# 充电口识别系统V1.0

# -设计说明书

目 录

[充电口识别系统V1.0 1](#_Toc164248666)

[-设计说明书 1](#_Toc164248667)

[1 引言 3](#_Toc164248668)

[1.1 编写目的 3](#_Toc164248669)

[1.2 使用对象 3](#_Toc164248670)

[1.3 开发背景 3](#_Toc164248671)

[2 系统概述 3](#_Toc164248672)

[2.1 系统开发环境 3](#_Toc164248673)

[2.2 系统运行环境 4](#_Toc164248674)

[2.2 系统功能概述 4](#_Toc164248675)

[2.3 系统功能要求 4](#_Toc164248676)

[3 软件设计 4](#_Toc164248677)

[3.1 软件需求分析 4](#_Toc164248678)

[3.2 软件设计概要 5](#_Toc164248679)

[3.3 软件详细设计说明 6](#_Toc164248680)

## 1 引言

### 1.1 编写目的

本文档为充电口识别系统 V1.0设计说明文档，为软件系统的设计与使用提供信息基础，便于开发人员及使用者快速了解本系统的基本信息。

### 1.2 使用对象

本文档的使用对象主要为系统测试与使用人员。

### 1.3 开发背景

当今社会的高级便携式识别模式大多基于yolo模型，在性能上有一定的提高，但需要占用较大的内存，让识别设备的二次处理受到极大的限制。不适用于小内存的嵌入式设备。而对于智能充电桩的小体积、智能化而言，大容量的识别无疑极高的提高了成本，不利于大范围集群化的部署，且效率得不到充分的利用。因此，开发一款能够运行在小内存设备上高效运行的识别模式显得尤为重要。

该系统的开发旨在通过FOMO神经网络的小型模型与openmv二次图像开发的结合，实现充电口的高效识别与轻量级部署。通过这种方式，可以极大的提高识别效率并减少部署成本，提高运行效率。

## 2 系统概述

### 2.1 系统开发环境

1）硬件环境

最低要求：

处理器： STM32H7或同级处理器

内存： 4GB RAM

建议配置：

处理器： STM32H7或同级以上处理器

内存： 16GB RAM

2）软件环境

编程语言： Python

集成开发环境：OpenMV IDE

操作系统：Windows 10或更高版本，macOS 10.14或更高版本

### 2.2 系统运行环境

1）硬件环境

MCU：STM32H750XBH6

SDRAM：32-Mbytes

板载支持：I2C、SPI、UART等通讯协议

2）软件环境

Windows操作系统

### 2.2 系统功能概述

本系统运用基于stm32f4主控的OpenMV作为识别主体，镜头采用ov5640。通过嵌入基于FOMO模型的神经网络对目标物的特征进行扫描识别并返回图像坐标信息，借助VL53L1X测距模块与小孔成像原理进行图像坐标换算。并通过串口实时传递坐标数据包、接收主控信号信息。

### 2.3 系统功能要求

1）通过串口协议使系统与主控通信，接收相关指令、传递坐标信息。

2）通过小孔成像原理与测距模块将图像坐标转换为地理坐标。

3）通过FOMO神经网络进行充电口的识别与坐标返回

## 3 软件设计

### 3.1 软件需求分析

1) 实现串口协议与主控通信模块：

a）该系统应支持串口协议，以便与主控进行通信。

b）能够接收来自主控的相关指令，并进行相应的处理。

2） 实现图像坐标转地理坐标算法模块：

a）该软件系统需要包含小孔成像算法，用于将识别到的图像坐标数据转换为地理坐标数据。

b）算法需具备准确性和高效性保正计算的坐标正确。

3）实现充电口智能识别模块：

a）该软件系统需FOMO神将网络模型，以便于充电口的识别智能化与高效化。

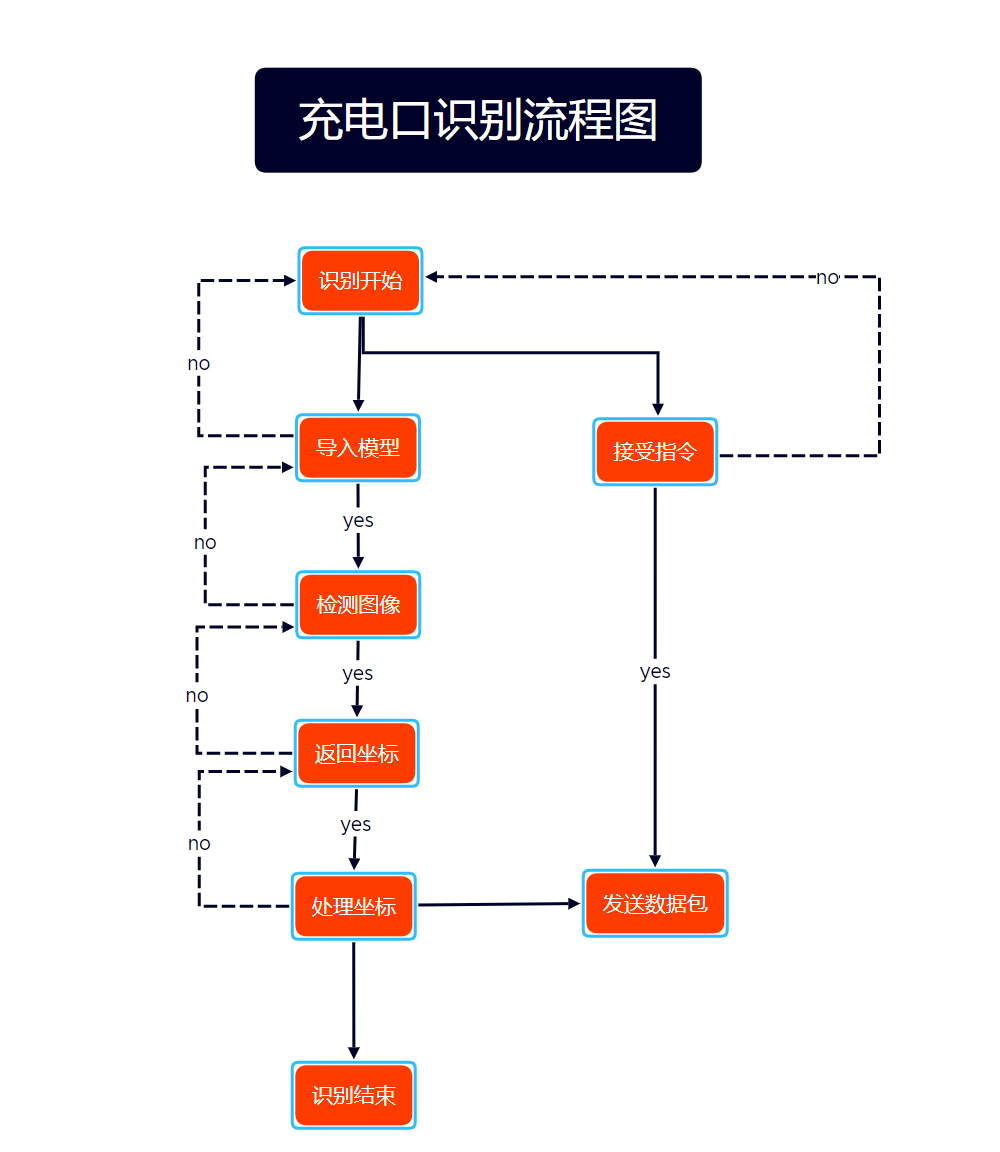
5）该软件系统应具有稳定性和可靠性，能够长时间运行。

6）该软件需要充分考虑通信模块的实时性和数据的准确性。

7）该软件应易于维护和扩展，方便进行后续功能的添加和修改。

### 3.2 软件设计概要

结合充电口识别系统的实际功能需求，设计如下软件系统。如图所示为该软件系统的流程图。



通信模块负责系统与主控之间的数据交换。串口通信模块利用串口通信协议与主控建立连接。在接受到发送指令后通过数据包的形式发送坐标信息给主控。

坐标二级处理模块根据神经网络传递的坐标信息结合测距模块返回的实时距离进行小孔成像变换，将图像坐标信息转换成地理坐标信息，并通过通信模块传递给主控。

神经网络模块包括特征点检测算法和轻量级部署算法。特征点检测模块对接收到的图像数据进行相应特征点的检测，并进行正误分析，使识别精度提高，同时，传递相应图像坐标返回二级处理算法中。轻量级部署算法模块内嵌于FOMO神经网络中，实时检测部署情况，并进行自适应调整。

### 3.3 软件详细设计说明

1）串口通信模块设计说明：

\*\*功能描述：\*\*

该模块可建立与主控的串口连接，并接收由主控发送的发送指令与暂停指令，以启动系统的相应功能。

\*\*具体实现：\*\*

在OpenMV中使用串口库。首先配置必要的连接参数，包括串口号、波特率、校验位、停止位，以确保与主控的安全、稳定连接。

接下来，定义数据包函数，便于串口发送坐标信息。随后定义通信主体代码，该模块将持续监听来主控的消息，并在接收到发送指令时触发相应的系统功能。

\*\*代码接口设计：

\*\*功能描述：配置与主控通信的连接参数。

- 输入：串口号、波特率、比特位、校验位、停止位

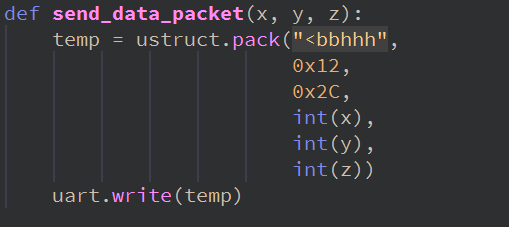
- 输出：无



\*\* 功能描述：定义发送包函数

- 输入：坐标信息

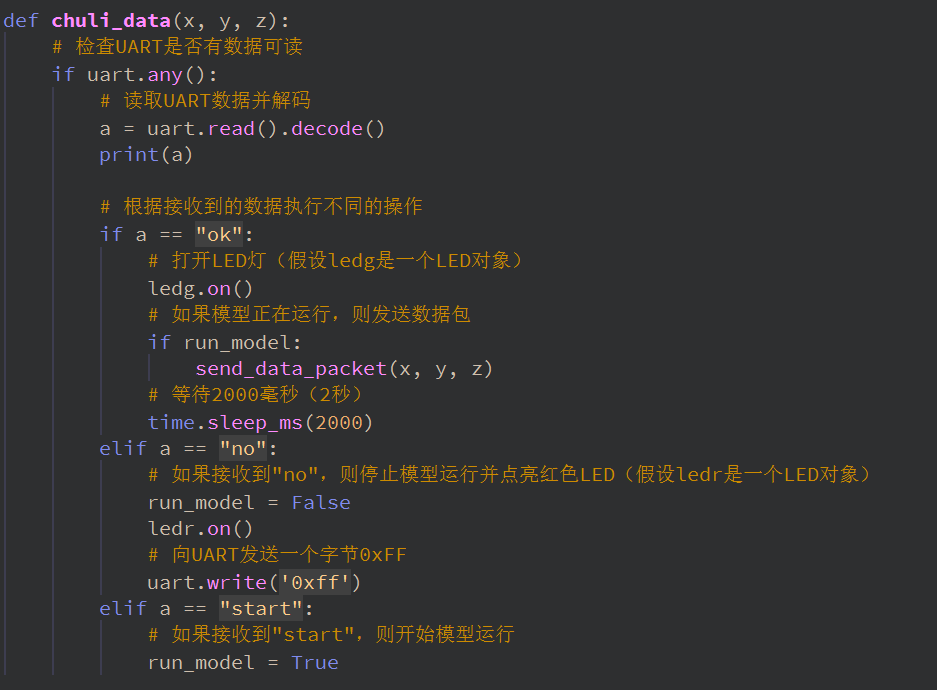
- 输出：串口发送函数



\*\* 功能描述：通信主体代码，实时监控主控信息

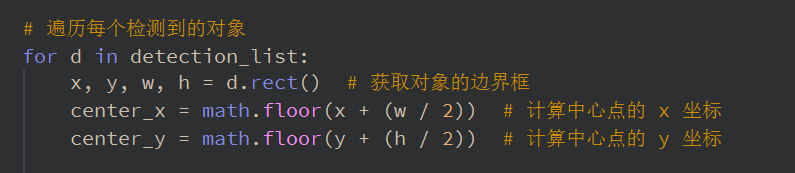
- 输入：坐标信息

- 输出：无



2）坐标二级处理模块设计说明：

初始图像坐标定义：



二级处理模块定义：

\*\*距离检测算法：

功能描述：该算法用于获取实时的距离信息

具体实现：配置VL53L1X库函数，调动tof测距模块获取实时数据。







\*\*小孔成像算法：

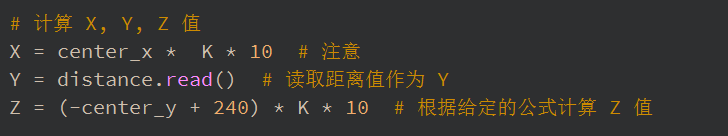
功能描述：该算法负责将得到的图像坐标信息通过小孔成像原理与实时监测到的距离结合换算出实际的地理坐标信息

具体实现：通过各比例系数与原坐标信息运算得到

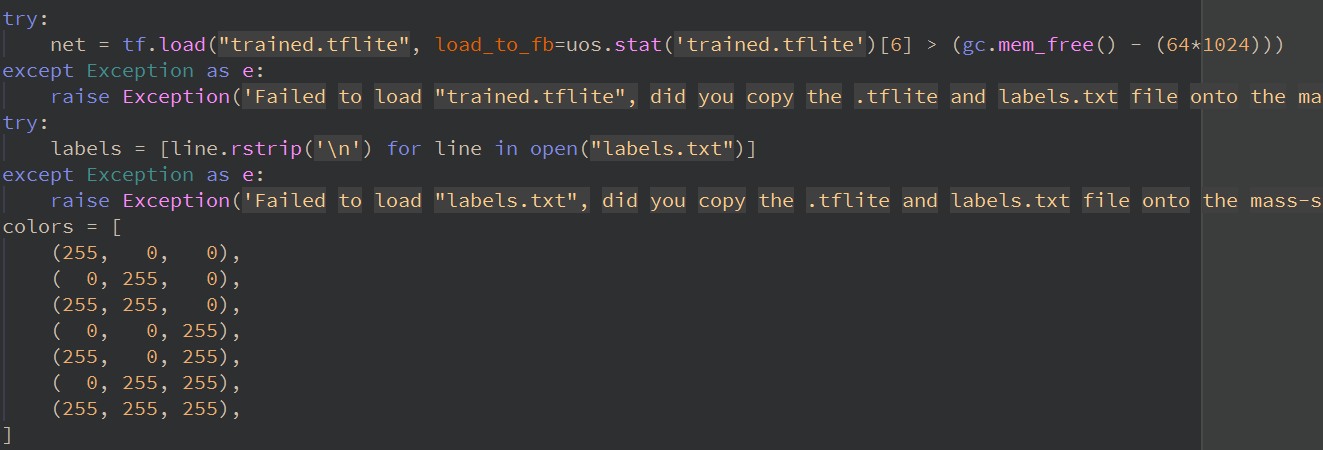
输入：图像坐标信息

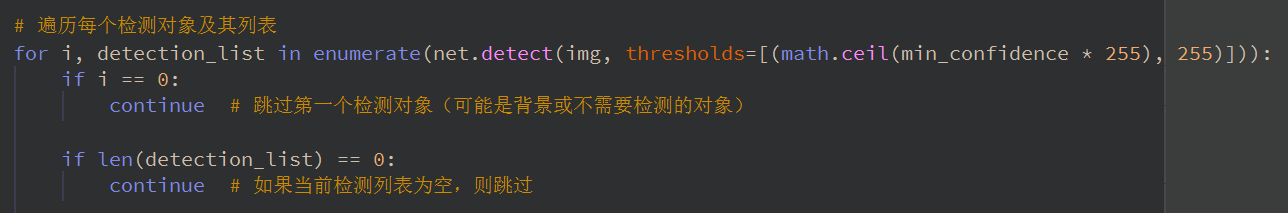
输出：地理坐标信息





3） 神经网络模块





特征点模块设计说明：

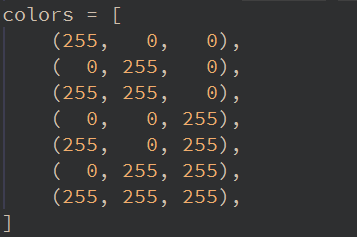
功能描述：该模块实现图像的特殊点检测，用于识别并返回坐标

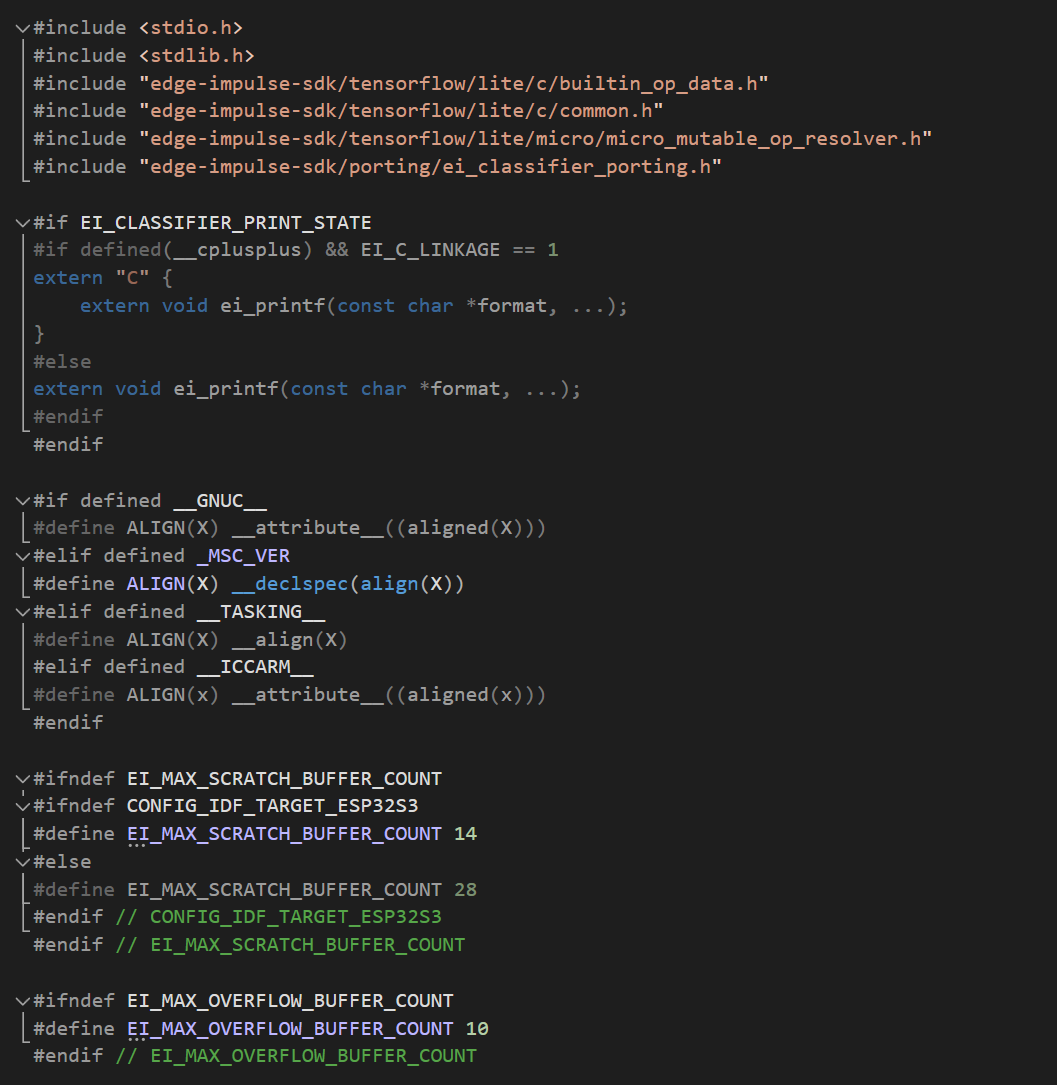
具体实现：首先选定图像的几个特定点模拟特殊值，如颜色、形状等，再进行模型的数据集训练提高识别精度，最后部署到嵌入式设备中。

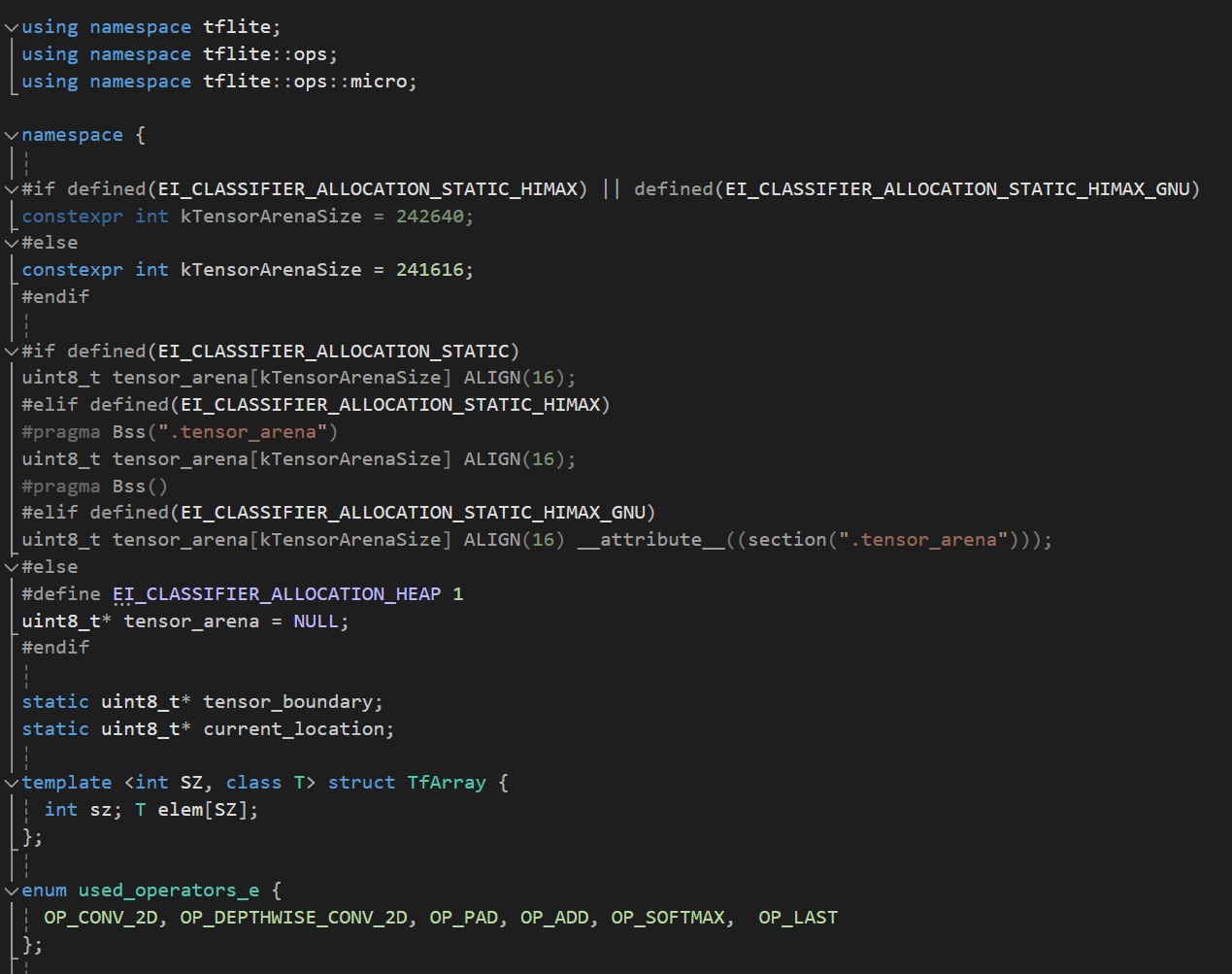
接口设计：包括特征点获取、数据训练结果等接口

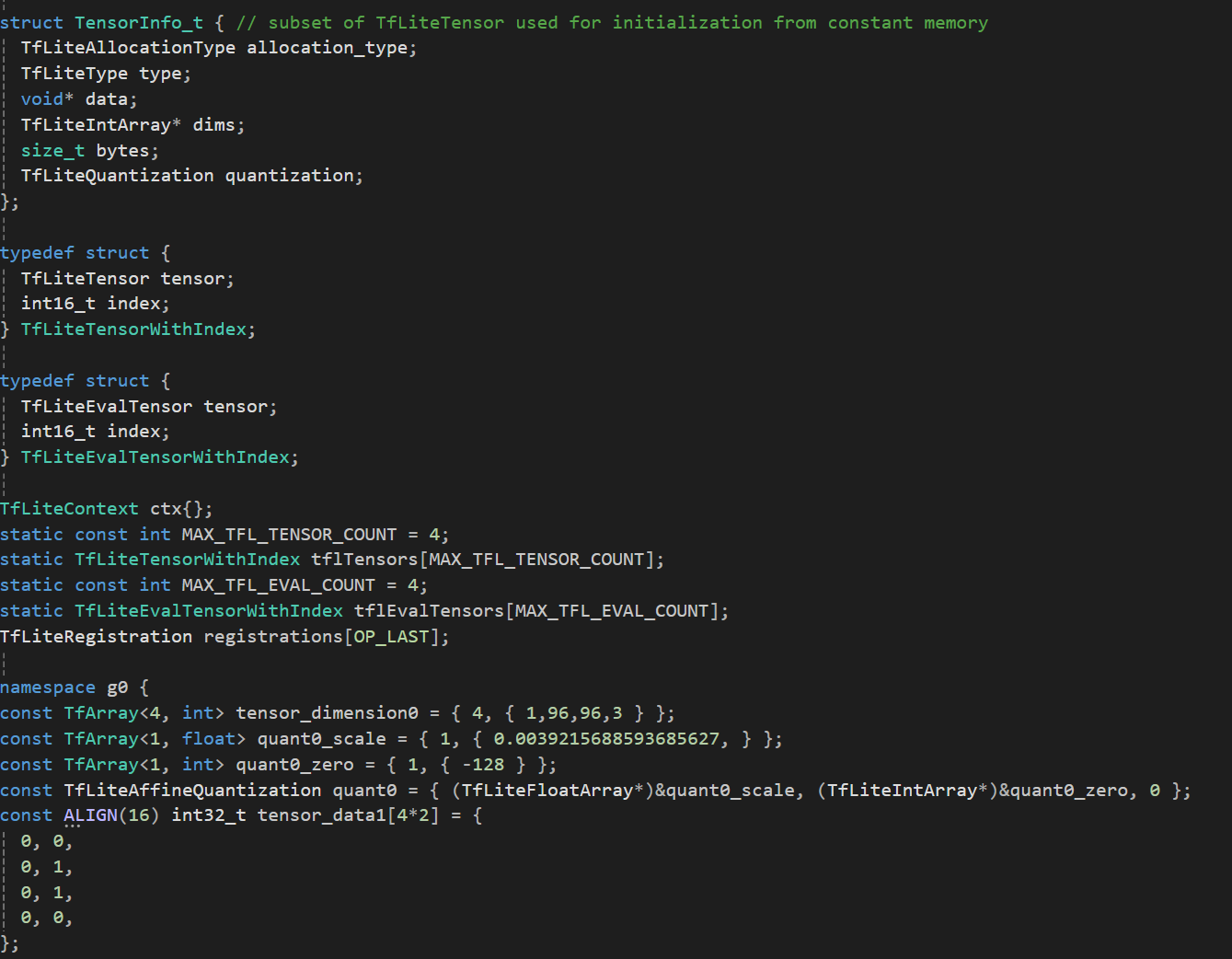
输入：图像

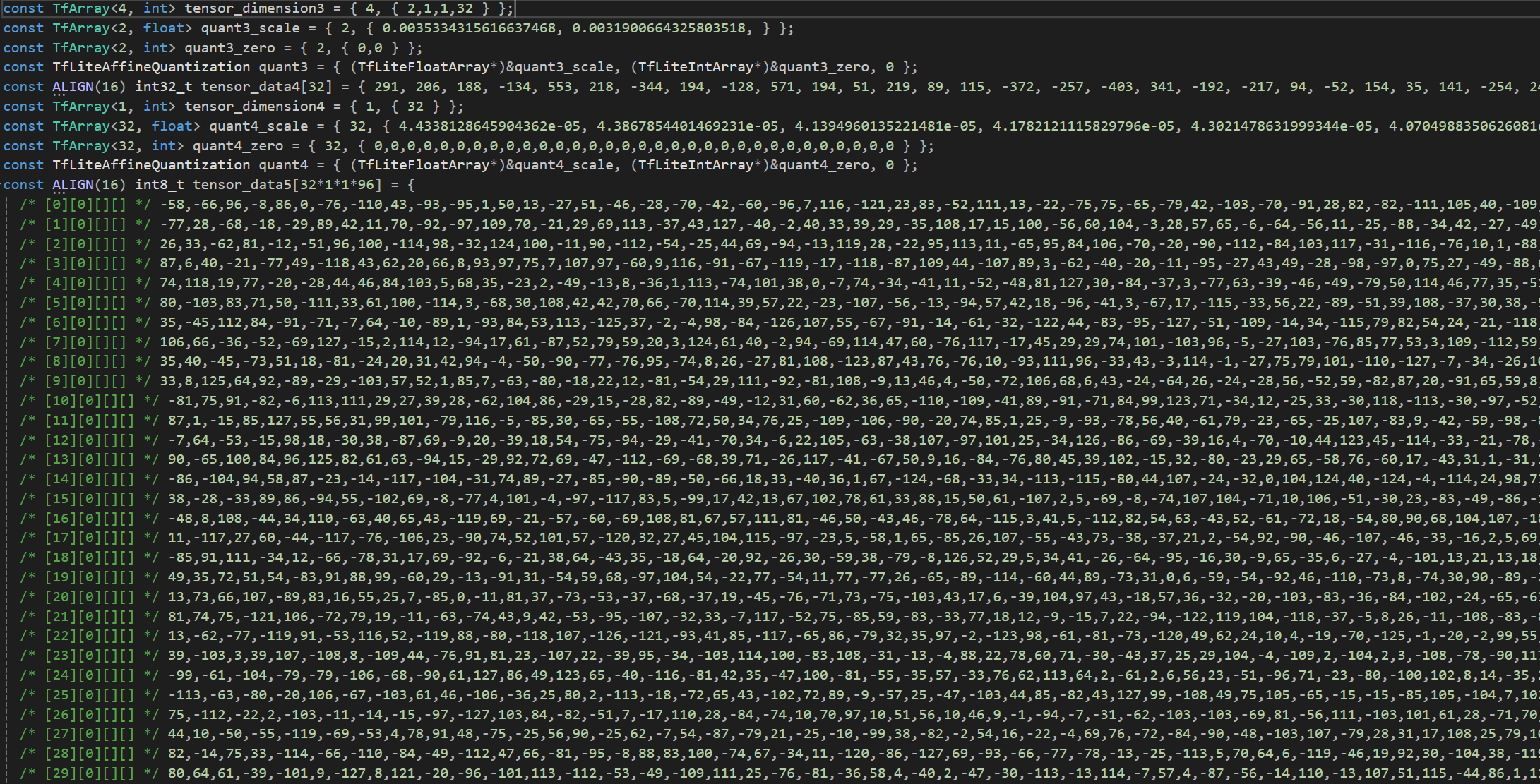
输出：模型参数

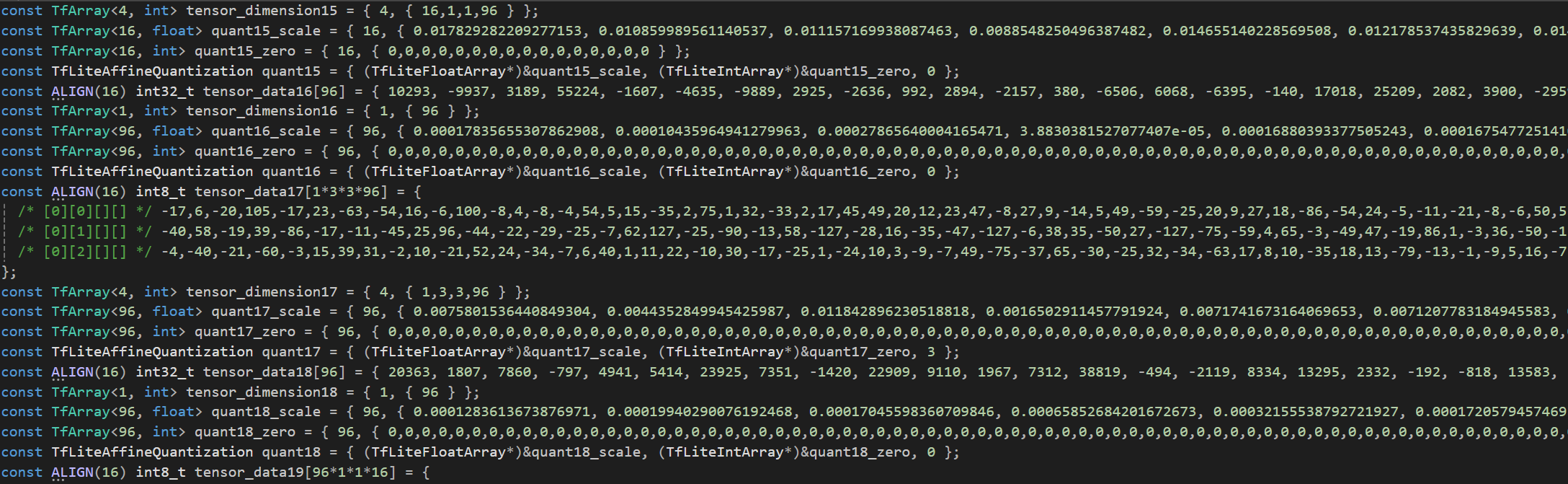


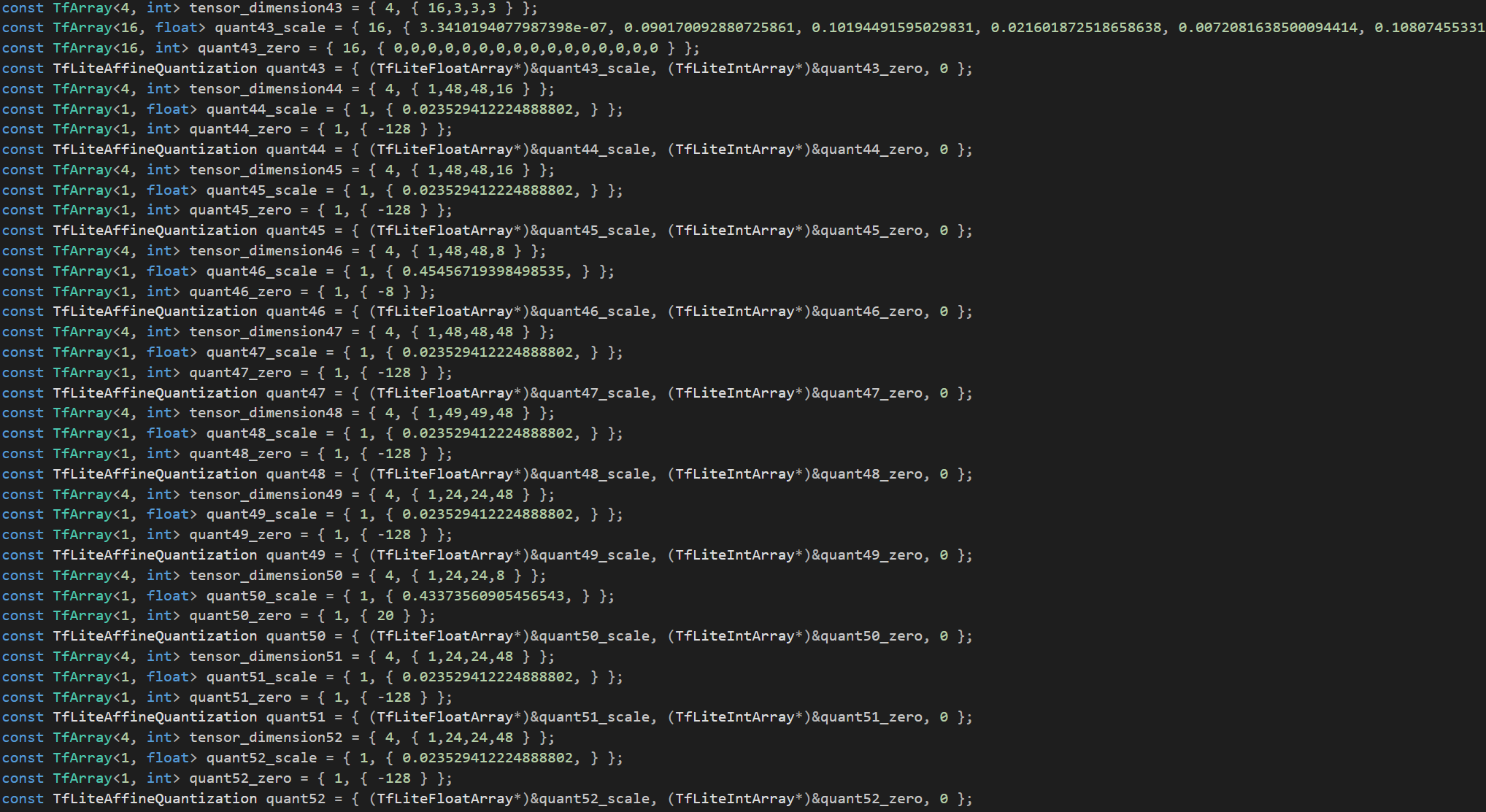


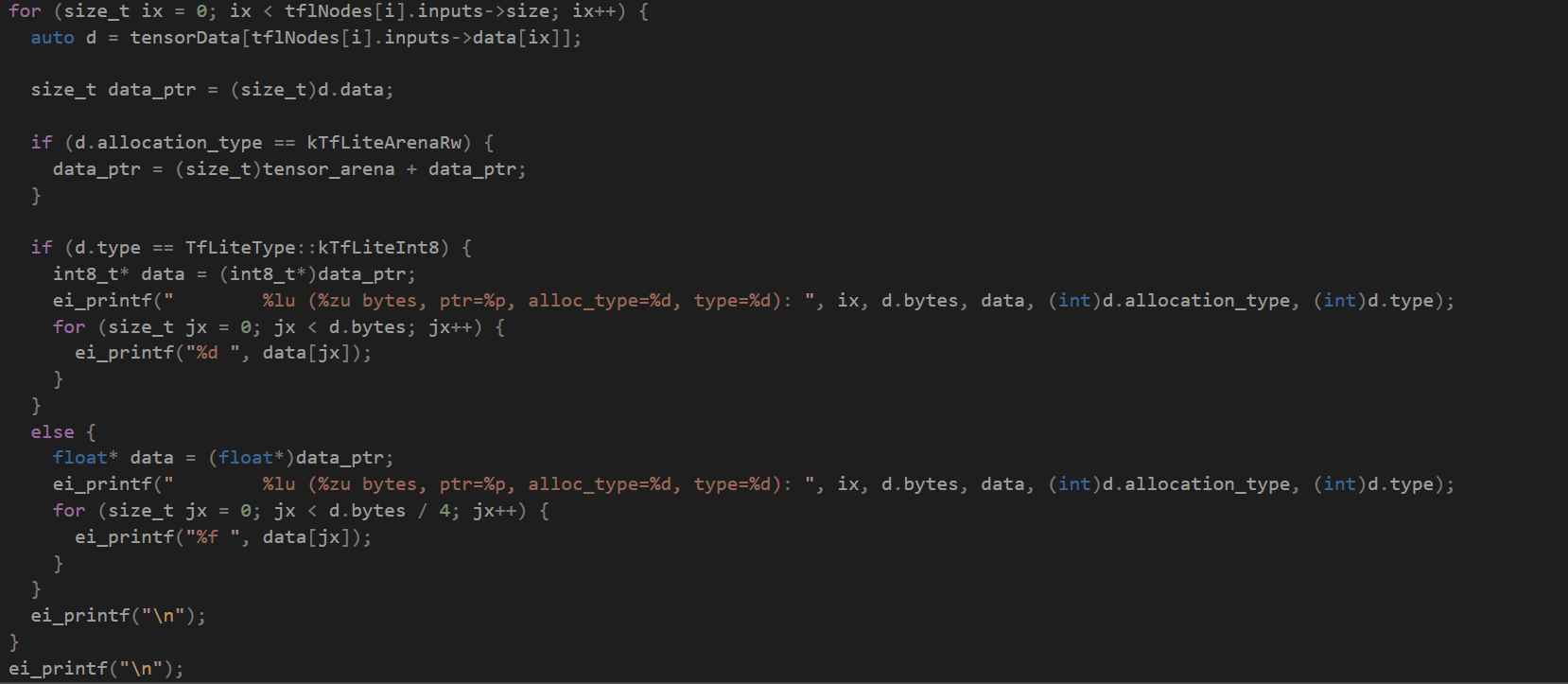


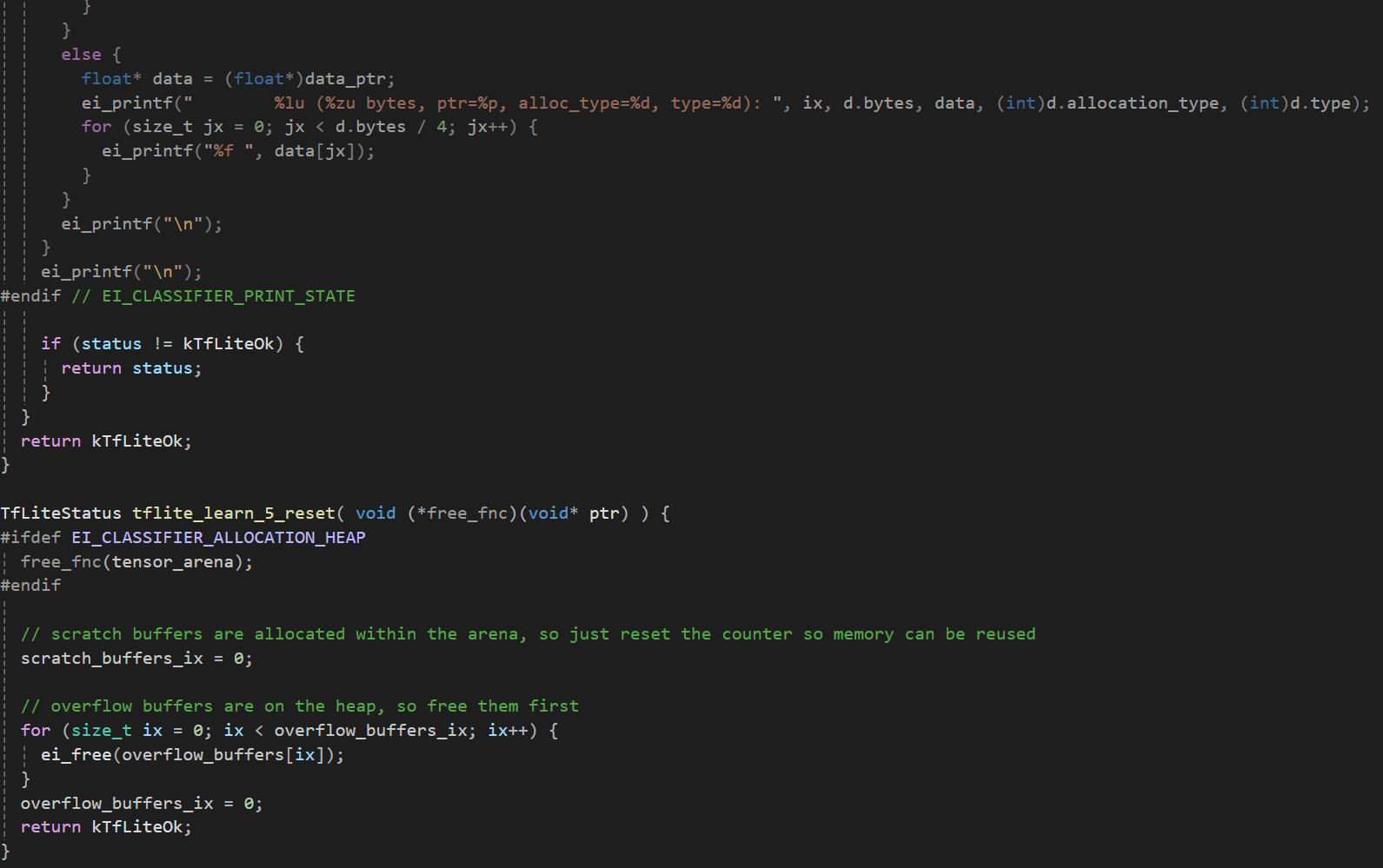


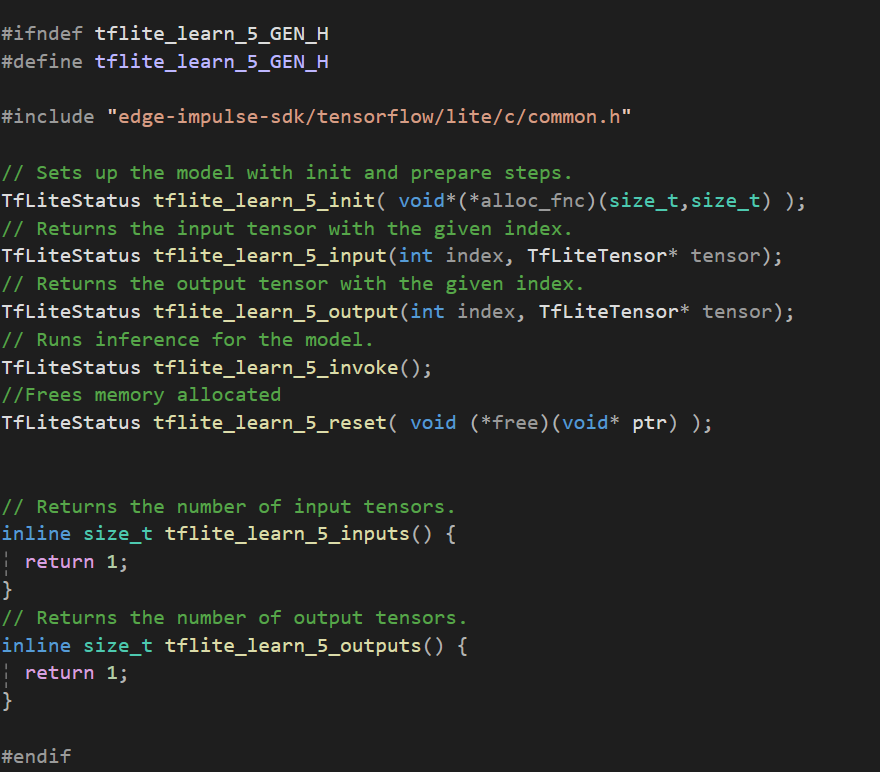












轻量级部署算法模块设计说明：

功能描述：该算法用于模型训练时提高模型的准确度

具体实现：通过对模型的数据集进行抽样检测，实时矫正训练结果

接口设计：包括数据参数等接口。

输入：数据信息

输出：矫正量。

