基于 RK3588 的智能桌面与实时用户状态检测与提醒功能

摘要

在当今快节奏的生活与工作中，年轻人普遍面临长时间伏案工作或学习的挑战，其工作环境的具体设置（如光线、屏幕距离、坐姿空间）以及个人在专注状态下的无意识习惯（如不良坐姿、用眼过度），都显著影响着他们的工作效率和长期健康水平。为了系统性地解决这一普遍性问题，我们创新性地设计并开发了一套深度集成于桌面设备（如独立智能终端）的先进 AI 辅助系统。

该系统的核心能力在于利用集成摄像头和 **MediaPipe** 计算机视觉框架，实现对用户工作环境和自身状态的**实时、非接触式监测**。它持续采集并分析关键视觉数据：

1. **环境参数：** 精确测量用户眼睛与显示器屏幕之间的距离是否处于健康范围。
2. **用户状态：** 核心功能是**实时姿态检测**。MediaPipe 高效地追踪用户头部、脊椎、肩部、肘部等关键骨骼点的位置，精准识别用户是否出现低头前倾、弯腰驼背、身体过度侧倾、单臂支撑时间过长等典型不良坐姿。

基于上述实时采集并分析的**多维度数据**，系统内置的 AI 算法引擎能够进行即时评估，并通过桌面设备的显示屏、指示灯或连接的音响设备，向用户推送**高度定制化的、具体的改进建议**。例如：“请注意，距离屏幕过近”、“检测高低肩，请放松双肩”“请注意，你有驼背倾向”。这些精准的建议旨在帮助用户即时调整，**优化物理工作空间的人体工学设置，并引导形成更健康的用眼和工作习惯，从而有效提升专注度和作效率，预防相关肌肉骨骼劳。**

选择桌面作为部署核心，正是因为其作为用户工作学习活动的绝对中心区域，位置固定且视角理想，为具备视觉识别能力的 AI 技术提供了最佳的、

无感的集成平台。

我们团队的技术实现依托于强大的 **MediaPipe 框架**（用于高效精准的姿态与环境要素识别）、**稳定的云端数据存储服务**以及**直观易用的微信小程序平台**（用于家长端通知与管理）。这三者协同工作，无缝实现了**姿态检测、环境分析、数据安全存储与关键信息推送**的核心功能闭环。这套桌面集成 AI 系统致力于将传统静态的工作环境转变为能够感知、理解并主动响应用户需求的智能伙伴，显著提升其**智能化和人性化**水平，为用户的健康与高效保驾护航。

第一部分　作品概述

1.1 功能与特性

1.实时姿态监测：通过摄像头捕捉用户姿态，识别 4 种不良姿势；

2.智能预警系统分级预警机制（语音提醒→云端记录→家长通知）；

3.守卫模式：专为未成年人设计，异常数据同步至家长端小程序；

4.语音交互助手：集成 DeepSeek 大模型，支持自然语言交互；

5.云端数据管理：腾讯云 COS 存储异常数据，微信云开发实现数据同步

1.2 应用领域

1.教育场景：学生书桌监测，预防近视和脊柱侧弯；

2.办公环境：职场人士久坐提醒，提高工作效率；

3.居家看护：独居老人/儿童安全保护；

4.康复训练：术后姿势矫正辅助；

1.3 主要技术特点

1.多模态感知：融合视觉分析（MediaPipe）与语音交互（讯飞 API）；

2.边缘-云协同：本地快速处理+云端长期存储；

3.自适应算法：动态调整识别阈值适应不同体型；

4.双模通信：支持 WebSocket 实时通信和 HTTP REST API；

1.4 主要性能指标

类别	参数	备注

处理器	Rockchip RK3588 (8 核)	4× Cortex-A76@2.4GHz + 4× Cortex-A55@1.8GHz
GPU	Mali-G610 MP4	支持 OpenGL ES 3.2 / Vulkan 1.2
NPU	6TOPS AI 算力	支持 INT8/INT16/FP16 推理
内存	8GB/16GB LPDDR4/4X	带宽 51.2GB/s
存储	- eMMC 5.1 (默认 32GB/64GB) - MicroSD 卡槽 (最高 256GB) - M.2 NVMe (PCIe 3.0 x4)	可选配 SSD
USB 接口	USB 3.1/2.0 Type-A/Type-C (支持 OTG)	多设备扩展

1.5 主要创新点

1.姿态-环境联合分析模型：首次将工作环境参数纳入姿势评估体系；

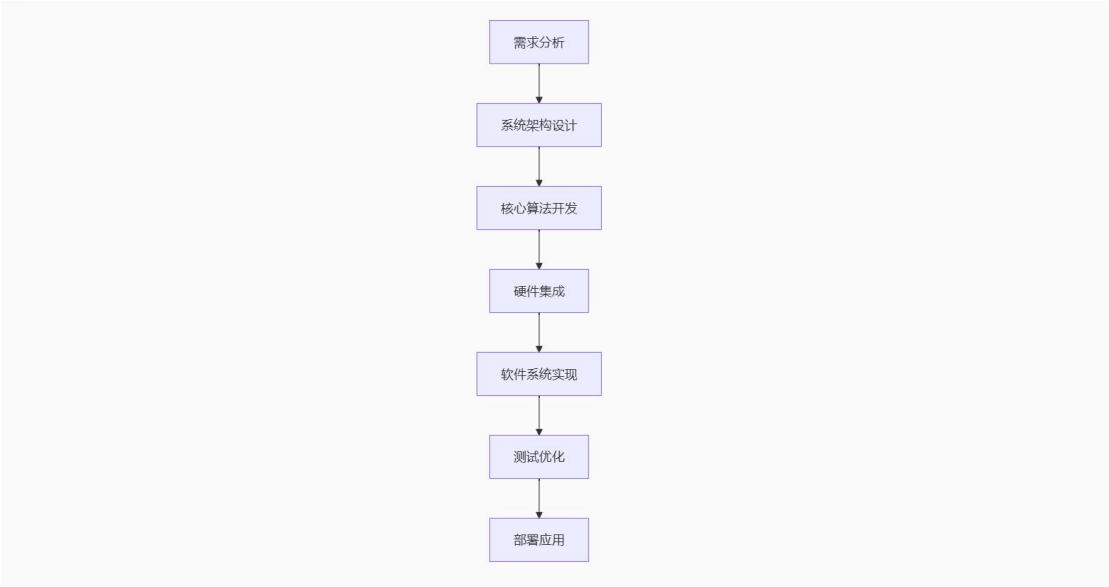
2.守卫模式：创新的家长端实时监护机制；

3.自适应阈值算法：根据用户体型动态调整检测参数；

4.语音-视觉双模交互：切换语音控制和视觉反馈；

5.轻量化部署：可在 elf-board 等嵌入式设备运行；

1.6 设计流程



附图 1

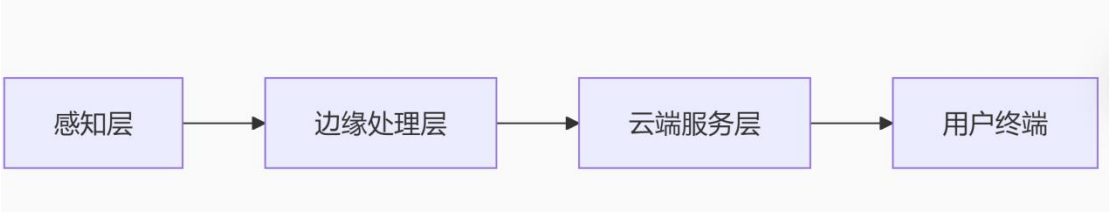
第二部分 系统组成及功能说明

系统采用嵌入式边缘计算架构，以 Elf-board2 开发板为核心构建完整的智能

监测平台。USB 摄像头作为视觉感知单元，直接连接到开发板的 USB 接口，负责实时捕捉用户姿态视频流。双麦克风阵列构成音频感知系统，分别部署在设备两侧，实现立体声采集。Mini 显示屏作为主要的人机交互界面，通过与开发板连接，实时显示坐姿和系统状态。开发板内置 WiFi 模块，建立与云端服务的稳定连接，实现数据同步和远程通知功能。系统采用三层处理流程：感知层通过摄像头和麦克风采集原始数据；边缘处理层在 Elf-board 上运行轻量化 MediaPipe 算法，实时分析用户姿态；应用层通过显示屏提供即时反馈，同时将关键数据上传至云端。这种架构充分发挥了边缘计算的优势，在本地完成核心姿态分析任务，仅将预警事件和统计摘要上传云端，大幅降低网络带宽需求和云端存储成本。

附图

2

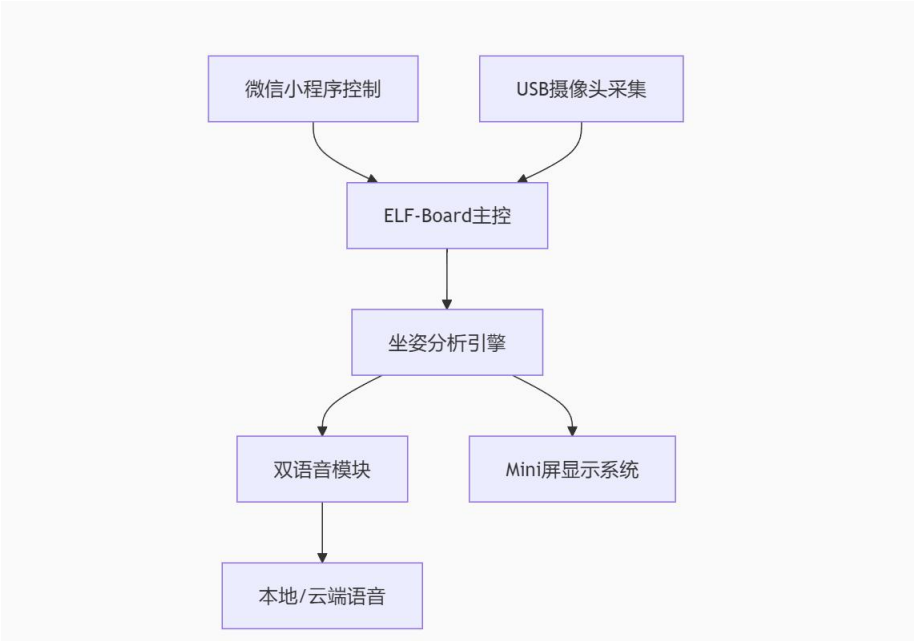


2.1 整体介绍

1.USB 摄像头 → ELF-Board 主控：通过 USB 接口传输原始视频数据；

2.ELF-Board 主控 → 坐姿分析引擎：主控将视频帧送入分析引擎处理；

3. 坐姿分析引擎 → 双语音模块/Mini 屏：分析结果分发给输出设备

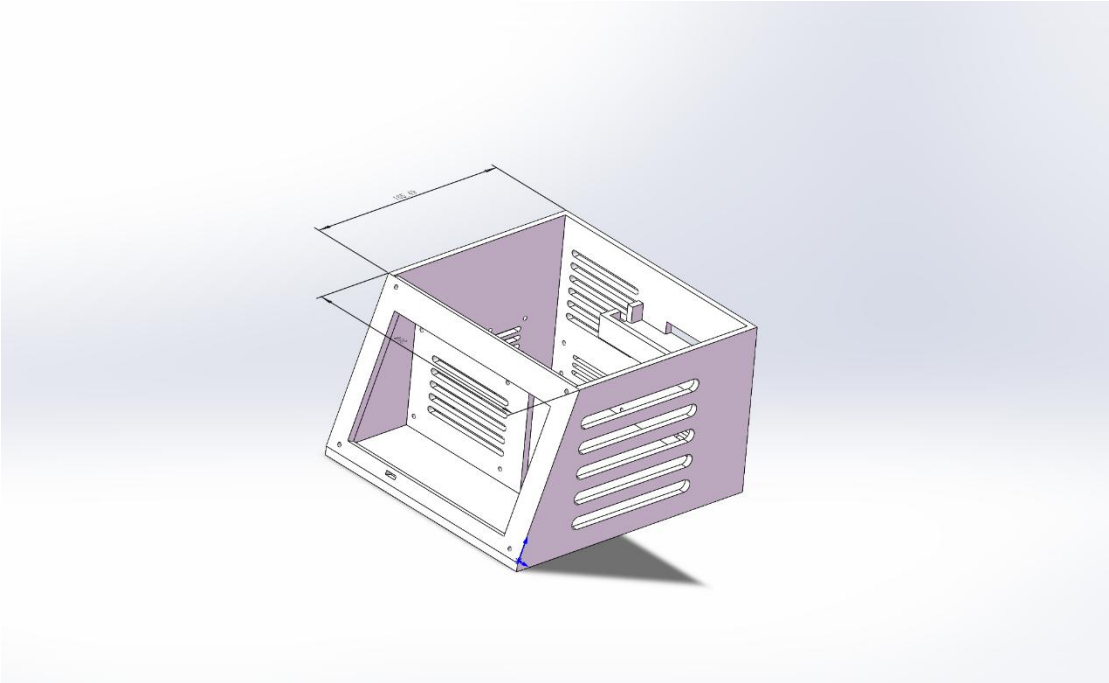


附图 3

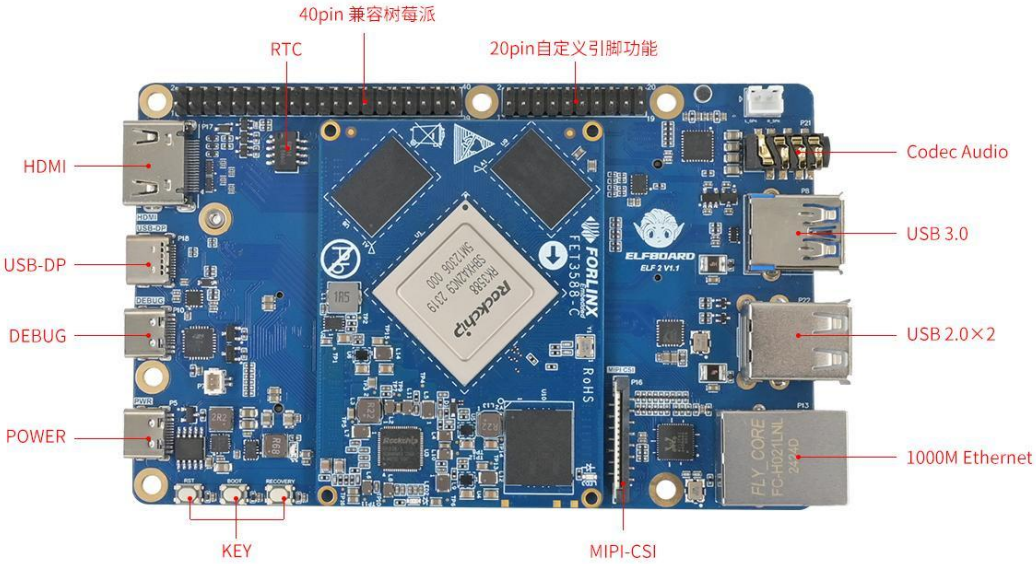
2.2 硬件系统介绍

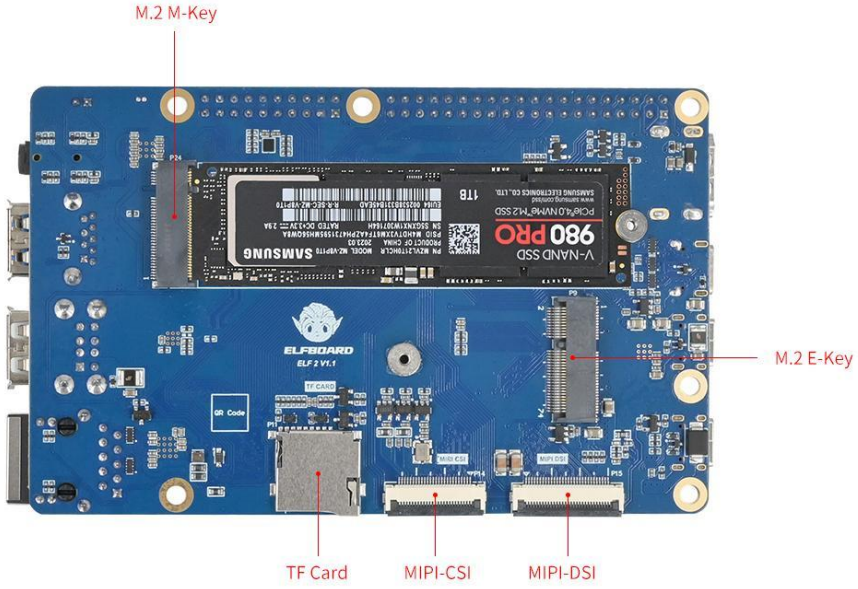
2.2.1 硬件整体介绍

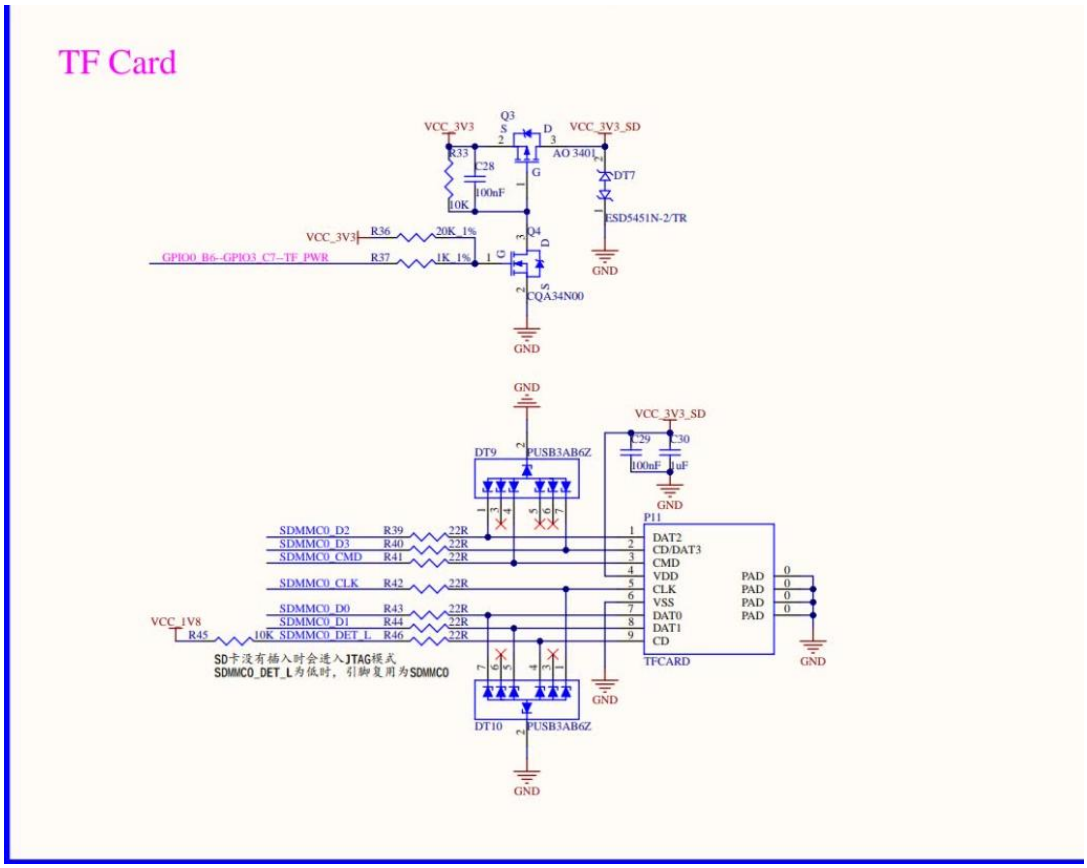
2.2.2 机械设计介绍

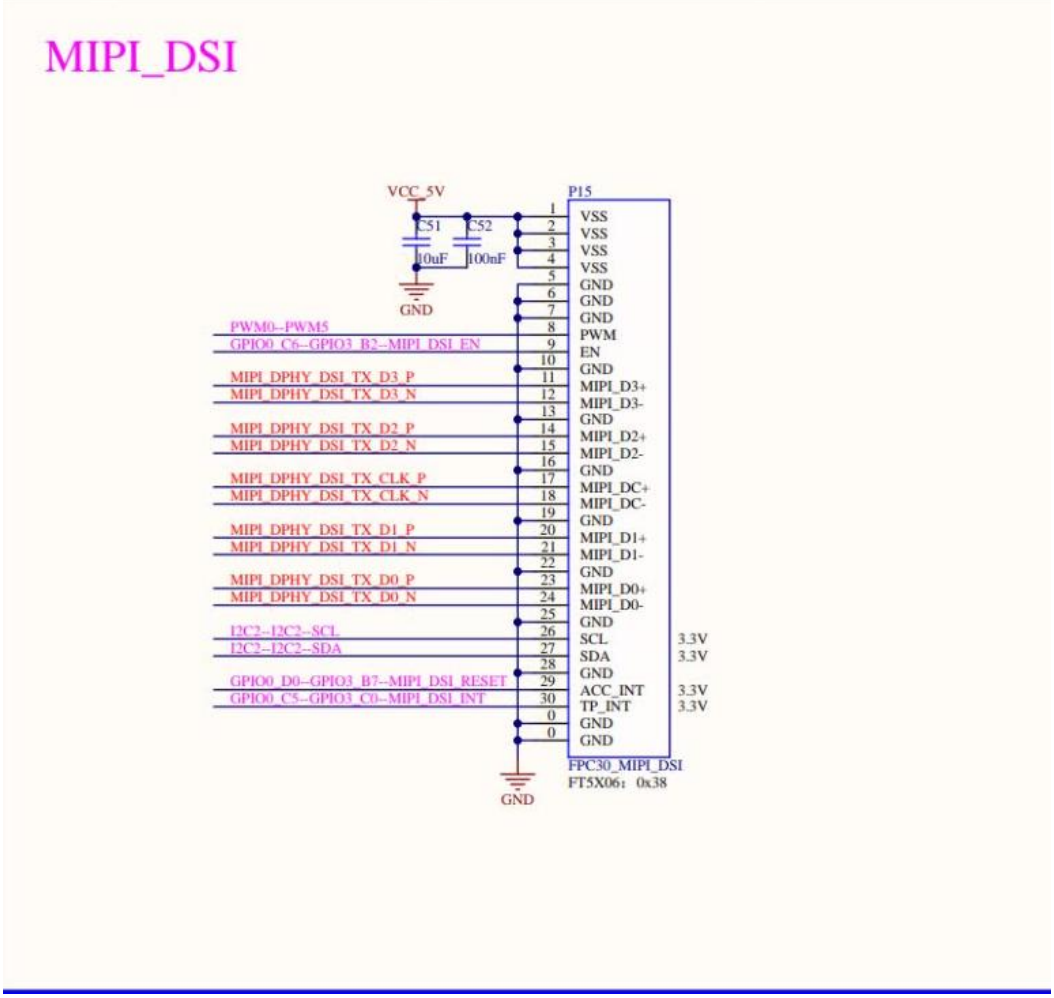


2.2.3 电路各模块介绍（从总体到局部，逐级给出各模块的具体设计图，并标记出关键的输入、输出信号线，可以是电路图、SCH 原理图、PCB 版图等截图）；


40pin 兼容树莓派
RTC
20pin自定义引脚功能
HDMI
USB-DP
DEBUG
POWER
KEY
MIPI-CSI
Codec Audio
USB 3.0
USB 2.0×2
1000M Ethernet


M.2 M-Key
M.2 E-Key
TF Card
MIPI-CSI
MIPI-DSI


TF Card
VCC_3V3
Q3
S
D
VCC_3V3_SD
R33
C28
100nF
10K
AO 3401
G
DT7
ESD5451N-2/TR
VCC_3V3
R36
20K_1%
Q4
GPIO0_B6--GPIO3_C7--TF_PWR
R37
1K_1%
CQA34N00
GND
GND
VCC_3V3_SD
DT9
PUSB3AB6Z
C29
C30
100nF
1uF
GND
P11
SDMMC0_D2
R39
22R
SDMMC0_D3
R40
22R
SDMMC0_CMD
R41
22R
SDMMC0_CLK
R42
22R
SDMMC0_D0
R43
22R
SDMMC0_D1
R44
22R
SDMMC0_DET_L
R46
22R
VCC_1V8
R45
10K
DAT2
CD/DAT3
CMD
VDD
CLK
VSS
DAT0
DAT1
CD
PAD
TFCARD
SD卡没有插入时会进入JTAG模式
SDMMC0_DET_L为低时，引脚复用为SDMMC0
DT10
PUSB3AB6Z
GND
GND



2.3 软件系统介绍

2.3.1 软件整体介绍（含 PC 端或云端，结合关键图片）；

系统分层设计

层级	组件	功能说明
桌面应用层	test.py	提供图形化用户界面（GUI）和入口
功能服务层	微信小程序+语音+坐姿 1.py voice_communicate.py	坐姿检测、语音交互核心逻辑
云端服务层	微信云开发+腾讯云COS	数据存储、多端同步

本系统采用三层架构设计：

- **桌面应用层**：提供用户交互界面（`test.py`）
- **主要功能服务层**：坐姿检测（`微信小程序+语音+坐姿1.py`）和语音助手（`voice_communicate.py`）
- **云端服务层**：微信云开发+腾讯云COS存储

系统工作流程：

1. 用户通过桌面应用启动功能
2. 坐姿检测/语音助手处理本地数据
3. 异常数据上传云端
4. 微信小程序访问云端数据

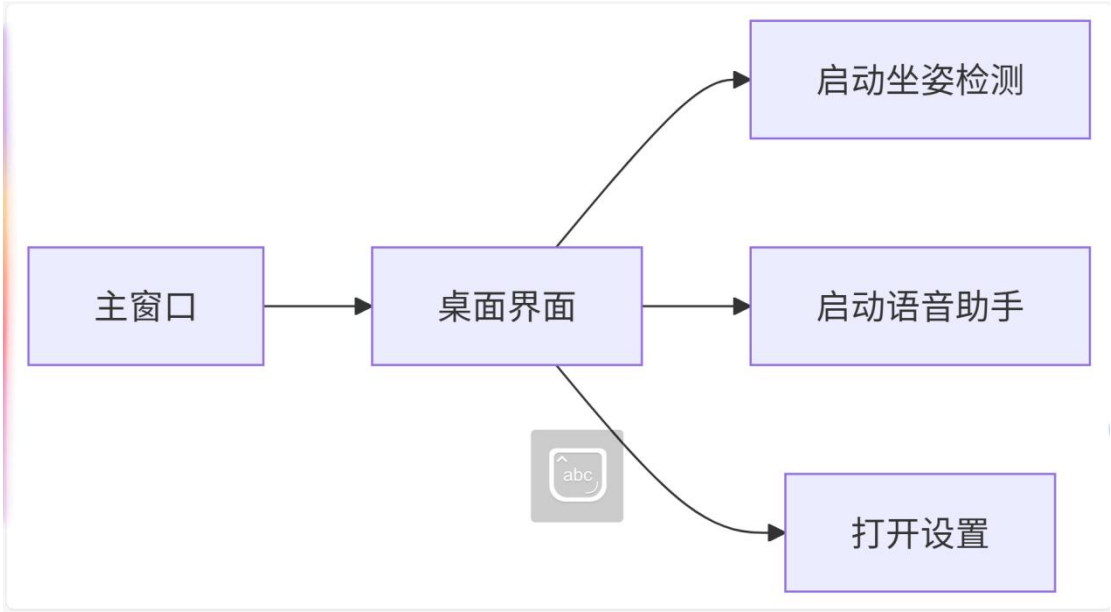
关键界面截图


智能坐姿监测系统
文件
帮助
FPS: 15.37
距离过近
异常姿势：1
持续时间：0.0秒
Hip Angle: 5.2??
Spine Angle: 0.5??
TOO CLOSE DETECTED!
状态: 检测到2个异常姿势
姿势状态
HUNCHBACK: 正常
SLOUCHING: 正常
UNEVEN SHOULDERS: 正常
TOO CLOSE: 警告
CROSSED LEGS: 正常
系统控制
摄像头:
自动检测
停止摄像头
手动截图
上次上传: 无
实时统计
FPS: 15.4
当前警告: 1
持续时间: 0.0秒
冷却时间: 0.0秒
检测设置
轮廓阈值:
坐姿阈值:
距离阈值:
30%
启用语音提醒
启用云服务
退出系统
就绪


智能语音助手
退出
开始录音
清空
录音时长 (秒):
10
准备就绪

2.3.2 软件各模块介绍（根据总体框图，给出各模块的具体设计说明。从顶层到底层逐次给出各函数的流程图及其关键输入、输出变量）；

1. 桌面应用模块 (test.py)



核心函数说明：

姿势检测核心函数说明

1. calculate_angle(a, b, c)

功能：计算三个点之间的角度

参数：

- a, b, c: 三个点的坐标 (x,y)

返回值：三个点形成的角度 (度)

算法：使用 arctan2 计算向量夹角

2. calculate_hip_angle(shoulder, hip, knee)

功能：计算髋关节角度 (用于检测坐姿倾斜)

参数：

- shoulder: 肩膀坐标
- hip: 髋部坐标
- knee: 膝盖坐标

返回值：髋关节角度 (度)

算法：使用向量点积公式计算角度

3. update_frame()

功能：主处理循环，处理每一帧摄像头图像

工作流程：

1.读取摄像头帧

2.计算 FPS

3.使用 MediaPipe 检测人体姿势

驼背 (HUNCHBACK)

- 检测依据: 脊柱弯曲角度超过预设阈值

- 说明：当人体脊柱的弯曲程度过大，形成明显的向前拱起状态时，判定为驼背姿势

肩部不平（UNEVEN SHOULDERS）

- 检测依据：左右肩膀的高度差超过设定阈值
- 说明：通过对比左右肩膀的坐标位置，计算其在垂直方向上的差值，当差值过大时判定为肩部不平

距离过近（TOO CLOSE）

- 检测依据：下巴位置超过图像高度阈值
- 说明：当用户与摄像头（或检测设备）的距离过近时，下巴在图像中的相对位置会偏高，超过设定的图像高度阈值则触发该检测结果

二郎腿（CROSSED LEGS）

- 检测依据：膝盖高度差超过阈值且脚踝位置接近
- 说明：此姿势需同时满足两个条件，一是左右膝盖在垂直方向上的高度差达到阈值，二是左右脚踝的坐标位置较为接近，以此判断是否处于二郎腿状态

1. 云服务相关函数

   3.1 save_to_cloudbase(display_url)
   功能：将截图 URL 保存到微信云开发数据库

- display_url：图片在 COS 的访问 URL

  返回值：成功 / 失败
  3.2 upload_file(file_path)
  功能：上传文件到腾讯云 COS

- file_path：本地文件路径

  返回值：文件在 COS 的访问 URL

1. 语音提醒系统

   VoiceAlerts 类
   功能：管理语音提醒队列和播放
   核心方法：

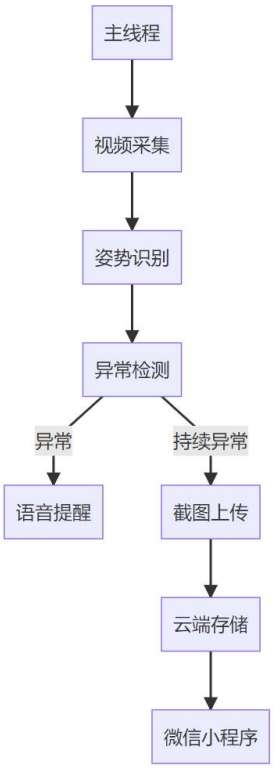
- add_alert (alert_type)：添加提醒到队列
- _play_alert (alert_type)：播放指定类型的提醒
- _process_alerts ()：处理提醒队列的后台线程

1. 状态管理

   GlobalState 类
   功能：管理全局状态和云指令轮询
   核心方法：

- poll_cloud_commands ()：从云端查询最新指令
- execute_command ()：执行云指令 (如启动 / 停止摄像头)
- refresh_token ()：刷新微信云开发访问令牌

2. 坐姿检测模块



算法实现（伪代码）

python 复制 下载

```python
def detect_posture(landmarks):
    # 脊柱弯曲检测
    upper_back = (left_shoulder + right_shoulder)/2
    lower_back = (left_hip + right_hip)/2
    spine_angle = calculate_angle(upper_back, lower_back, nose)

    # 多条件判断
    if spine_angle > threshold_hunch:
        add_warning("驼背")
    if hip_angle < threshold_slouch:
        add_warning("坐姿倾斜")
```

```
def calculate_angle(a, b, c)  # 计算关节点角度
输入：三个关节点坐标(x,y)
输出：角度值(0-180°)
流程：向量AB→向量BC→计算夹角

def detect_abnormal_postures()  # 异常姿势检测
输入：关节点坐标
输出：异常类型列表
流程：
  1. 计算脊柱弯曲角→驼背检测
  2. 计算髋关节角→坐姿倾斜
  3. 比较肩膀高度→肩部不平
  4. 检测下巴位置→距离过近
  5. 检测膝盖高度差→二郎腿
```

异常处理流程

一级检测：单次异常 → UI 标红提示

二级检测：持续 2 秒 → 播放语音提醒

三级检测：持续 5 秒 → 截图并上传云端

3. 语音助手模块

时序图关键步骤：

用户说话 → 录音线程启动 → 实时音频分帧 → WebSocket 传输 → 讯飞 API 返回文本 → DeepSeek 生成回复 → TTS 语音播放

关键参数配置：

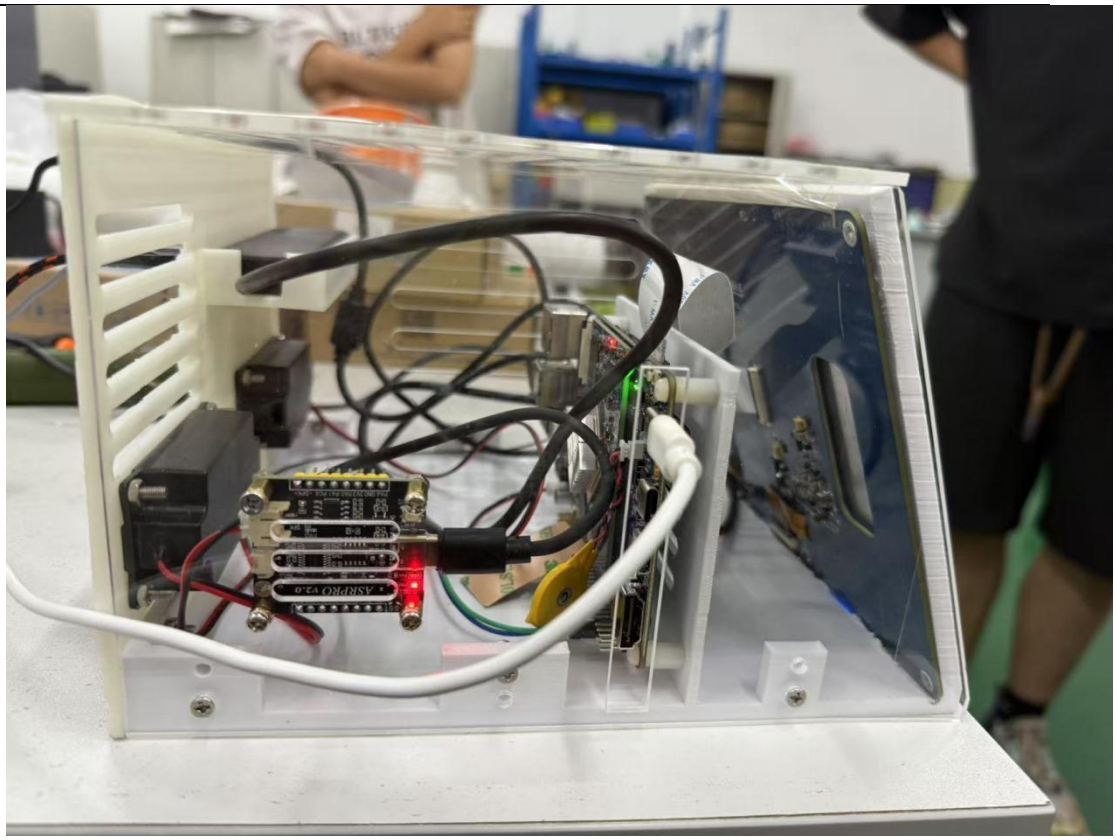
设置项	阈值	判定规则
驼背检测阈值（hunchback_threshold）	11（度）	脊柱弯曲角度超过 11 度，将被判定为驼背。例如，当系统检测到人体脊柱弯曲角度达到 12 度时，就会认为出现了驼背情况。
坐姿倾斜阈值（slouching_threshold）	68（度）	髋关节角度小于 68 度,将被判定为坐姿倾斜(葛优躺）。比如，若检测到髋关节角度为 65 度，此时就符合坐姿倾斜的判定标准。
肩部不平阈值（shoulder_diff_threshold）	0.015（图像比例）	左右肩膀高度差超过图像高度的 0.015 比例，将被判定为肩部不平。假设图像高度为 100 像素，当左右肩膀高度差超过 1.5 像素（100×0.015）时，就会判定肩部不平。

设置项	阈值	判定规则
距离过近阈值（desk_distance_threshold）	0.30（图像比例）	下巴位置超过图像高度的 0.30 比例，将被判定为距离屏幕过近。若图像高度为 200 像素，当下巴位置超过 60 像素（200×0.30）时，会被认为距离屏幕过近。
二郎腿阈值（leg_cross_threshold）	0.12（膝盖水平差）	膝盖高度差超过 0.12 且脚踝交叉，将被判定为二郎腿。例如，检测到膝盖高度差为 0.13 且脚踝处于交叉状态，就会判定为有二郎腿的姿势。
可见度阈值（min_visibility）	0.65	关键点可见度低于 0.65 将被忽略。在人体姿势检测中，若某个关键关节点（如肘部）的可见度为 0.6，那么在后续的姿势判定中这个点的信息将不被采用。
触发条件 - 异常姿势数量（min_warnings）	2	需要同时检测到至少 2 种异常姿势才会触发警告。比如同时检测到驼背和肩部不平两种异常姿势时，系统才会发出警告。
触发条件 - 异常姿势持续时间（min_duration）	5（秒）	异常姿势持续至少 5 秒才会触发截图。如果检测到坐姿倾斜，但持续时间只有 4 秒，就不会触发截图操作。
触发条件 - 截图间隔时间（cooldown）	10（秒）	两次截图之间的最小间隔时间为 10 秒。若在第 1 秒触发了一次截图，那么下一次截图最早要在第 11 秒才可能进行。

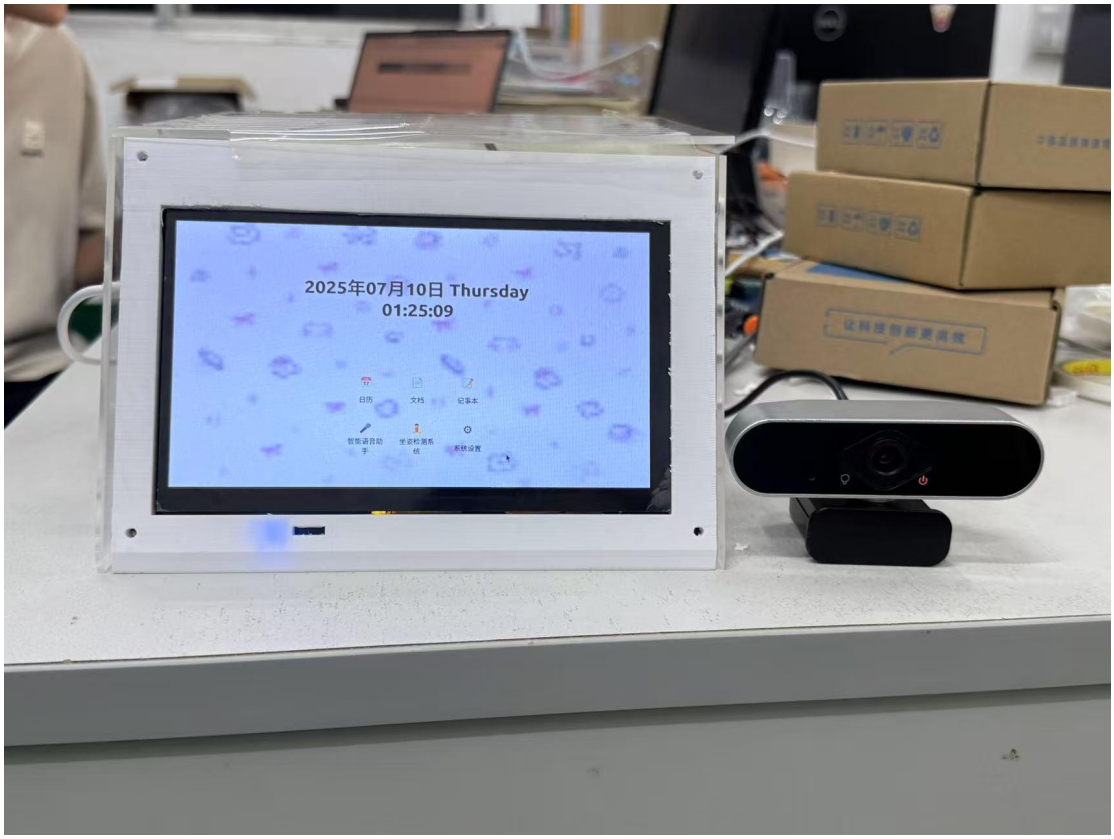
```
class VoiceRecognitionThread(QThread):
    # 语音识别线程
    def run():
        1. 开启录音设备
        2. 实时上传音频到讯飞API
        3. 解析返回的JSON结果
        4. 发送识别结果信号

    def on_message():  # WebSocket回调
        处理实时识别结果→拼接完整语句

class ModernVoiceAssistant(QMainWindow):
    # 主界面控制
    def start_recording():
        1. 创建识别线程
        2. 连接信号槽
        3. 启动线程

    def pyttsx3_say():  # 语音合成
        1. 初始化TTS引擎
        2. 设置语音参数(语速/音量)
        3. 播放生成的文本
```

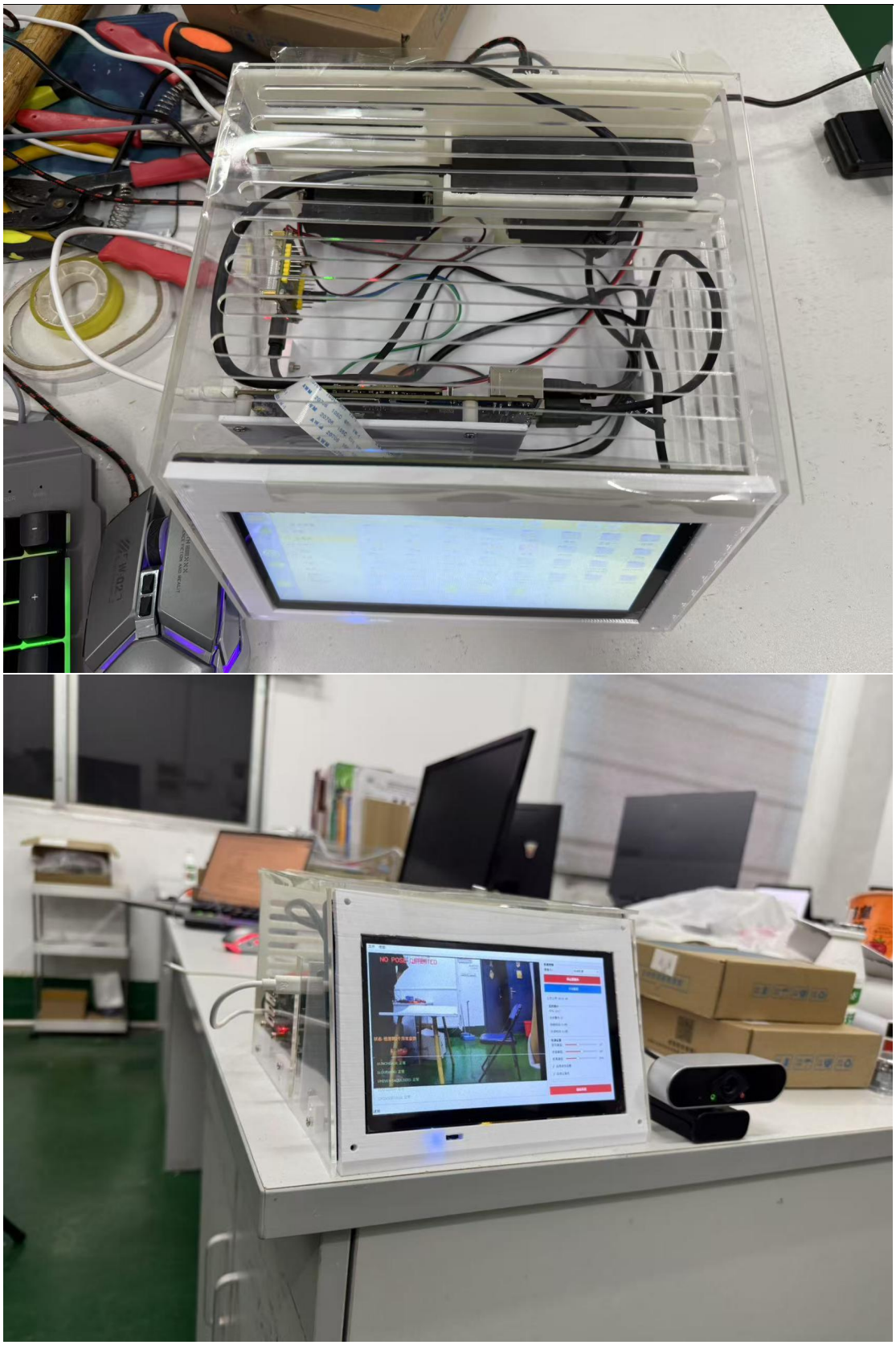
第三部分　完成情况及性能参数

阐述最终实现的成果（图文结合，实物照片为主）



3.1 整体介绍（整个系统实物的正面、斜 45°全局性照片）





3.2 工程成果（分硬件实物、软件界面等设计结果）

3.2.1 机械成果；（实物照片）

3.2.2 电路成果；（实物照片）

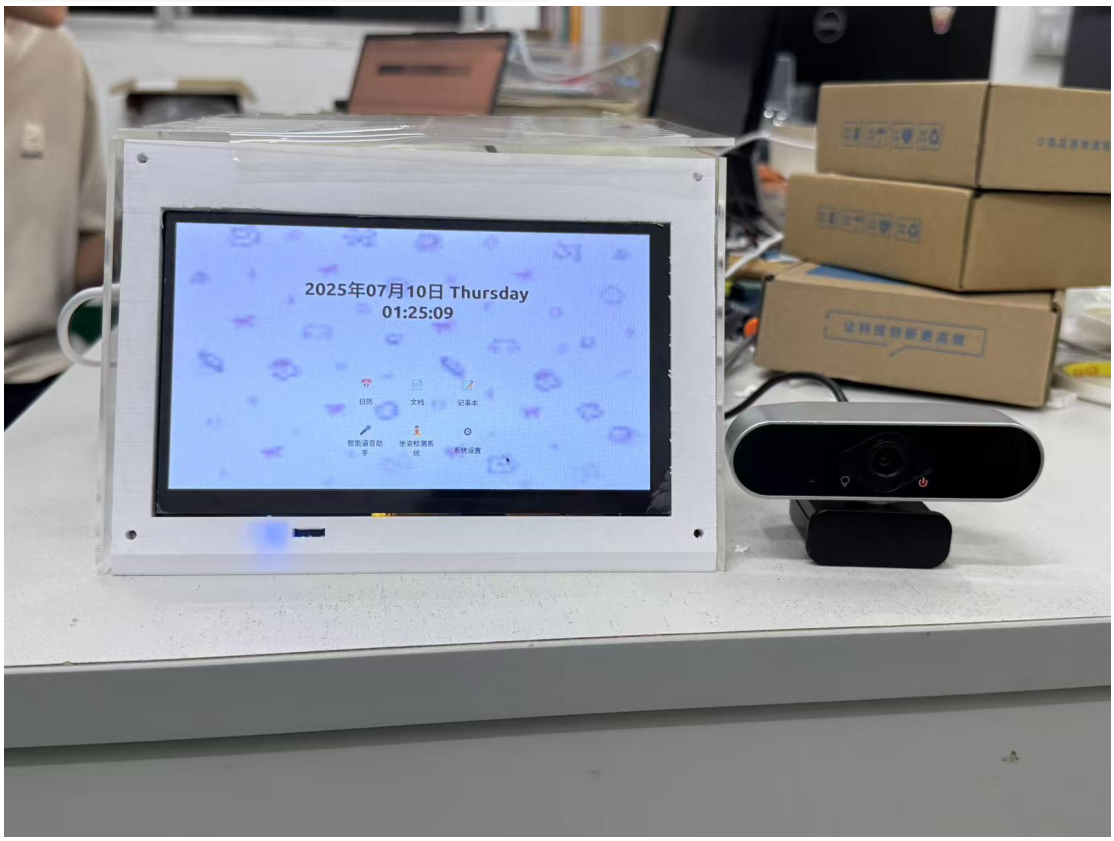
3.2.3 软件成果；（界面照片）

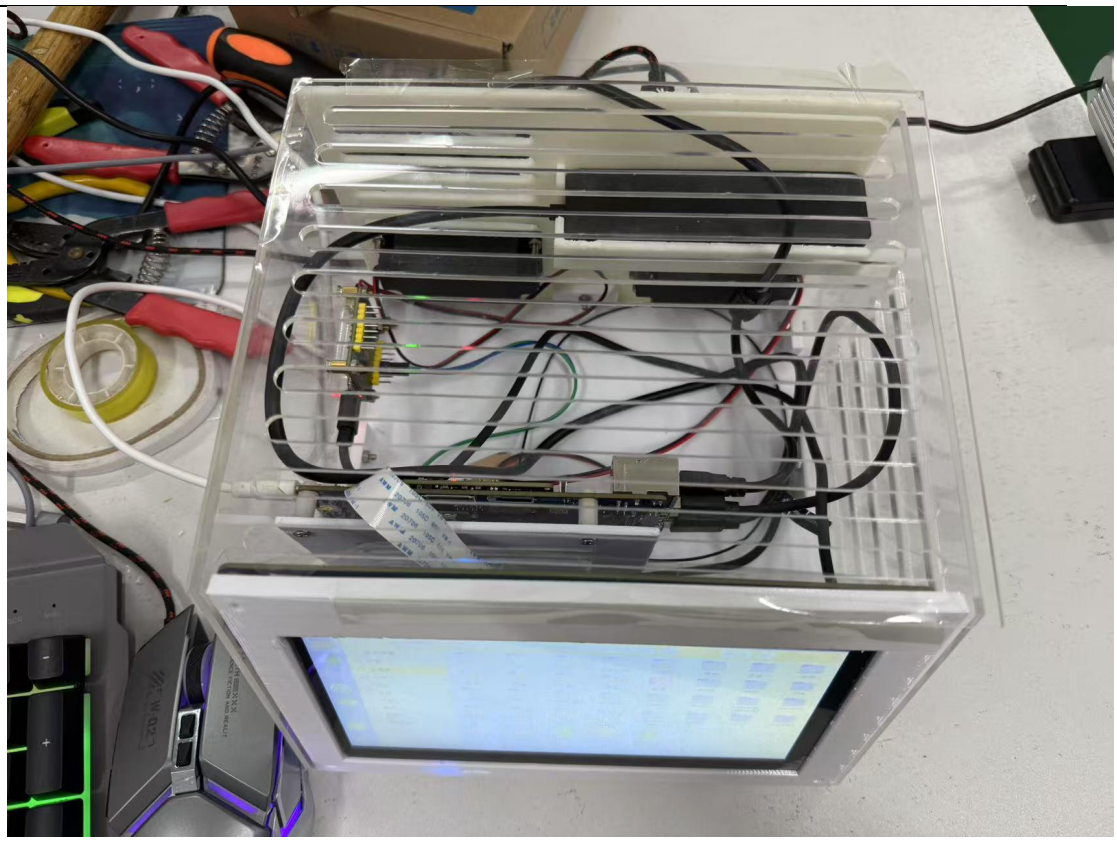

智能语音助手
退出
开始录音
清空
录音时长 (秒):
10


14:50
云开发 ...
已连接 远程
刷新连接
局域网
云服务
智能桌面
姿态检测
启动监测
停止监测
截图
设置
清空历史
操作记录
清空
python:/home/elf/main/微信小程···
14:50 云
快速开始
基础能力
我


14:51
基础能力
停止监听
手动刷新
导出数据
19:30:46
2025-07-04
19:30:23
2025-07-04
19:30:20
2025-07-04
19:29:56
2025-07-04
19:29:53
2025-07-04
19:29:42
2025-07-04
19:29:18
快速开始
基础能力
我


14:51
基础能力
停止监听
手动刷新
导出数据
19:30:46
驼背 | 肩部不平 | 距离过近
19:30:46 2025-07-04
19:29:53
19:29:42
19:29:18
快速开始
基础能力
我


2025年07月10日 Thursday
01:25:09



第四部分　总结

4.1　可扩展之处

1.多设备协同：对接智能升降桌：检测到驼背时自动调整高度；联动智能照明：根据环境光自动调节亮度；物联网集成：通过 MQTT 协议连接智能家居系统

2.健康数据分析平台：长期姿势变化趋势分析；生成周/月健康报告；提供个性化改善建议

3.AR 辅助矫正：HoloLens 3D 姿势指导；虚拟教练实时反馈；矫正动作游戏化

4.情感计算集成：面部表情疲劳分析；注意力集中度检测；情绪状态评估

4.2　心得体会

在本次"智能坐姿监测与语音助手系统"的研发过程中，我们团队经历了从需求分析、技术选型、系统设计到编码实现和测试优化的完整开发周期。整个项目融合了计算机视觉、语音交互、云服务和桌面应用开发等多个技术领域，是一次极具挑战性又收获颇丰的工程实践。

研发细节与挑战：

1.**多模态交互集成** 语音助手模块（voice_communicate.py）需要同时处理录音、语音识别、AI 对话和语音播报。初期遇到音频设备冲突问题，通过引入状态机管理（录音/处理/播报状态互斥）和线程隔离技术解决。特别优化了 Pyttsx3 的语音播报逻辑，避免打断正在进行的录音过程。

2.**实时姿态检测优化** 坐姿检测模块（微信小程序+语音+坐姿 1.py）的关键挑战是保证帧率的实时性。我们通过以下优化实现：

1.将 1280x720 输入降采样至 640x480 处理；2.仅计算必要的关节点角度（脊柱/髋关节/肩部）；3.使用多线程分离图像处理和云上传任务；4.设置关键点可见度阈值（min_visibility=0.65）过滤低质量检测

2.**云服务协同问题** 微信云开发与腾讯云 COS 的集成中遇到凭证刷新问题。最终实现 AccessTokenManager 自动刷新机制，并在上传失败时启用本地缓存队列，确保在网络波动时数据不丢失。

3.**跨进程通信设计** 桌面系统（测试（复件）.py）需要协调多个 Python 进程。通过串口监听线程实现硬件交互（如 YYTH 启动语音助手），并使用 subprocess 模块管理子进程生命周期，避免资源泄漏。

团队协作经验：

项目采用模块化开发模式，三人分别负责语音交互、坐姿检测和桌面系统三大核心模块。通过这次项目，我们深刻体会到系统级开发的复杂性，特别是在多模块协同和资源受限环境下的优化挑战。最大的收获是学会了如何在工程实践中

平衡功能实现与性能优化，这为我们后续开发更复杂的嵌入式智能系统积累了宝贵经验。

第五部分 参考文献

1.Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., ... & Grundmann, M. (2019). MediaPipe: A Framework for Building Perception Pipelines. arXiv preprint arXiv:1906.08172.

2.Rockchip. (2022). RK3588 Datasheet. Rockchip Electronics Co., Ltd.