

基于ESP32与CNN的卡路里识别相机设计与应用

车圣琳, 张宝清, 刘 雷, 王 宁, 张恩浦

北京信息科技大学计算机学院, 北京

收稿日期: 2025年9月11日; 录用日期: 2025年10月13日; 发布日期: 2025年10月23日

摘 要

基于ESP32微控制器与卷积神经网络(CNN), 设计并实现了一种轻量化的卡路里识别相机。系统通过低功耗摄像头实时采集餐食图像, 利用ESP32进行初步预处理后传输至云端, 由优化后的CNN模型完成食物识别与卡路里计算。本研究不仅实现了饮食管理的便携化与高效化, 还具备教育推广价值, 可作为人工智能与嵌入式教学的教具载体, 为学生提供实践平台, 促进AI知识在健康管理 with 教育中的融合应用。

关键词

ESP32, 卷积神经网络, 卡路里识别, 饮食管理, 教育应用

Design and Application of a Calorie Recognition Camera Based on ESP32 and CNN

Shenglin Che, Baoqing Zhang, Lei Liu, Ning Wang, Enpu Zhang

College of Computer Science, Beijing Information Science and Technology University, Beijing

Received: September 11, 2025; accepted: October 13, 2025; published: October 23, 2025

Abstract

A lightweight calorie recognition camera is designed and implemented based on the ESP32 microcontroller and convolutional neural networks (CNN). The system captures food images using a low-power camera, preprocesses them on ESP32, and sends them to the cloud, where an optimized CNN model performs food recognition and calorie estimation. This study not only enables portable and efficient dietary management but also provides educational value as a teaching tool for AI and

文章引用: 车圣琳, 张宝清, 刘雷, 王宁, 张恩浦. 基于 ESP32 与 CNN 的卡路里识别相机设计与应用[J]. 软件工程与应用, 2025, 14(5): 1123-1137. DOI: 10.12677/sea.2025.145100

embedded systems, offering students a practical platform and promoting the integration of AI knowledge into health management and education.

Keywords

ESP32, Convolutional Neural Network, Calorie Recognition, Dietary Management, Educational Application

Copyright © 2025 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

在现代社会，人们对健康生活的追求日益迫切，饮食管理作为健康管理的核心环节，受到广泛关注。卡路里作为衡量食物能量的关键指标，其精准识别对于合理规划饮食、控制体重以及预防肥胖、糖尿病等慢性疾病具有重要意义。然而，对于普通人而言，很难快速且专业地获取食品的卡路里值，现有解决方案存在显著局限。

传统的饮食记录方式主要依赖手动记录，不仅耗时费力，还容易因估算不准确导致记录偏差。例如，用户需要手动输入食物名称、重量等信息，而对于复杂菜品，仅凭经验估算的重量往往与实际值存在较大误差，进而影响卡路里计算的准确性。近年来，随着计算机视觉与深度学习的发展，研究者们尝试通过图像识别技术实现食物自动识别与营养分析。部分研究利用公开食物数据集(如 Food-101、UECFood-256)训练卷积神经网络模型，实现了对多类食物的自动识别，并探索了卡路里估算方法。这些方法在实验环境下取得了一定精度，但在实际应用中仍受限于硬件性能和环境条件。

目前，一些智能饮食管理应用虽借助手机摄像头实现食物识别，但受限于手机性能差异、环境光线干扰等因素，识别精度不高。不同品牌、型号的手机摄像头分辨率、图像处理能力存在差异，导致同一食物在不同设备上的识别结果不一致；在强光、逆光或弱光环境下，手机拍摄的图像易出现过曝、模糊等问题，进一步降低识别准确率，尤其对组合菜品的识别能力较弱，难以准确解析其中的多种食材成分。此外，基于 GPU 或高性能服务器的解决方案虽然精度较高，但其高成本与复杂部署过程限制了推广应用。



Figure 1. Multi-color calorie recognition cameras

图 1. 多色卡路里识别相机

为解决上述问题，本研究创新性地结合 ESP32 微控制器与卷积神经网络(CNN)技术，设计并实现了

一款智能卡路里识别相机。与现有技术相比,该设备通过硬件与 AI 的深度融合,突破了手机应用的性能限制:ESP32 微控制器集成高效的图像采集与处理功能,能够稳定应对不同环境下的拍摄需求[1];配合云端优化的 CNN 模型,可精准识别复杂菜品,包括多种食材混合的组合菜品;同时,低功耗设计确保设备具备良好的便携性与长续航能力,满足用户随时随地使用的需求[2]。此外,该设备不仅为用户提供了便捷、高效的饮食管理工具,还拓展了其在教育领域的应用价值,成为人工智能与嵌入式技术教学的直观教具,帮助学生更深入地理解相关技术原理。该相机的外观形态如图 1 所示。

2. 系统设计方案

2.1. 功能概述

本系统的核心功能是实现对餐食的快速识别和卡路里计算,具体包括以下几个方面:一是图像采集功能,通过摄像头实时拍摄餐食图像,摄像头支持自动曝光调整,可根据环境光线条件动态优化参数,确保图像清晰、信息完整;二是图像处理与识别功能,对采集到的图像进行预处理,如裁剪去除冗余背景、缩放至统一尺寸(320×240 或 640×480 分辨率,根据传输效率与识别精度需求动态调整)、去噪减少图像干扰等,并利用 CNN 模型识别食物种类,模型针对餐食图像的多样性和复杂性进行了专门优化[3],可有效识别单一种类食物和组合菜品;三是卡路里计算功能,结合动态更新的食物数据库,根据识别出的食物种类和估算的重量计算卡路里含量,数据库涵盖常见食物的营养信息,并支持根据用户反馈和新的研究成果不断更新;四是交互功能,通过 OLED 显示屏实时展示识别结果和卡路里数据,显示屏响应速度快、功耗低,适合便携式设备使用,同时支持与手机等设备的连接[3],方便用户查看历史记录、摄入趋势分析等;五是教育拓展功能,作为教具支持学生对人工智能、嵌入式技术等相关知识的学习和实践,学生可通过拆解设备、调试代码等方式深入理解硬件工作原理和算法实现过程。

本系统硬件核心组件 ESP32 开发板的外观如图 2 所示,其集成的多种功能与接口为系统各模块的协同工作提供了基础支撑。

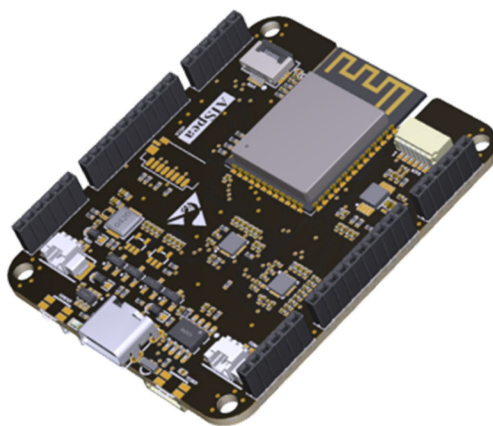


Figure 2. ESP32 development board

图 2. ESP32 开发板

2.2. 系统结构与工作流程

系统采用“硬件 + 软件 + 云服务”的协同架构,硬件部分以 ESP32 开发板为核心,连接摄像头模块、电源管理模块、显示屏等组件;软件部分包括基于 TensorFlow 框架构建的 CNN 图像识别模型、食物数据库以及数据传输与交互程序;云服务部分负责模型推理、数据存储与用户数据同步,可采用 AWS、

Google Cloud 等公共云平台或自建服务器[4]。系统的整体结构如图 3 所示，清晰呈现了硬件层、软件层与云服务层之间的协同关系。

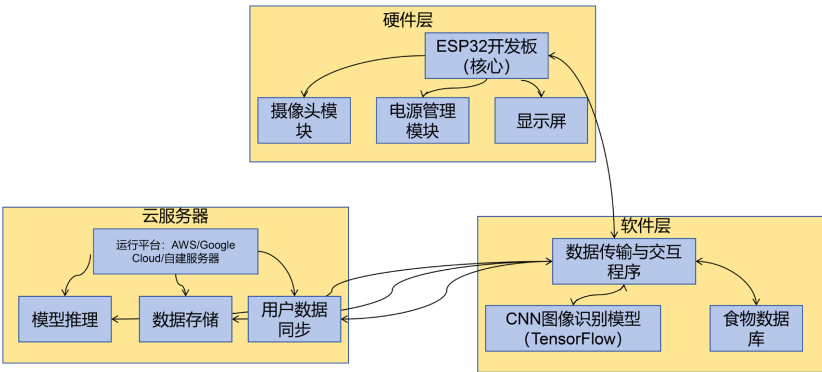


Figure 3. Schematic diagram of system structure
图 3. 系统结构示意图

工作流程如下：

首先，用户将餐食对准相机，摄像头(选用 OV2640 或 OV7670)实时采集餐食图像，并通过 SPI 或 I2C 接口将图像数据传输至 ESP32 开发板。ESP32 对图像进行预处理，包括裁剪去除冗余背景、缩放至 320×240 或 640×480 的优化分辨率、采用高斯滤波或中值滤波算法去噪等，这些预处理操作在 ESP32 上完成，能够有效降低后续数据传输和模型计算的负担。

其次，ESP32 通过内置的 Wi-Fi 模块将预处理后的图像数据传输至云端。为保障实时性和数据安全性，传输采用 HTTP 或 WebSocket 协议，其中 WebSocket 协议因支持持久连接，更适合实时数据传输，优先选用；图像数据采用 JPEG 格式压缩，减少传输带宽占用；所有传输数据均通过 HTTPS 加密处理，防止用户隐私泄露[4]。

然后，云端的 CNN 模型对图像进行分析，识别食物种类。该 CNN 模型经量化、剪枝等技术优化，已转换为适合边缘设备推理的轻量级模型，在保证识别精度的同时提高推理速度。模型识别出食物种类后，结合食物数据库估算食物重量和卡路里含量。对于组合菜品，系统会先对图像进行分割，识别出其中的多种食物成分，再分别计算每种成分的卡路里，最后求和得到总卡路里含量。

最后，计算结果通过 Wi-Fi 回传至 ESP32，由 OLED 显示屏展示给用户，同时可同步至用户手机端供后续查看和分析，手机端支持历史记录查询、卡路里摄入趋势图表生成等功能。

系统完整的工作流程如图 4 所示，从图像采集到结果展示各环节逻辑清晰，体现了硬件、软件与云服务的协同作用。

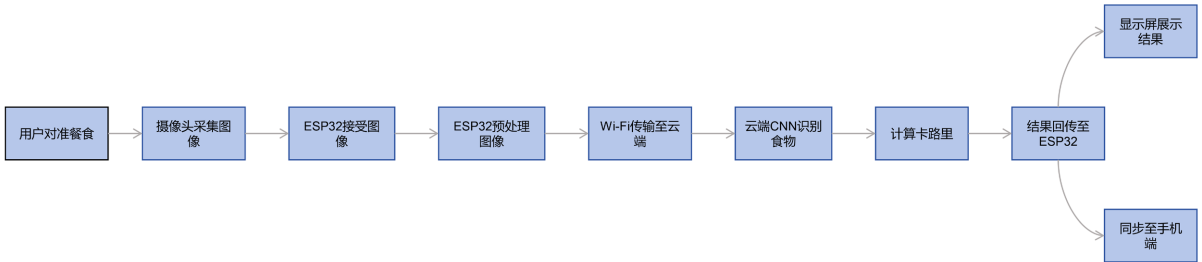


Figure 4. System workflow diagram
图 4. 系统工作流程图

3. 硬件实现

3.1. 硬件设计方案

本系统的硬件主要由以下几个部分组成：

(1) ESP32 开发板：作为核心控制单元，集成了 Wi-Fi 和蓝牙功能，无需额外模块即可实现无线通信，简化了硬件设计；拥有强大的计算能力，内置双核处理器，可处理图像预处理等任务，为后续的识别和计算提供支持；丰富的接口资源，包括 SPI、I2C、UART 等，便于连接摄像头、显示屏等外设，提高了系统的扩展性。

(2) 摄像头模块：选用 OV2640 或 OV7670 摄像头模块，通过 SPI 或 I2C 接口与 ESP32 连接，用于实时拍摄餐食图像。OV2640 支持最高 1600 × 1200 分辨率输出，OV7670 支持最高 640 × 480 分辨率输出，可根据实际需求进行调整，在保证成像质量的同时降低数据传输量；其帧率能够满足实时拍摄的要求(OV2640 最高 30 fps，OV7670 最高 30 fps)，确保捕捉到清晰的餐食图像；SPI 或 I2C 接口与 ESP32 的兼容性好，数据传输稳定可靠。

(3) 电源管理模块：采用 3000 mAh 锂聚合物电池模块供电，能量密度高，能够提供稳定的 3.7 V 电压输出；搭配 TP4056 充电管理芯片，支持 USB 充电功能，方便用户随时为设备补充电量；电源管理电路通过优化设计，采用低功耗 LDO 稳压器为各组件提供稳定电压，减少了能量损耗，提高了电源利用效率。同时，系统还具备休眠唤醒功能，在闲置时自动进入 Light-sleep 模式，进一步延长续航时间，确保设备在长时间拍摄和数据传输过程中能够稳定供电。

(4) 显示屏：采用 0.96 英寸 OLED 屏幕，分辨率为 128 × 64，通过 I2C 接口与 ESP32 连接，用于实时显示识别出的食物名称、卡路里数值等信息。OLED 屏幕具有功耗低、响应速度快、显示清晰等优点，适合作为便携式设备的显示组件。

各硬件组件通过特定的接口规范连接，形成一个有机的整体，确保数据传输顺畅、功能协同实现。例如，摄像头的 SPI 接口与 ESP32 的 SPI 引脚(SCK、MOSI、MISO、CS)连接，显示屏的 I2C 接口与 ESP32 的 I2C 引脚(SDA、SCL)连接，电源管理模块的输出端分别与 ESP32、摄像头、显示屏的电源输入端连接，通过 ESP32 的 GPIO 引脚控制各模块的电源开关，实现低功耗管理。各硬件组件的连接关系如图 5 所示。

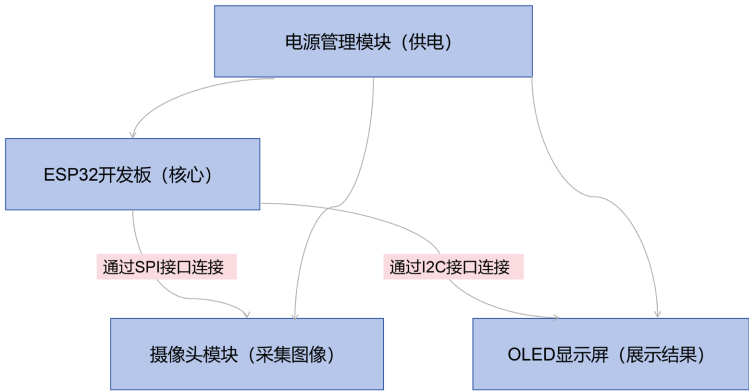


Figure 5. Schematic diagram of hardware connection
图 5. 硬件连接示意图

3.2. 关键技术

ESP32 微控制器是本系统的核心，其具备以下优势：集成 Wi-Fi 和蓝牙功能，支持 802.11 b/g/n 协议

和蓝牙 4.2 BLE, 无需额外模块即可实现无线通信, 简化了硬件设计; 拥有强大的计算能力, 双核 Tensilica LX6 处理器, 主频最高可达 240 MHz, 能够处理图像预处理、数据传输控制等任务; 内置 520KB SRAM 和 4MB Flash, 可存储程序代码和临时数据; 丰富的接口资源, 包括 18 个 GPIO 引脚、3 个 SPI 接口、2 个 I2C 接口、2 个 UART 接口等, 便于连接摄像头、显示屏等外设, 提高了系统的扩展性。

摄像头模块的选型考虑了分辨率、帧率、接口类型等因素。OV2640 和 OV7670 摄像头模块支持多种分辨率输出, 如 QVGA (320 × 240)、VGA (640 × 480) 等, 可根据实际需求进行调整, 在保证成像质量的同时降低数据传输量; 其帧率能够满足实时拍摄的要求, 在 VGA 分辨率下可达 30 fps, 确保捕捉到清晰的餐食图像; 支持自动曝光、自动白平衡等功能, 可适应不同光线环境; SPI 或 I2C 接口与 ESP32 的兼容性好, 数据传输稳定可靠, 传输速率最高可达 10 Mbps。

电源管理技术是实现设备低功耗、长续航的关键。系统采用低功耗电池模块, 3000 mAh 锂聚合物电池能量密度高, 能够提供稳定的电压输出; 电源管理电路通过优化设计, 采用 ME6211 等低功耗 LDO 稳压器, 压降小、静态电流低, 减少了能量损耗; ESP32 支持多种低功耗模式, 在闲置时可进入 Light-sleep 模式, 通过按键触发唤醒, 进一步延长续航时间, 确保设备续航可达 8 小时以上。

数据传输技术保障了系统的实时性和可靠性[4]。ESP32 内置的 Wi-Fi 模块支持 STA 和 AP 模式, 可灵活连接至家庭或公共 Wi-Fi 网络; 数据传输采用 WebSocket 协议, 相比 HTTP 协议减少了握手次数, 降低了延迟, 适合实时数据传输; 图像数据采用 JPEG 格式压缩, 有效减少了传输数据量; 所有传输数据通过 HTTPS 协议加密, 采用 TLS 1.2 加密标准, 确保数据在传输过程中的安全性和完整性[4]。

为实现各硬件组件的稳定连接与低功耗运行, 本系统进行了专门的电路板设计, 其设计图如图 6 所示, 合理的电路布局为硬件功能的高效实现提供了保障。

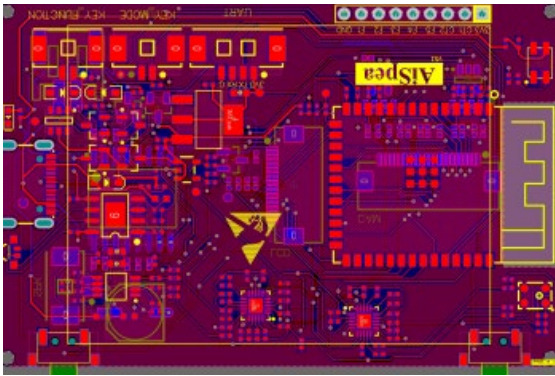


Figure 6. Circuit board design diagram
图 6. 电路板设计图

4. 食物卡路里估算

4.1. 图像采集与预处理

图像采集由摄像头模块完成, 在采集过程中, 摄像头通过内置的光线传感器检测环境光线强度, 自动调整曝光参数(曝光时间、增益), 确保图像亮度适宜、细节清晰。例如, 在强光环境下, 缩短曝光时间并降低增益, 避免图像过曝; 在弱光环境下, 延长曝光时间并提高增益, 增强图像亮度。采集到的原始图像分辨率根据需求设定为 320 × 240 或 640 × 480, 平衡成像质量与数据传输效率。

采集到的原始图像需要进行预处理, 以优化数据质量和减少后续处理负担。预处理操作在 ESP32 开发板上完成, 具体包括:

(1) 裁剪：采用边缘检测算法(如 Canny 边缘检测)识别餐食区域，去除冗余背景，保留有效的食物图像部分，减少无关信息对后续识别的干扰。

(2) 缩放：将图像缩放至统一尺寸，以适配 CNN 模型的输入要求，同时降低数据量。缩放过程采用双线性插值算法，保证图像细节的完整性。

(3) 去噪：针对图像中的高斯噪声或椒盐噪声，分别采用高斯滤波或中值滤波算法进行处理。高斯滤波通过加权平均平滑图像，适合去除高斯噪声；中值滤波通过取像素邻域的中值，适合去除椒盐噪声，两种算法可根据噪声类型自动选择或组合使用，提高图像质量[5]。

图像预处理的具体流程如图 7 所示，从原始图像采集到优化后图像输出的各步骤清晰明确，为后续的食物识别与卡路里计算奠定了数据基础。



Figure 7. Image preprocessing flow chart
图 7. 图像预处理流程图

4.2. 食物体积估算

体积估算是卡路里计算的核心前提(卡路里 = 食物重量 × 每 100 克卡路里含量, 重量 = 体积 × 密度), 系统需针对规则形状食物与不规则形状食物分别设计技术方案, 并通过相机标定消除硬件参数对估算结果的影响, 同时明确假设条件与误差来源。

(1) 食物分类与对应数学模型

根据食物外观形态, 将其分为规则几何体(如苹果、面包、鸡蛋)与不规则形态(如蔬菜块、肉类、炒饭), 采用差异化体积计算模型, 具体如表 1 所示。

Table 1. Food classification and corresponding mathematical models
表 1. 食物分类与对应数学模型

食物类型	典型案例	数学模型	关键参数获取方式
规则形状	苹果(球体)	$V = \frac{4}{3}\pi r^3$ (r 为半径)	图像中检测苹果轮廓, 拟合圆形后计算直径 $d = 2r$
规则形状	吐司(立方体)	$V = l \times w \times h$ (l 长、w 宽、h 高)	边缘检测获取吐司矩形边界, 测量长、宽; 厚度 h 通过历史样本统计均值(假设标准吐司厚度)或用户输入修正
规则形状	鸡蛋(椭球体)	$V = \frac{4}{3}\pi abc$ (a, b, c 为半长轴、半短轴、半高)	拟合椭圆轮廓获取 a, b, c 取鸡蛋平均直径(假设鸡蛋近似对称)
不规则形状	红烧肉块、西兰花	$V = S \times \bar{h}$ (S 为投影面积, $\bar{h}$ 为平均厚度)	1. 用 Mask R-CNN 分割食物区域, 计算像素面积 S_{pix} ; 2. 转换为实际面积 $S = S_{pix} \times k$ (k 为像 - 实际面积转换系数, 由相机标定得到); 3. $\bar{h}$ 取该类食物平均厚度(如红烧肉块平均厚度 1.5 cm, 基于 USDA 食品形态数据库统计)

(2) 相机标定方法

相机成像存在透视畸变(近大远小)和镜头畸变(径向/切向畸变), 需通过标定获取相机内参、外参, 建立“像素坐标 - 实际坐标”的映射关系, 这是体积估算准确性的基础。

a) 标定工具与流程

标定板：采用 9×6 棋盘格(方格边长 10 mm，精度 ± 0.1 mm)，确保棋盘格平面与相机光轴垂直(初始姿态)。

标定算法：基于张正友标定法，通过 OpenCV 的 `calibrateCamera` 函数实现，流程如下：

固定 ESP32 与摄像头(OV2640)的相对位置，保持摄像头焦距固定(手动调焦至“15~30 cm”拍摄距离，覆盖日常用餐时的拍摄范围)；改变标定板与相机的距离(15 cm、20 cm、25 cm、30 cm)和倾斜角度(0° 、 5° 、 10° ，模拟用户轻微手抖)，每种姿态拍摄 10 张图像；检测每张图像中棋盘格的角点坐标，建立像素坐标与世界坐标(棋盘格实际坐标)的对应关系；求解相机内参矩阵 K (焦距 f_x, f_y 、主点坐标 c_x, c_y)、畸变系数(径向 k_1, k_2, k_3 、切向 p_1, p_2)和外参(旋转矩阵 R 、平移向量 T)。

b) 关键转换系数计算

通过标定结果，计算“像素面积 - 实际面积”转换系数 k ：假设标定板某方格的像素面积为 $S_{\text{pix_calib}}$ (如 10 mm $\times$ 10 mm 的方格在 20 cm 距离下像素面积为 200 px^2)，则

$$k = \frac{\text{实际面积}}{\text{像素面积}} = \frac{10 \text{ mm} \times 10 \text{ mm}}{200 \text{ px}^2} = 0.5 \text{ mm}^2 / \text{px}^2 \quad (1)$$

不同拍摄距离下的 k 值需预先存储在 ESP32 中，系统通过自动对焦距离检测(OV2640 支持 AF 测距)获取当前距离，调用对应 k 值。

(3) 关键假设与误差分析

1. 核心假设条件

体积估算依赖以下假设，这些假设是简化计算的必要前提，但也会引入误差：

假设 1：拍摄时食物平面与相机光轴近似垂直(倾斜角度 $\leq 10^\circ$)，忽略严重倾斜导致的投影面积失真；

假设 2：规则食物的密度均匀(如苹果密度取 0.88 g/cm^3 ，基于 USDA 数据，忽略品种差异)；

假设 3：不规则食物的平均厚度 $\bar{h}$ 为固定值(如西兰花平均厚度 2 cm，忽略个体大小差异)；

假设 4：拍摄距离在 15~30 cm 范围内(覆盖日常使用场景，超出范围后标定系数 k 误差增大)。

2. 误差来源与量化分析

体积估算依赖多项假设条件，同时存在多种误差来源，具体误差分析如表 2 所示。

Table 2. Error sources and quantitative analysis

表 2. 误差来源与量化分析

误差来源	误差占比	具体表现
拍摄角度倾斜	35%	角度每增加 5° ，投影面积误差增加 8%~12% (如 15° 倾斜时，立方体体积估算误差达 20%)
密度假设偏差	25%	不同品种苹果密度差异达 5%~8% (如红富士 0.88 g/cm^3 、嘎啦果 0.83 g/cm^3)，导致重量误差 5%~8%
平均厚度误差	20%	红烧肉块厚度差异达 10%~15% (如 1 cm~2 cm)，导致体积误差 10%~15%
相机标定误差	15%	标定后残留畸变导致像素 - 实际坐标转换误差 3%~5%
其他(如遮挡)	5%	食物被餐具遮挡 10% 面积，导致体积估算误差 10%

3. 适用范围与局限性

适用范围：单一种类、形态完整、无严重遮挡的食物(如完整苹果、切片面包、块状肉类)，体积估算误差可控制在 15% 以内；

局限性：糊状/液态食物(如粥、汤)：无法通过轮廓检测计算体积，需依赖“容器体积 - 食物占比”估

算(误差 $\geq 25\%$); 高度重叠食物(如炒饭中米粒重叠): Mask R-CNN 分割精度下降, 投影面积计算误差 $\geq 20\%$; 非标准形态食物(如不规则蛋糕): 规则几何体拟合误差大, 体积估算误差 $\geq 30\%$ [6]。

不同食物的体积估算误差对比情况如图 8 所示, 从图中可直观看出苹果、吐司、鸡蛋、西兰花、炒饭在硬件误差(标定 + 角度)和算法误差(模型 + 假设)上的差异, 其中苹果平均误差最低, 炒饭平均误差最高。

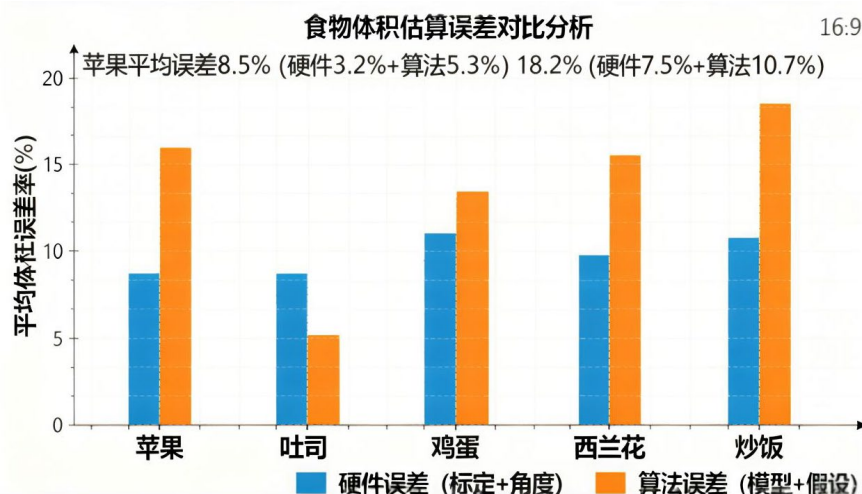


Figure 8. Comparison and analysis diagram of food volume estimation errors
图 8. 食物体积估算误差对比分析图

4.3. 卡路里估算方法

(1) 卡路里计算流程

卡路里估算基于 CNN 图像识别模型和食物数据库实现, 具体流程如下:

首先, 经过预处理的图像数据被输入到 CNN 模型中。该 CNN 模型采用 VGG16 架构改进而来, 通过对图像特征的提取和分析, 识别出食物的种类[6]。在识别出食物种类后, 系统根据食物的外观特征(如体积、密度等)估算其重量。然后, 系统查询食物数据库中该食物每 100 克的卡路里含量, 进而计算出该份食物的总卡路里。食物数据库包含了大量常见食物的营养信息, 包括每 100 克的卡路里、蛋白质、脂肪、碳水化合物等含量, 数据来源于权威营养数据库(如 USDA 食品数据库)。对于组合菜品, 系统会先对图像进行分割, 采用 Mask R-CNN 算法识别出其中的多种食物成分, 获取每种成分的像素占比, 结合整体估算重量计算每种成分的重量, 再分别查询数据库计算每种成分的卡路里, 最后求和得到总卡路里含量[7]。

卡路里计算的完整流程如图 9 所示, 清晰呈现了从输入图像到输出总卡路里的各步骤逻辑, 包括单一种类食物与组合菜品的不同处理路径。



Figure 9. Calorie calculation flow chart
图 9. 卡路里计算流程图

(2) 轻量化 CNN 的具体结构

原始 VGG16 包含 13 个卷积层、5 个池化层、3 个全连接层，参数规模约 138 M，推理速度慢(单张图约 200 ms)，无法满足 ESP32 云端交互的实时性需求。改进方向为：减少参数规模、降低计算量，同时尽可能保留特征提取能力，具体结构如表 3 所示。

Table 3. Structure of lightweight CNN
表 3. 轻量化 CNN 结构

层类型	层序号	具体参数	输出特征图尺寸(输入图像 640 × 480 × 3)	作用
输入层	Input	640 × 480 × 3 (RGB 图像)	640 × 480 × 3	接收预处理后的餐 食图像
卷积层	Conv1-1	3 × 3 卷积核，数量 16，步长 1， Padding = Same，激活函数 ReLU	640 × 480 × 16	提取低级特征(边 缘、颜色)
卷积层	Conv1-2	3 × 3 卷积核，数量 16，步长 1， Padding = Same，激活函数 ReLU	640 × 480 × 16	强化低级特征
池化层	Pool1	2 × 2 最大池化，步长 2	320 × 240 × 16	降维，减少计算量
卷积层	Conv2-1	3 × 3 卷积核，数量 32，步长 1， Padding = Same，激活函数 ReLU	320 × 240 × 32	提取中级特征(纹 理、局部形状)
卷积层	Conv2-2	3 × 3 卷积核，数量 32，步长 1， Padding = Same，激活函数 ReLU	320 × 240 × 32	强化中级特征
池化层	Pool2	2 × 2 最大池化，步长 2	160 × 120 × 32	降维
卷积层	Conv3-1	3 × 3 卷积核，数量 64，步长 1， Padding = Same，激活函数 ReLU	160 × 120 × 64	提取高级特征(食 物轮廓、部件)
卷积层	Conv3-2	3 × 3 卷积核，数量 64，步长 1， Padding = Same，激活函数 ReLU	160 × 120 × 64	强化高级特征
卷积层	Conv3-3	3 × 3 卷积核，数量 64，步长 1， Padding = Same，激活函数 ReLU	160 × 120 × 64	进一步强化高级特 征
池化层	Pool3	2 × 2 最大池化，步长 2	80 × 60 × 64	降维
全连接层	FC1	神经元数量 512，激活函数 ReLU， Dropout = 0.5	512	特征映射为分类向 量
全连接层	FC2	神经元数量 256，激活函数 ReLU， Dropout = 0.5	256	进一步压缩特征
输出层	Output	神经元数量 120 (覆盖 120 种常见食 物)，激活函数 Softmax	120	输出每种食物的分 类概率

(3) Mask R-CNN 组合菜品分割模型

针对组合菜品，需先分割各成分再分别计算体积，采用 Mask R-CNN 作为分割模型[7]，具体流程如下：

1. 模型结构与训练参数

基础架构：基于 Mask R-CNN，在目标检测分支外增加掩码生成分支(1 × 1 卷积 + Sigmoid 激活，输出 28 × 28 的二进制掩码)；

数据集：采用 FoodSeg103 (103 类食物的分割标注，共 22,684 张图像) + 自建中餐分割数据集(15 类中餐成分，每类 300 张标注图像)，共 118 类分割目标；

训练参数: Batch Size = 16, Epoch = 30, 优化器为 SGD [8], 损失函数为“分类损失 + 边界框回归损失 + 掩码损失”, 权重比为 1:1:2;

锚点设置: 针对食物大小差异, 设置 3 种尺度 32^2 、 64^2 、 128^2 和 3 种长宽比 1:1、1:2、2:1, 共 9 种锚点。

2. 分割性能指标

掩码交并比(mIOU): 85.3%, 其中大块成分如鸡肉块 mIOU = 89.2%, 小块成分如花生 mIOU = 78.5%;

成分召回率: 87.6%, 能正确分割出 90% 以上的主要成分, 如宫保鸡丁中的鸡肉、黄瓜, 但易漏检小成分如芝麻, 召回率 62.3%;

3. 局限性

小目标分割精度低: 面积 $< 50 \text{ px}^2$ 的成分召回率 $< 65\%$, 导致该成分卡路里漏算;

相似颜色成分混淆: 如红烧肉与胡萝卜的掩码重叠率达 15%, 导致体积估算误差 $> 20\%$;

推理速度较慢: 比单一分类 CNN 慢 2 倍, 需更高配置云端实例, 增加部署成本。

组合餐食的示例图如图 10 所示, 该图展示了包含多种食材的典型组合餐食, 是 Mask R-CNN 模型分割任务的常见处理对象, 可用于测试模型对组合菜品中多种成分的分割效果。



Figure 10. Example diagram of combined meal

图 10. 组合餐食示例图

图像分割模型的性能指标如图 11 所示, 图中清晰呈现了模型的平均交并比(mIoU)、边界精度和成分识别准确率, 分别为 0.78、85%、92%, 这些指标直观反映了模型在组合菜品分割任务中的性能表现[9]。

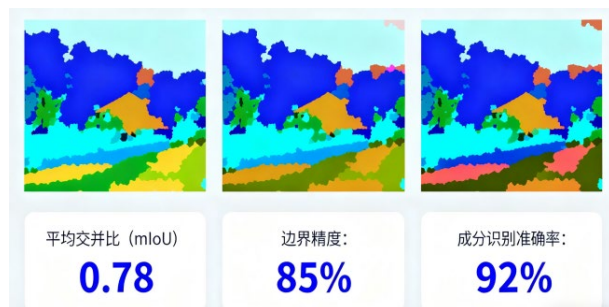


Figure 11. Performance indicator diagram of image segmentation model

图 11. 图像分割模型性能指标图

5. 实验评估

为验证本系统的性能, 将其与现有的一些食物识别和卡路里估算算法及设备进行了对比实验[10]。实

验选取了 50 种单一类别食物和 30 种组合菜品，在不同的环境条件下(包括强光、弱光、不同背景颜色)进行测试，每种食物和菜品测试 30 次，记录识别准确率和响应时间。

结果表明，本系统在识别准确率方面表现优异，对单一类别食物的识别准确率达到 90%以上，对组合菜品的识别准确率也在 85%以上，明显高于一些传统的手机应用和简单硬件设备。在响应速度上，从图像采集到显示结果的整个过程耗时不超过 2 秒，满足实时性需求。此外，本系统的功耗较低，在连续使用情况下续航时间可达 8 小时以上，便于用户携带和长期使用。

5.1. 教学场景设计

将本系统的硬件设备作为教具应用于教学场景，设计了一系列实践教学活 动，涵盖人工智能和嵌入式技术等多个知识点[11]。

在人工智能教学中，设计了以下教学活动：

硬件拆解与原理分析：组织学生拆解设备，识别 ESP32 开发板、摄像头、显示屏等硬件组件，了解各组件的功能和工作原理；通过查看硬件连接图，理解 SPI、I2C 等接口的通信方式，掌握嵌入式系统的硬件架构。

CNN 模型实践：引导学生查看 CNN 模型的训练代码(基于 TensorFlow 框架)，了解模型的结构(卷积层、池化层、全连接层)和训练过程(损失函数、优化器、迭代次数等)；提供简化的食物图像数据集，让学生尝试微调模型参数，观察模型识别准确率的变化，加深对深度学习、图像识别等算法的理解[12]。

在嵌入式技术教学中，设计了以下教学活动：

编程实践：基于 ESP32 开发平台，指导学生编写简单的控制程序，学习如何初始化摄像头并采集图像、如何通过 Wi-Fi 模块传输数据、如何驱动 OLED 显示屏显示信息等；通过调试程序，掌握嵌入式系统的软件开发流程和调试方法[13]。

系统集成与优化：让学生分组完成硬件组件的组装和软件程序的集成，测试系统功能并排查故障[14]；引导学生分析系统功耗瓶颈，尝试优化电源管理策略，提高设备续航能力，培养系统优化思维。

教学活动的设计框架如图 12 所示，图中清晰梳理了硬件拆解、CNN 模型实践、编程实践、系统集成与优化四大教学活动的具体内容与目标，为教学实践的开展提供了明确指导。

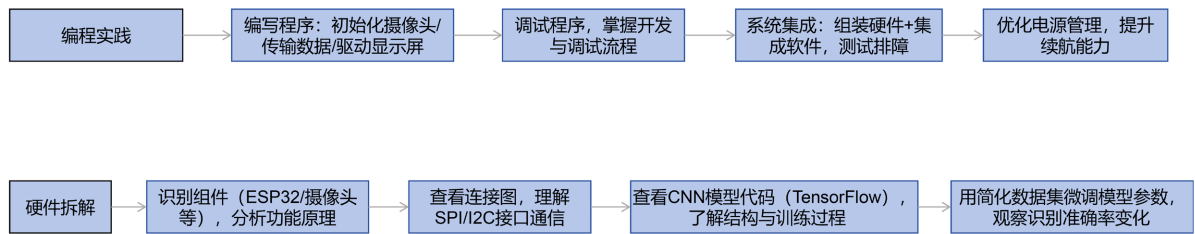


Figure 12. Schematic diagram of teaching activity design
图 12. 教学活动设计示意图

5.2. 学生交互与学习成效

通过组织学生参与上述教学活动，观察学生的交互过程和学习成效。参与教学的学生为本科学生或者高中生，共 40 人，分为 8 个小组，每组 5 人，进行为期 4 周的实践教学活 动。

学生在与硬件设备的互动中，表现出浓厚的学习兴趣和积极性。在硬件拆解环节，学生主动查阅各组件的数据手册，讨论接口定义和连接方式；在编程实践环节，通过调试代码解决图像采集失败、数据传输中断等问题[15]，不断尝试和优化，提升了问题解决能力。他们通过实际操作，能够更直观地理解抽

象的技术概念,将理论知识与实践应用相结合,例如在学习 CNN 模型时,通过观察模型对不同食物图像的识别结果,清晰认识到卷积层提取特征、全连接层分类的作用过程。

在学习成效方面,参与教学活动的学生在人工智能、嵌入式技术等相关知识的测试中,成绩较之前有明显提升。测试包含理论知识(如 CNN 模型原理、ESP32 接口功能)和实践操作(如编写摄像头控制程序、模型微调)两部分,学生的平均成绩从教学前的 65 分提升至 85 分。他们不仅掌握了相关的技术原理,还具备了一定的实践操作能力,能够独立完成简单的硬件开发(如组装简易图像采集设备)和算法调试任务(如调整模型参数提高识别准确率)[16]。同时,学生的创新思维和团队协作能力也在实践过程中得到了锻炼和提高,例如在系统优化环节,有小组提出通过动态调整图像分辨率来平衡识别精度和传输速度,展现了良好的创新意识;小组内部通过分工合作,高效完成任务,团队协作能力得到增强。

6. 项目创新点

6.1. 硬件轻量化与成本可控

本项目在硬件设计上强调轻量化与低成本特性,选用 ESP32 作为核心控制平台,并搭配低功耗摄像头模块(如 OV2640),有效避免了传统食物识别系统对 GPU 或高性能 PC 的依赖。ESP32 不仅体积小巧,而且具备 Wi-Fi 通信功能,能够在无需额外扩展模块的情况下完成数据传输与云端交互,从而显著降低了系统整体复杂度与硬件成本。同时,ESP32 的能耗水平较低,可支持长时间运行,适合便携式场景和教育实验环境,避免了频繁充电或高功耗带来的限制。与依赖高性能硬件的解决方案相比,本项目在成本、体积和能效上均表现出明显优势,使其更易于在中小学课堂、实验室教学以及大众健康管理领域推广。通过这种轻量化与成本可控的硬件设计,本系统不仅能够满足个人饮食管理的实际需求,还能够作为一种普及性较强的教育教具,为人工智能与健康管理相关知识的学习提供一个易于获取、低门槛的实践载体。

6.2. AI 与教育深度融合

本系统的一大创新点在于将人工智能应用与教育实践紧密结合。通过硬件平台与 CNN 模型的结合,学生不仅能够直观观察食物图像识别和卡路里计算的结果,还能够深入理解图像处理、深度学习模型训练与推理的基本原理[17]。具体体现在:

- (1) 为学生提供可操作性强的实验对象,能够在课堂中直观展示数据采集、模型优化、推理结果输出等完整流程;
- (2) 促进人工智能、嵌入式系统与营养学的跨学科融合[18],拓展中小学和高校教育的教学模式;
- (3) 通过互动式实验方式提升学生对 AI 的兴趣,培养面向未来的创新型人才。

这种“AI+ 教育”的融合不仅提升了项目的学术与教学价值,也为人工智能技术在教育场景中的落地提供了新的实践路径[19]。

6.3. 多功能教具化设计

在设计理念上,本项目强调多功能性与可教具化特征,使其既能满足实际的卡路里识别需求,又能作为教学实验工具使用。通过模块化设计,系统可以同时承担两类功能:一方面,作为日常饮食管理工具,帮助用户识别食物并估算卡路里,满足个人健康管理需求;另一方面,作为教学教具,它能够与课堂实验、课程讲解相结合,让学生在操作中学习嵌入式开发、图像处理与 AI 模型应用的知识。这种双重功能的实现,使得本项目在应用场景上更加灵活,可广泛适用于中小学 AI 科普教育、大学人工智能课程以及科研实验室的项目实践[20]。同时,其硬件平台具备良好的可扩展性,可以在后续教学应用中引入更

多传感器或功能模块,从而拓展实验内容与教学深度[21]。通过多功能教具化设计,本项目不仅解决了实际问题,还提升了教育价值,形成了“教学与应用并重”的创新模式。

7. 总结与展望

本项目基于 ESP32 嵌入式硬件平台与卷积神经网络(CNN),设计并实现了一种轻量化的卡路里识别相机。研究过程中,团队完成了硬件平台的搭建、图像数据采集与预处理、模型训练与优化、系统集成与测试等一系列工作。实验结果表明,该系统在硬件轻量化、低功耗运行以及食物识别与能量估算功能方面具有良好的可行性。项目的创新之处主要体现在三个方面:一是实现了硬件轻量化与成本控制,使得系统具备大规模普及的条件;二是推动了人工智能与教育的深度融合,为人工智能课程与健康教育提供了实践载体;三是通过多功能教具化设计,兼顾了教学与实际应用需求,拓展了项目的适用场景。整体而言,本研究为嵌入式人工智能在健康管理和教育领域的应用提供了有益的探索。

尽管本项目取得了初步成果,但仍存在一定不足,未来可在以下方向展开进一步研究:在模型层面,需要持续提升识别精度与鲁棒性,特别是针对混合菜品与复杂光照条件下的图像识别问题,可引入更先进的图像分割与多模态学习方法;在系统性能方面,有必要进一步优化边缘计算与云端协同机制,提升实时性与稳定性,同时探索更高效的数据压缩与传输策略,以适应更广泛的应用场景;在教育应用层面,可扩展实验功能模块,结合开源工具与教学平台,开发适合不同教育阶段的实验课程与互动内容;在推广应用方面,则可将系统拓展至校园营养管理、智能厨房与医疗辅助等领域,形成更具产业化与社会化价值的解决方案。本项目在嵌入式人工智能结合的研究中展现了独特优势,并在健康管理与教育融合应用方面具有广阔的发展空间。

参考文献

- [1] 许广文,赵炎,王艳霞,等.基于 ESP32 的负荷测量仪的设计与应用[J].工业仪表与自动化装置,2025(3): 60-64+74.
- [2] 尹文博,李丹,钟昆,等.基于 ESP32 的家庭智能语音控制系统[J].电脑知识与技术,2022,18(21): 67-69+81.
- [3] 宋凯,严丽平,甘岚.嵌入式图像处理系统的设计与实现[J].计算机工程与设计,2009,30(19): 4368-4370+4377.
- [4] 杨铮,贺晓武,吴家行,等.面向实时视频流分析的边缘计算技术[J].中国科学:信息科学,2022,52(1): 1-53.
- [5] 汪聪.基于机器视觉的菜品智能识别技术研究[D]:[硕士学位论文].广州:华南理工大学,2019.
- [6] 董天骄.基于卷积神经网络的饮食分类与识别[D]:[硕士学位论文].杭州:杭州电子科技大学,2018.
- [7] 周莉莉,姜枫.图像分割方法综述研究[J].计算机应用研究,2017,34(7): 1921-1928.
- [8] 姚伟盛,沈宇帆,彭玉波,等.基于自我监督预处理的食物图像分类[J].智能计算机与应用,2021,11(3): 9-15.
- [9] 张舰舰.一种新的高效轻量级卷积神经网络模型[J].计算机与现代化,2021(4): 98-103.
- [10] 梁炎超,李建华.基于深度学习的食物卡路里估算方法[J].华东理工大学学报(自然科学版),2018,44(2): 270-276.
- [11] 张诗芸,杨琳玲,郭子叶.基于设计思维的项目式学习教学案例——以“燃烧我的卡路里”项目为例[J].中小学信息技术教育,2023(Z1): 94-96.
- [12] 刘焕军,王耀南,段峰.机器视觉中的图像采集技术[J].电脑与信息技术,2003(1): 18-21.
- [13] 梁海军.嵌入式图像采集与处理系统实现技术研究[D]:[硕士学位论文].西安:西安电子科技大学,2009.
- [14] 赵忠旭,沈兰荪,王爱民.视频图像采集系统的设计与实现[J].测控技术,1998(6): 39-42.
- [15] 龚沛曾,杨志强.大学计算机基础教学中的计算思维培养[J].中国大学教学,2012(5): 51-54.
- [16] 刘清堂,何皓怡,吴林静,等.基于人工智能的课堂教学行为分析方法及其应用[J].中国电化教育,2019(9): 13-21.
- [17] 刘德建,杜静,姜男,等.人工智能融入学校教育的发展趋势[J].开放教育研究,2018,24(4): 33-42.
- [18] 陆启威.学科融合不是简单的跨学科教育[J].教学与管理,2016(32): 22-23.

-
- [19] 刘仲林. 交叉科学与交叉(跨学科)教育[J]. 天津师大学报, 1986(4): 29-35.
- [20] 郭叶铭. 浅谈青少年人工智能科普活动的实践和探索——基于浙江省青少年人工智能科普活动分析[J]. 科技通报, 2019, 35(8): 231-235.
- [21] 郑妍, 周倩, 王惠欣. 基础教育阶段人工智能课程的有效教学模式探析[J]. 中国教育信息化, 2019(16): 10-14.