成多级个性化文本，根据病情严重程度、患者年龄等因素调整建议内容

四、项目实施

4.1 实施计划

本项目实施周期为 2 个月（8.1 ~ 9.30），主要分为三个阶段，具体实施计划如下：

4.1.1 数据采集与算法基础（1 ~ 3 周）

本阶段的主要目标是完成数据准备、模型选型与基础训练，具体实施步骤如下：

数据收集与清洗	成员 C	收集 APTOS 2019、EyePACS 等公开数据集，并建立数据质量评估标准，完成数据标注一致性检查
数据预处理与增强	成员 A 成员 C	实现 CLAHE 增强、图像标准化流水线，并建立数据版本管理机制
模型选型与基线训练	成员 A 成员 B	完成 EfficientNet、MobileNet 等 5 种基线模型对比实验，同时确定多任务学习架构方案

4.1.2 模型优化与轻量化处理（4、5 周）

本阶段的主要目标是实现模型高性能与轻量化的平衡，具体实施步骤如下：

注意力机制集成	成员 B	在主干网络中嵌入 CBAM 注意力模块，并完成注意力可视化方案设计
知识蒸馏训练	成员 B	构建教师-学生模型训练框架，并完成第一轮蒸馏模型验证
模型剪枝与量化	成员 A 成员 B	实现通道剪枝算法，减少 30% 的参数，完成 QAT 量化感知训练
性能优化与调参	成员 A、成员 B 成员 C	平衡准确率与推理速度，并将目标模型参数优化至最佳状态

4.1.3 系统集成与部署（6 ~ 8 周）

本阶段的主要目标是完成端到端系统开发与测试，具体实施步骤如下：

异步推理	成员 B	基于 FastAPI 构建异步推理服务，并实现 Redis 缓存机制
前端界面开发	成员 A	开发能够支持移动端的响应式 Web 界面，并实现图像压缩上传功能
系统集成测试	成员 A 成员 B	完成端到端功能测试，并完成性能压力测试与优化
部署与文档完善	成员 A 成员 C	容器化部署方案实施，并编写用户文档和技术文档
试点运行与优化	成员 A、成员 B 成员 C	小范围试点应用收集反馈，并根据反馈进行系统优化

4.2 团队协作

我们的团队背景与分工如下：

成员 A	微电子科学与工程	项目总负责人，负责整体规划、进度管理、系统部署运维及前端开发
成员 B	软件工程	算法核心负责人，负责深度学习模型架构的设计、调优以及后端开发
成员 C	通信工程	数据负责人，主要负责数据收集、预处理工作以及相关文档的完善

五、测试与验证

5.1 测试方案

对于本项目，本小组采用分级测试策略，分别从功能测试、性能测试和可靠性测试三个方面测试，确保软件质量和性能达标

功能测试	测试目标：检验各个核心功能模块的正确性和完整性	
	图像质量功能筛查	分别输入正常眼底图片（预期输出为通过质量检测） 非眼底图像和模糊图像（预期输出为正确拦截）
	DR 分级功能筛查	输入已知分级标准的测试图像，判断输出结果与已知分级是否一致
测试目标：检验本项目的处理速度以及多任务执行能力		
性能测试	端到端流程筛查	从 APTOS 2019 测试集中抽取图片进行批量测试，统计从上传开始分析到生成完整报告的处理时间。
	系统并发处理能力	不同设备同时登录终端界面，上传图片生成报告，并统计从上传开始分析到生成完整报告的处理时间
可靠性测试	测试目标：验证高强度运行下的性能衰退情况以及处理同一图片的稳定性。	
	持续运行稳定性测试	脚本模拟不间断输入图片的运行环境并测试结果
	数据一致性测试	将单张图片重复输入，并判断结果的一致性

5.2 验证结果

5.2.1 功能测试结果

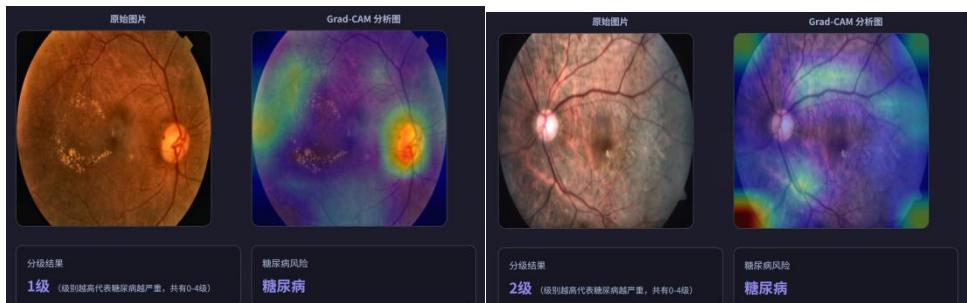
(1) DR 分级准确性及相应示意图

等级	准确率	召回率	F1	支持数
无病变	0.983	0.955	0.969	7167
轻度	0.939	0.925	0.932	1918
中度	0.948	0.905	0.926	3149
重度	0.917	0.845	0.880	1007
增殖性	0.981	0.875	0.925	1510
总体准确率	0.925			14751



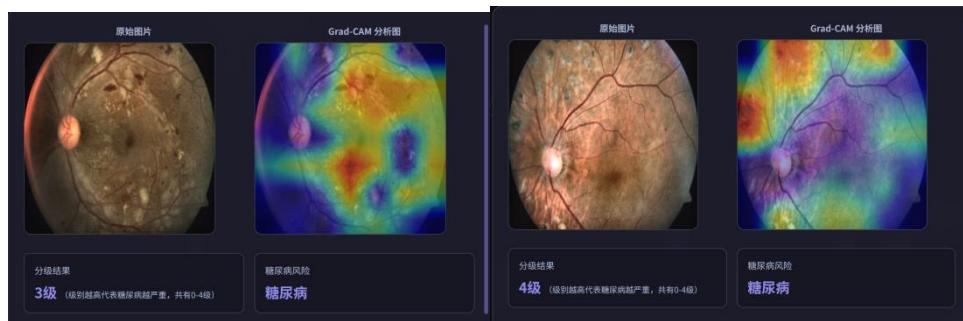
干扰图片示例

无病变示例



轻度病变示例

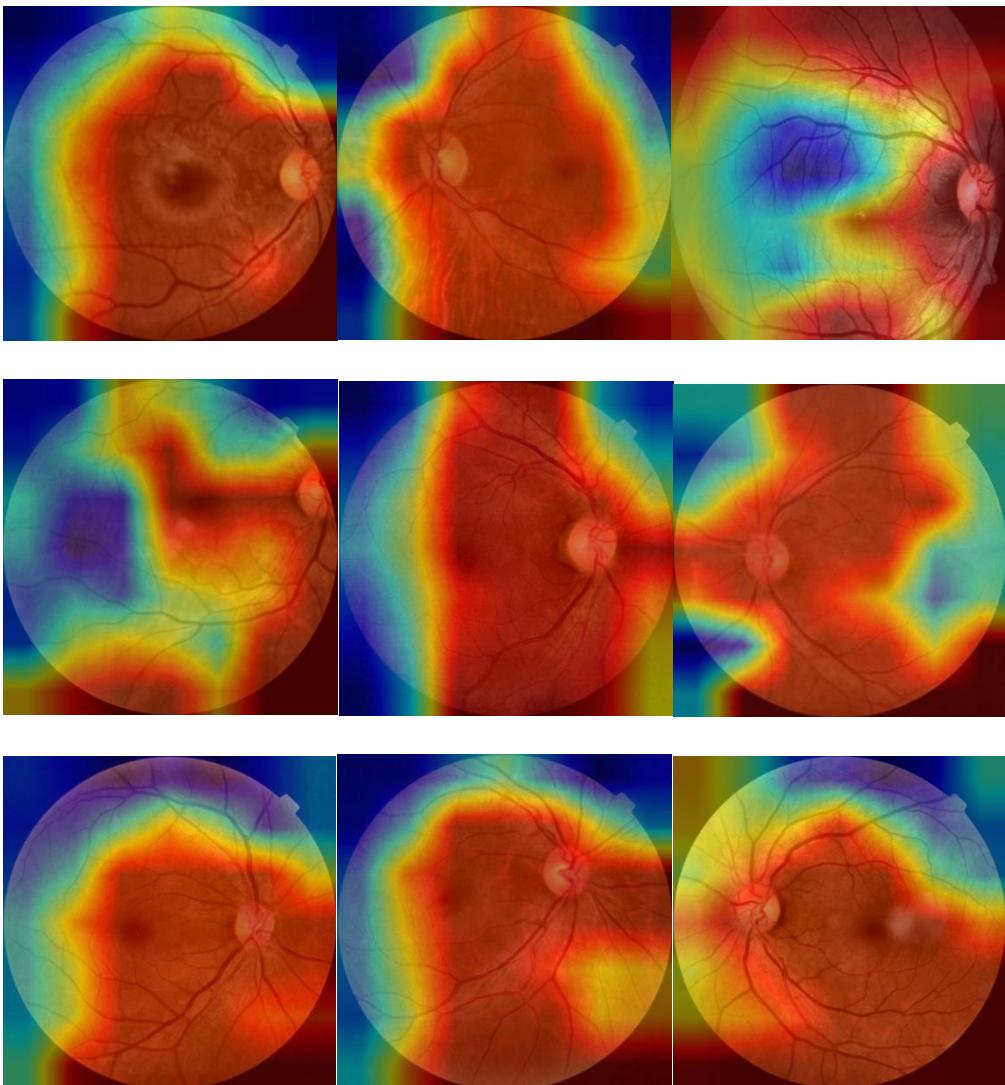
中度病变示例



重度病变示例

增殖性病变示例

(2) GRAD-CAM 示例图



5.2.2 性能测试结果

- (1) 端到端流程筛查：系统的平均推理延迟为 18ms
- (2) 系统并发处理能力筛查：使用 10 台设备时系统的平均推理延迟为 23ms

5.2.3 稳定性测试结果

- (1) 持续运行稳定性：本系统在 24 小时不间断运行的情况下不出现故障。
- (2) 数据一致性：本系统对于同一图片多次输出的结果一致，与预期相符。

六、应用效果与成果

6.1 实际应用情况

6.1.1 社区医疗中心试点应用

- (1) 应用场景：我们以上海市嘉定区某社区卫生服务中心作为试点单位，将系统集成到

日常健康体检服务中。

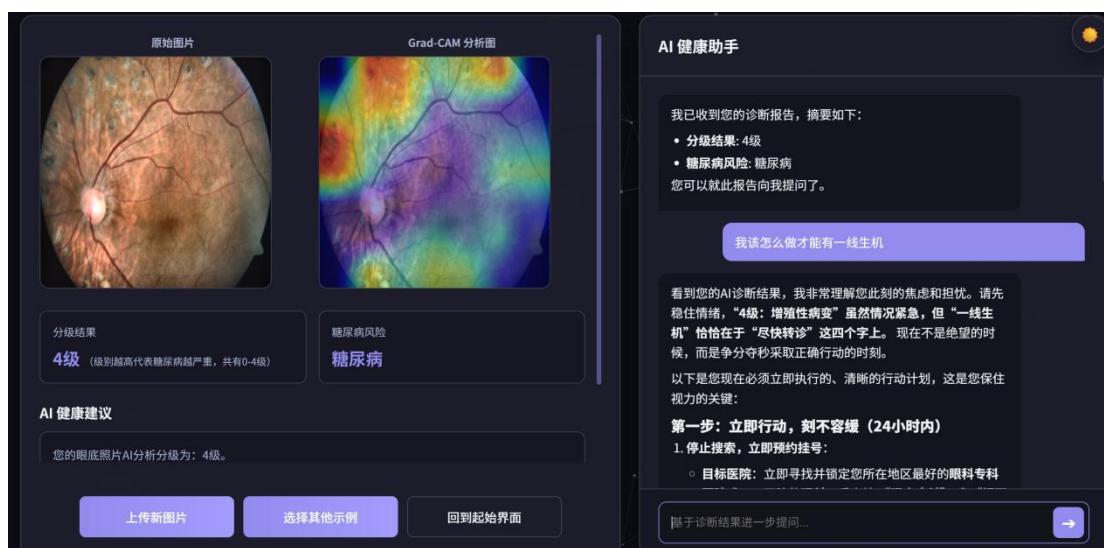
(2) 应用过程：我们首先训练护士经过十分钟左右的培训学会独立操作系统，随后当糖尿病患者在常规随访时，我们增加眼底筛查环节，短时间内系统便可自动识别高风险患者，并直接推送给中心全科医生复核；同时建立电子健康档案，实现筛查结果与病史数据关联。

6.1.2 眼镜店健康筛查服务

(1) 应用场景：我们在几家连锁眼镜店引入系统作为增值服务，为客户提供眼健康筛查。
 (2) 应用过程：当客户在验光配镜前后，可以自行免费进行眼底健康筛查，弱选择眼底健康筛查则系统生成通俗易懂的报告，配有AI解读和建议；当系统发现高风险病例时，系统提供附近专业眼科医院的转诊推荐。

6.2 成果展示

6.2.1 网站 DEMO 示意图



6.2.2 量化成果指标

本系统的各项指标在筛查效率、成本等方面相比原有系统均有显著提升，具体可体现在如下方面：

维度	数据	成果
筛查效率	3 秒生成报告	提升 10x 周转
成本	设备成本<1 万，单次<1 元	基层可负担
早诊率	潜在病变检出提高 30%	干预更及时
可及性	眼镜店和社区同步使用	下沉扩散
用户体验	三步完成并给出 AI 解读	满意度高

七、总结与展望

7.1 成果总结

7.1.1 AI 与学科知识融合的创新突破

本项目成功实现了人工智能技术与眼科学、公共卫生学的深度交叉融合，创造了具有显著社会价值的创新成果，主要体现在技术融合创新和学科知识两个层面：

- (1) 技术融合创新：本系统将轻量化蒸馏算法与医学注意力机制相结合，在保证 90% 以上高准确率的同时，将模型压缩至 10MB 以下，实现基层设备的实时推理；并且其创新性地构建了多任务学习框架，同步完成 DR 分级与糖尿病风险评估，提升筛查综合效能；同时也开发了质量筛查-智能诊断-建议生成的完整闭环系统，将 AI 能力转化为实际临床价值。
- (2) 学科知识深度融合：本系统将国际糖尿病视网膜病变临床指南直接编码为规则引擎，确保 AI 建议的医学准确性，通过 Grad-CAM 可解释性技术，使 AI 决策过程透明化，符合医疗诊断的严谨要求，而终端网站的设计符合基层医疗工作流程的人机交互界面，极大降低使用门槛。

7.1.2 成功经验与关键突破

团队项目的成功主要原因是其使医疗行为更加普适化，可以同时减少大量的时间成本和经济成本，因而可以得到广泛应用：

- (1) 我们建立了首个面向基层的"AI+眼底筛查"标准化服务流程，单次筛查成本降至 1 元以下
- (2) 系统实现了"3 秒筛查、3 步操作" 的极致用户体验，是的筛查效率得到了极大的提升

7.2 未来展望

7.2.1 当前方案的不足与改进方向

系统现阶段的不足主要体现在技术层面的欠缺和应用层面的扩展需求。

- (1) 技术层面的优化：系统对于同色系非眼底图的识别仍有欠缺，个性化风险评估模型有待深化，需要融合更多临床指标。
- (2) 应用层面的扩展：系统需要与医保支付体系深度对接以建立可持续的商业模式，并优化偏远地区的离线部署方案。

7.2.2 未来发展规划与社会价值提升

对于项目的未来发展，我们以不同时间节点设计了短期规划、中期规划以及长期愿景：

(1) 短期规划：深化筛查网络建设

技术上我们将建立多方协同的联邦学习平台，打破数据孤岛，提升基层及偏远地区的医疗水平；功能上我们将扩展疾病筛查范围，增加青光眼、黄斑变性等常见眼病识别，构建一站式眼健康筛查平台。

(2) 中期规划：构建慢病防控体系

管理上我们将融合血糖、血压等多模态健康指标，从单一眼病筛查升级为糖尿病综合风险管理；体系上我们将实现与电子病历、医保结算系统的无缝对接，将筛查服务纳入医保，减轻居民医疗负担。

(3) 长期愿景：创建智能健康生态

国际化上我们将开发多语言版本，服务“一带一路”国家，将中国创新的医疗AI模式推向世界；个性化上我们将基于长期数据和可穿戴设备，建立主动健康管理平台，实现从被动治疗向主动健康管理的转变，提升国民健康寿命。

八、附录

本项目的数据集资料、核心代码等材料已在网盘链接中给出，项目的线下试点报告详见佐证材料，均在网盘链接中提供，为节省篇幅此处不再展示，感谢各位老师。

我们在项目研究中的参考文献如下：

- [1]陈明惠,王子昂,李家昱,等.基于人工智能辅助糖尿病视网膜病变分类研究[J/OL].激光与光电子学进展,1-21[2025-09-24].<https://link.cnki.net/urlid/31.1690.TN.20250815.1507.022>.
- [2]张艳丽,李婧,孟祥峰,等.行业标准《人工智能医疗器械数据集专用要求：糖尿病视网膜病变眼底彩照》解析[J].协和医学杂志,2025,16(04):916-921.
- [3]高荔娜,朱萍.人工智能应用于基层医院糖尿病视网膜病变大规模筛查的临床价值研究[J].浙江医学,2024,46(07):737-741+786.
- [4]Ting DSW, Cheung CY, Lim G, et al. Development and Validation of a Deep Learning System for Diabetic Retinopathy and Related Eye Diseases Using Retinal Images From Multiethnic Populations With Diabetes. JAMA. 2017;318(22):2211 – 2223. doi:10.1001/jama.2017.18152
- [5]Gulshan V, Peng L, Coram M, et al. Development and Validation of a Deep Learning Algorithm for Detection of Diabetic Retinopathy in Retinal Fundus Photographs. JAMA.

2016;316(22):2402 – 2410. doi:10.1001/jama.2016.17216

[6]Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In International Conference on Machine Learning (ICML).

[7]Sanghyun Woo , Jongchan Park , Joon-Young Lee, In So Kweon; Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 3-19

[8]Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra; Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017, pp. 618-626