

边缘计算在病理图像识别方向的应用

孙涛¹, 王书浩², 王伟²

1. 北京交通大学计算机与信息技术学院, 北京 100080; 2. 北京透彻未来科技有限公司透彻实验室, 北京 100080

【摘要】病理图像是医学图像的重要组成部分,对于癌症等疾病的早期诊断至关重要。本文提出一种对医学病理图像进行识别分析的边缘推理解决方案,将边缘计算引入病理图像的研究方向,旨在利用边缘计算的优势,结合深度学习方法,实现实时的病理图像检测。实验结果表明,此方案的应用可以极大地保护患者隐私、提高病例图像处理的效率和实时性,并节省成本,具有很高的实际应用价值,可为医学诊断和治疗提供更快速和更准确的支持。

【关键词】边缘计算;病理图像;深度学习

【中图分类号】R318;TP3-05

【文献标志码】A

【文章编号】1005-202X(2025)03-0328-08

Application of edge computing in pathological image recognition

SUN Tao¹, WANG Shuhao², WANG Wei²

1. School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100080, China; 2. Thorough Lab, Beijing Thorough Future Technology Co., Ltd, Beijing 100080, China

Abstract: Pathology images, being an essential part of medical imaging, are critical for early diagnosis of diseases such as cancer. The study introduces edge computing into pathology image research, and presents an edge inference solution for medical pathological image recognition and analysis, aiming to leverage the advantages of edge computing and combine with deep learning methods to achieve real-time pathology image detection. Experimental results show that the solution protects patient privacy well, improves the efficiency and real time of medical image processing, and greatly reduces costs, demonstrating that it has high practical application value and can provide more and efficient accurate support for medical diagnosis and treatment.

Keywords: edge computing; pathological image; deep learning

前言

病理图像作为医学诊断中的重要组成部分,承载着非常丰富的生物学信息,对于疾病的诊断、治疗和预后评估具有至关重要的意义。通过对组织结构和细胞形态的观察和分析,医生可以快速识别疾病的类型、病变的程度和组织的状态,从而指导临床决策并辅助制定个性化的治疗方案。然而,传统的病理图像识别往往依赖于医生的经验和专业知识,存在诊断效率低^[1-2]、主观性强等问题,尤其是在面对大量、复杂的病理图像数据时,容易导致诊断结果的不一致性和误差。

近些年来,随着深度学习技术的快速发展和广

泛应用,特别是卷积神经网络(Convolutional Neural Network, CNN)的兴起,在包括医学图像分析在内的各个领域取得了巨大的成功^[3-5]。深度学习模型通过大量的病理图像数据进行训练,可以自动学习和提取图像中的特征,实现对病变区域的准确定位和分割^[6-7],结合云端计算模型可以为医生提供可靠的辅助诊断工具^[8]。传统的云端计算模式虽然可以在云端服务器上进行病理图像的分析计算,但也面临着隐私安全、延迟高、数据传输成本高等问题^[9],限制了其在实时诊断和移动医疗领域的应用。这是因为云端计算需要将大量的病理图像数据从边缘设备传输到云服务器进行处理,导致了较长的数据传输延迟和高额的服务器机房维护成本。同时,由于病理图像可能包含敏感的个人健康信息,云端计算需要确保数据传输和处理过程中的隐私安全,增加了额外的隐私保护措施和成本。

为了解决这些问题,边缘计算技术应运而生,作为一种新兴的计算模式^[10],具有去中心化、数据本地

【收稿日期】2024-10-11

【基金项目】国家自然科学基金(61901436)

【作者简介】孙涛,硕士研究生,主要研究方向:计算机视觉算法,
E-mail: 243825727@qq.com

化、低延时等特点^[11-12],将计算任务推送到离数据源更近的边缘设备上进行处理和分析,能够实现实时的数据处理和响应,大大降低了数据传输的延迟和成本^[13],同时保护患者的隐私安全。本研究将边缘计算引入病理图像识别领域,提出一种新的病理图像识别边缘计算解决方案,结合深度学习技术,实现对病理图像的实时计算和分析,为医学诊断提供更安全、更快速、更可靠的支持,为医学图像诊断领域的发展和创新提供新的思路和方法^[14]。

1 方案设计

病理图像识别可以选择云计算的方法^[15]。然而在云计算模式下,存在患者数据隐私和安全问题,并且网络传输可能导致较高的延迟,影响实时性。此外,维护云计算资源也会导致较高的运营成本。针对上述问题,边缘计算模式是一种很好的解决方案。本文提出一种新的病理图像识别边缘计算解决方案,结合深度学习方法,将深度网络模型部署在边缘设备上,旨在实现安全、高效、低成本的病理图像分析。实验结果显示,该方案在实时诊断和移动医疗领域具有很高的应用价值,具有广泛的适用性和可扩展性。

1.1 病理图像采集端

该端用于病理图像的采集,通过将电子显微镜与计算机相连接,实现对病理组织样本的图像采集,并将采集到的图像传输到计算机端进行图像预处理。从推理端获得病理图像推理结果后,还可以进行可视化等后处理操作,供医学研究人员进行分析研究。

1.2 边缘计算推理端

该端用于病理图像的推理,具备推理所需的计算能力和存储资源。作为病理图像识别推理的计算平台,在设备上部署经过训练的深度学习模型,例如细胞检测和计数、病理分类、器官和病变检测等,负责对接收到的病理图像进行识别推理任务,并将结果返回给图像采集端的计算机。

1.3 整体方案架构

如图1所示,首先使用电子显微镜对病理标本进行高分辨率的图像采集。采集到的图像通过连接到计算机的电子显微镜系统传输到计算机,然后在计算机端的应用程序进行图像预处理操作,例如图像亮度调整、选择感兴趣区域等。接下来,计算机与推理端边缘设备建立TCP/IP网络协议连接,用于数据的通信传输。当计算机接收到来自于电子显微镜的病理图像数据后,通过TCP/IP协议将图像数据传输至推理端,在推理端利用已经部署好的模型进行推

理操作。在推理完成后,将病理图像的识别推理结果返回给计算机,实现双向通信传输。最后,计算机端进行推理结果解析与可视化等后处理步骤,以供医学研究人员参考。

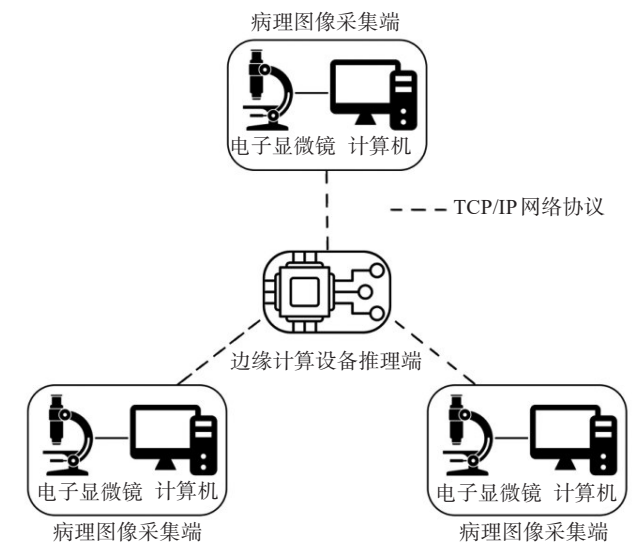


图1 病理图像识别边缘计算解决方案
Figure 1 Edge computing solution for pathological image recognition

推理端与采集端为一对多关系,即一个推理端的边缘设备可以同时供多个采集端进行推理。通过以上步骤,实现一种基于边缘计算的病理图像识别推理解决方案,将病理图像的采集和识别任务分布在不同的设备上,实现计算任务的协同处理和资源的有效利用。

2 实验过程

2.1 边缘设备

本文实验使用英伟达 Jetson Orin Nano 8 GB 嵌入式开发套件作为硬件设备(图2)。此设备具有高性能的GPU和CPU,适用于边缘计算场景下的深度学习推理任务。此设备CPU搭载6核心ARM Cortex-A78AE处理器,GPU基于NVIDIA Ampere架构,具有1 024个CUDA(Compute Unified Device Architecture)核心和32个用于AI处理的张量核心,每秒可进行40万亿次浮点运算(40 TOPS),支持FP16和INT8精度数据的运算^[16-17]。

2.2 深度学习模型部署方法

TensorRT是用于高性能深度学习推理的加速库^[18-19],通过优化网络结构和计算流程,提高模型在GPU等硬件设备上的推理效率。利用TensorRT对神经网络算法模型进行加速,尤其适用于在边缘设备



图2 Jetson Orin Nano 产品示意图

Figure 2 Jetson Orin Nano developer kit

上部署和运行深度学习模型。TensorRT 主要有以下几种优化方法:层间融合或张量融合、数据精度校准和量化、内核自动调整、动态张量显存^[20-21]。

PyTorch 模型转换方法:(1)训练模型。根据任务需求,选择或设计深度神经网络,使用PyTorch训练框架搭建模型、训练模型以及评估模型。(2)模型导出。将训练好的模型导出为ONNX(Open Neural Network Exchange)格式。ONNX是一种开放的深度学习模型表示格式,可在不同的深度学习框架之间进行模型的转换和交互。(3)引擎序列化。将构建好的推理引擎序列化为文件,以便在部署和推理时直接加载使用。序列化后的引擎文件包含了优化后的模型结构和参数,以及TensorRT所需的配置信息。(4)模型部署。将序列化的TensorRT引擎文件加载到目标应用程序中,用于实际的推理任务。在部署过程中,根据具体需求进行推理输入和输出的处理、批处理管理等。

通过以上步骤将PyTorch模型转换为可用于TensorRT的模型,并利用TensorRT的优化策略和高性能推理引擎实现模型的加速和部署。

2.3 CoNIC2022数据集

CoNIC2022数据集是一个用于医学图像分析的开放数据集,包含标注过的苏木精-伊红染色(H&E)图像^[22],示例图像如图3所示,主要用于细胞核检测和实例分割任务^[23],将每个细胞核分为以下类别:上皮细胞、淋巴细胞、血浆、嗜酸性粒细胞、中性粒细胞或结缔组织。

2.4 模型训练与转换

在本实验中,笔者选择了基于YOLOv8的实例分割模型架构,具体选用了X配置版本(Xlarge),并使用CoNIC2022数据集,对病理图像中的细胞核进行实例分割识别。使用开源YOLOv8代码进行模型训练。

首先在GPU上使用PyTorch框架进行模型训练,

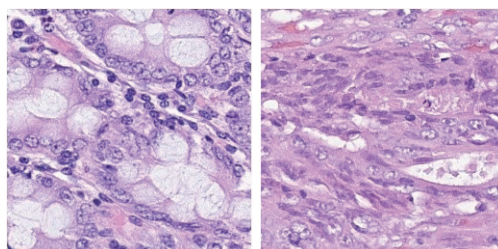


图3 CoNIC2022图像数据集中两个示例图像

Figure 3 Two example images from the CoNIC2022 pathological image dataset

然后使用C++语言编程和TensorRT相关工具,将PyTorch模型转换为TensorRT模型,以实现在Jetson Orin Nano上的部署推理。

在训练过程中,使用交叉熵损失函数和Adam优化器,学习率设置为0.001,动量设置为0.9,批量大小设置为16,进行了200个训练轮次。训练过程的损失值和相关评估指标如图4和图5所示。

2.5 输入和输出配置

在实验中,对病理图像进行预处理,将其调整为640×640的尺寸。使用640×640的图像作为输入数据,批量大小设置为1。输出数据包含预测框、置信度、类别以及分割图掩码(mask)。将经过预处理后的输入数据提供给TensorRT执行推理操作,获取输出结果,可以得到每个输入图像的预测框位置、对应的置信度和类别,以及针对每个目标的分割图掩码。通过这些输出结果,可以进行后续的可视化分析和应用。

2.6 实验设计

2.6.1 模型推理准确度 选择3种不同的量化精度配置,对服务器中的PyTorch模型和边缘设备上的TensorRT模型的准确度进行评估和对比分析,在服务器使用PyTorch默认的FP32权重精度,在边缘设备Jetson Orin Nano上使用FP16和INT8量化精度,对比3种不同权重精度下模型推理的结果,以评估边缘计算模式下模型在实际应用中的效果。

2.6.2 推理时间测试 记录TCP/IP协议从采集端计算机发送数据至推理完成并返回结果所需的时间,并且记录基于YOLOv8实例分割的5个不同规模模型以及每个量化精度配置下的推理时间,比较不同精度对推理性能的影响,以评估边缘计算模式下模型推理的实时性。

2.6.3 推理性能评估 在进行推理过程中设置循环持续推理,记录设备所需的内存、显存占用情况、功率消耗以及设备温度等指标。利用这些指标评估模型在实际硬件上的性能表现,包括资源利用情况和能效表现。特别关注推理持续时间较长(例如2 h)后的

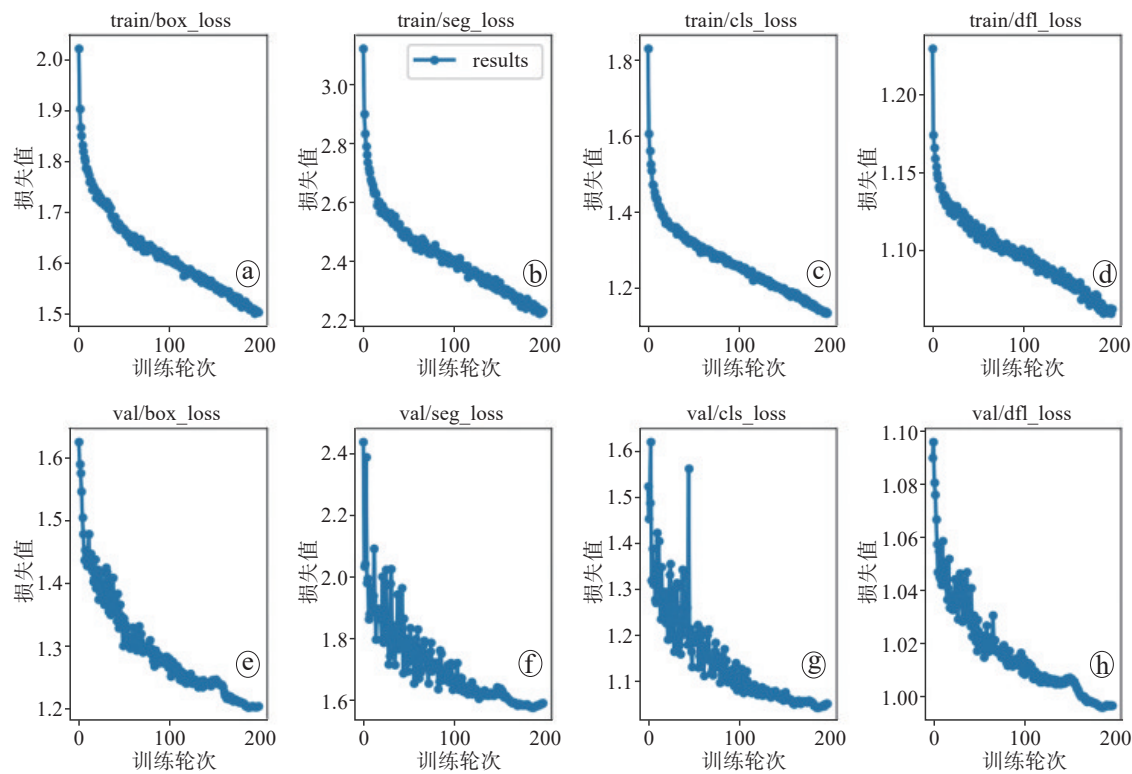


图4 模型训练集和验证集损失值

Figure 4 Training and validation loss values of the model

图a~d分别为训练集的边界框预测损失、实例分割损失、分类损失、分布式聚合损失;图e~h分别为验证集的边界框预测损失、实例分割损失、分类损失、分布式聚合损失

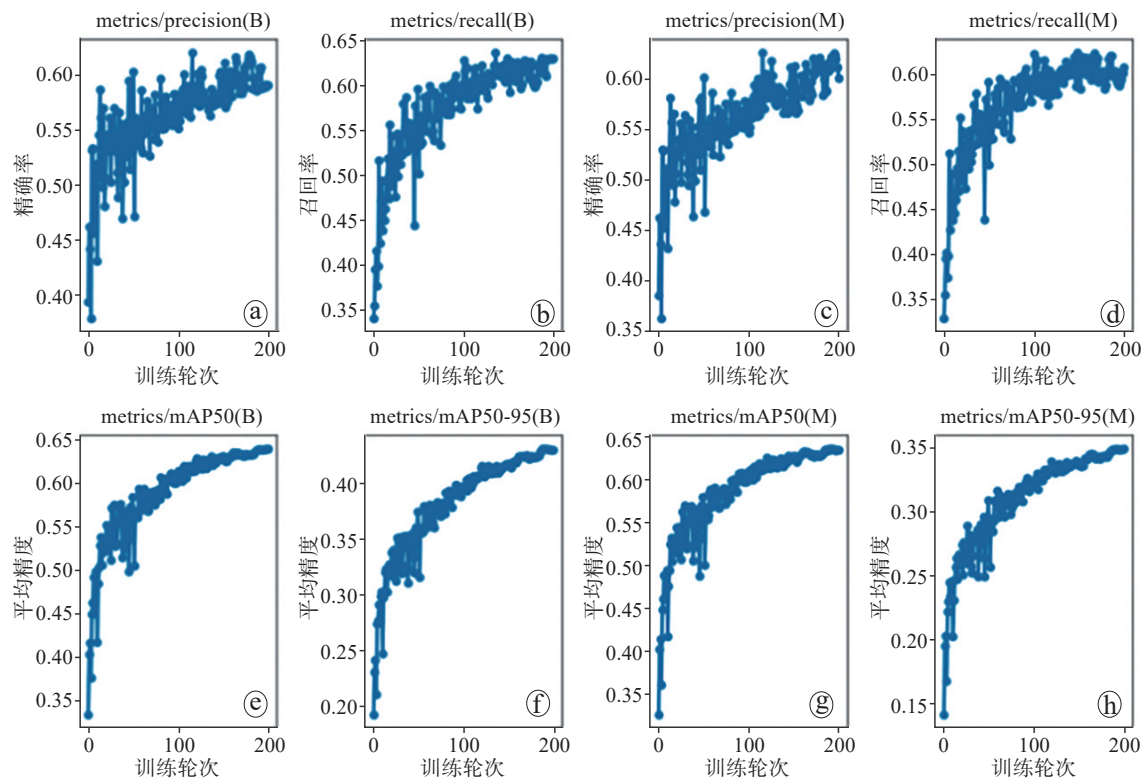


图5 精确率、召回率和mAP指标

Figure 5 Accuracy, recall rate and mAP index

图a~d分别为边界框的精确率和召回率以及分割掩码的精确率和召回率;图e~h分别为IoU(Intersection over Union)阈值为0.5(mAP50)和0.5~0.95(mAP50-95)下边界框的平均精度和分割掩码的平均精度

设备状态,以评估边缘计算模式下模型推理对硬件资源的影响。

3 实验结果

3.1 模型推理准确度

将权重张量的数据类型分别设置为 FP16 和 INT8,并根据第 2.2 小节生成相应的模型。通过在云服务器端使用 PyTorch 模型和在边缘设备上使用 TensorRT 模型对测试集图像进行推理。设置关键的超参数非最大值抑制(NMS)的阈值为 0.01,目标置信度阈值为 0.05。随后将推理结果可视化输出,以便

更直观地观察模型对图像的识别效果。假设以服务器端 PyTorch 推理结果为准,计算 TensorRT 推理结果的准确度。

3 个不同病理标本推理结果如图 6 所示,通过可视化方式将不同细胞类型区分开,3 个不同病理标本图像具体检测到的细胞数量如表 1 所示。图中不同的颜色代表不同类型的细胞实例,展示了每种细胞类型的边界和位置。这种可视化方式使得医学研究人员能够直观了解模型推理结果,更准确地识别病理图像中的不同细胞类型。

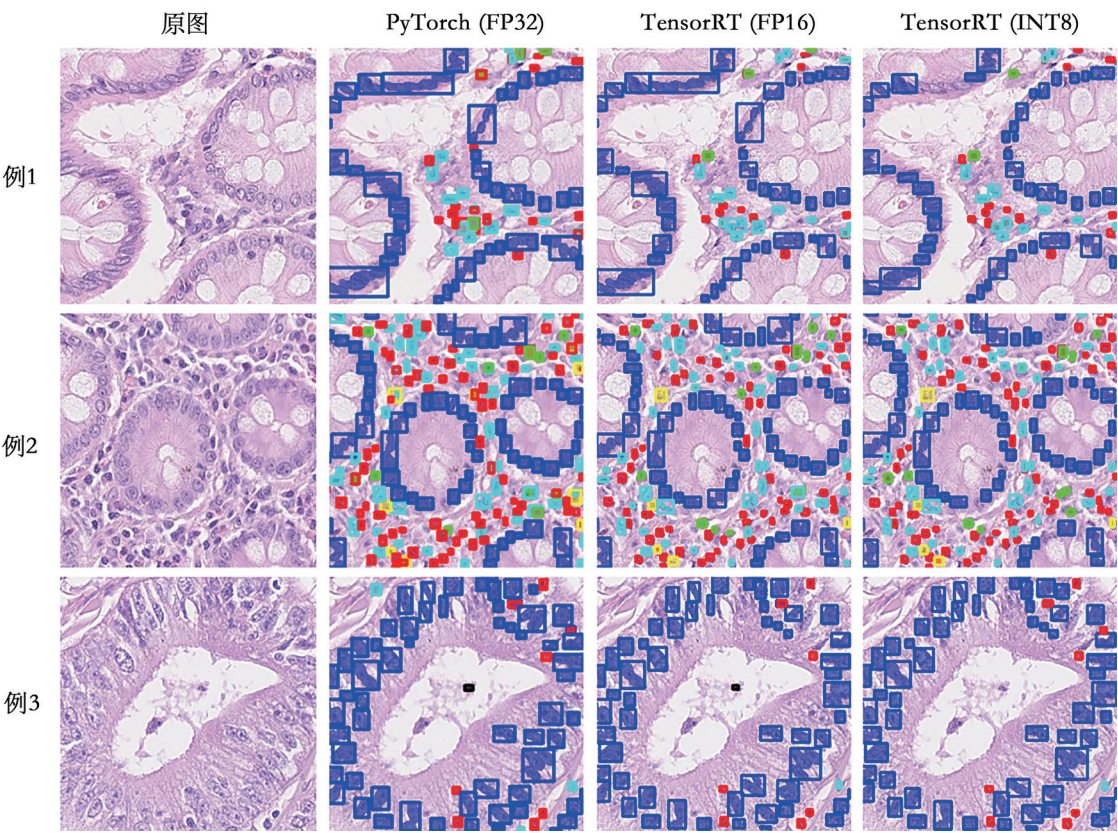


图 6 不同方式推理后的可视化结果

Figure 6 Visualization of different inference and post-processing results

第 1 列为原图,第 2~4 列分别为 PyTorch 在 FP32 精度、TensorRT 在 FP16、INT8 精度的推理结果

表 1 3 种推理方式的目标结果数

Table 1 Target results of 3 inference methods

图像	PyTorch (FP32)	TensorRT(FP16)		TensorRT(INT8)	
	结果	结果	相对差异/%	结果	相对差异/%
例 1	81	78	-3.7	79	-2.4
例 2	207	205	-1.0	197	-4.8
例 3	73	66	-9.5	67	-8.2

相对差异:以 PyTorch 的 FP32 精度模型推理结果为基准,计算其他两种精度模型推理结果的差异

使用 FP16 精度的 TensorRT 模型进行推理时,相比于 PyTorch(FP32),会存在一定程度的精度损失,但推理结果的准确度并未明显降低。在实际应用中,FP16 精度的模型已经能够满足许多场景的需求。进一步将模型的精度从 FP16 降至 INT8,虽然在一些个别大目标的细胞检测方面可能会出现未检测到的情况,但在一些实际应用中,这种精度损失也是可以接受的。特别是在边缘设备资源受限的环境下,采用低精度的模型能够提供更高的效率和性能,并满

足实际的应用需求。

综上所述,从FP32到FP16精度转换的过程可以节省一半的存储空间。将模型精度从FP16转为INT8,能够进一步显著减少存储空间的占用。这对于嵌入式设备和边缘计算平台具有重要意义。笔者观察到,在边缘设备上就推理精确度而言,无论是FP16还是INT8精度,模型推理结果与云端计算相比并没有明显下降,这表明边缘计算在保持推理精度的同时,能够实现更高效的推理任务。

3.2 推理时间测试

根据第1章所提出的解决方案,笔者测试了通过TCP/IP协议传输图像数据和推理结果的时间,选取几次完整传输过程中的时间记录数据,如表2所示。

表2 TCP/IP协议传输数据的时间及推理时间测试(ms)
Table 2 Data transmission and inference time of TCP/IP (ms)

测试	接收到解析完成	推理时间	后处理到发送完成
测试1	53	64	31
测试2	50	64	27
测试3	62	64	22
测试4	57	64	23

根据表2数据可以观察到,从接收数据到解析完成的平均时间约为55 ms,后处理到发送完成平均时间约为26 ms。这包括了数据的接收、解析以及推理完成后的数据处理和发送给采集端计算机。这表明边缘设备能够实时接收并解析通过TCP/IP协议传输的数据,迅速进行推理并返回处理结果。

再对基于YOLOv8 X配置的实例分割细胞核检测模型进行测试,并评估了在不同量化精度下的细胞核检测推理时间。首先生成了TensorRT中FP32精度的模型,然后使用该模型对测试集中的病理图像进行推理,并记录推理时间,FP32、FP16和INT8精度下的推理时间为139、64、39 ms。通过对比不同量化精度下的推理时间,可以评估不同精度下模型推理性能的差异。

再使用YOLOv8 X实例分割的5个不同规模配置[N(3.2 M参数量)、S(11.2 M参数量)、M(25.9 M参数量)、L(large, 43.7 M参数量)、X(68.2 M参数量)]的预训练模型进行推理实验,并记录推理过程中的性能指标。

首先,将PyTorch预训练模型转换为TensorRT的FP16和INT8两种量化模型。在转换过程中,设置模型的类别数为预训练模型的80类,以确保模型在推理过程中对输入数据的处理方式一致,并且保持与PyTorch预训练模型相同的输入规格。接下

来,对5个不同规模的模型分别进行推理。在推理过程中,记录推理时间(表3),并对推理结果进行评估和比较。

表3 不同类型的模型在两种量化精度下的推理时间(ms)
Table 3 Inference time for models under two quantization precisions (ms)

模型类型	规模配置				
	N	S	M	L	X
FP16	9	14	28	41	65
INT8	7	9	19	26	39

实验结果显示,随着量化精度的降低,推理时间逐渐减少。INT8精度的量化模型在推理过程中需要的计算量更少,进一步缩短推理时间。这对于在边缘设备上高效进行模型推理并快速生成结果非常关键,尤其在实时病理图像分析和移动医疗应用中。此项实验验证了边缘计算在病理图像识别中的实时性。

通过对模型在边缘设备上通过TCP/IP协议进行数据接收、推理和返回结果的实验,可以得出结论:边缘计算能够在较短时间内完成数据接收、推理和后处理,并通过TCP/IP协议将结果返回,表现出较低的延迟。在推理时间方面,边缘设备展现出显著的优势,相比于云端计算,由于推理任务在边缘设备上本地完成,无需依赖云端服务器的网络传输和计算资源,因此推理时间大大减少,提高实时性和响应速度。这为实时应用场景提供更好的性能。

3.3 推理性能评估

继续使用基于YOLOv8的5个实例分割预训练模型对更多指标进行监测和记录。在进行循环推理时,将记录推理过程中的内存占用、显存占用、功耗以及设备温度等参数。具体来说,监测每个模型在推理过程中的资源消耗情况,并在推理开始时和推理2 h后分别记录设备的温度。这样的监测和记录将有助于全面了解模型在不同环境下的性能表现,并评估其对硬件资源的需求和影响。表4和表5为同一个程序加载不同规模配置模型时,设备内存和显存的占用情况,表6为推理时的功率数据。

表4 内存占用情况(M)
Table 4 Memory usage (M)

模型类型	规模配置				
	N	S	M	L	X
FP16	623	622	616	615	618
INT8	620	619	611	617	613

表5 显存占用情况(M)
Table 5 Video memory usage (M)

模型类型	规模配置				
	N	S	M	L	X
FP16	864	888	926	973	1 024
INT8	862	871	897	915	945

表6 功率指标(W)
Table 6 Power index (W)

模型类型	规模配置				
	N	S	M	L	X
FP16	2.3	2.6	3.3	4.1	5.1
INT8	2.1	2.2	2.7	3.1	3.7

表7 推理开始(t_1)至第2小时(t_2)的温度变化(℃)
Table 7 Temperature changing from the start (t_1) to the 2nd hour (t_2) (℃)

模型类型	规模设置									
	N		S		M		L		X	
	t_1	t_2	t_1	t_2	t_1	t_2	t_1	t_2	t_1	t_2
FP16	45.0	47.2	44.7	48.3	44.3	49.0	44.7	51.0	44.9	53.0
INT8	44.7	47.2	43.0	47.3	45.2	48.2	44.9	48.9	45.0	49.8

综上所述,在边缘设备上进行模型推理时,边缘设备在内存、显存和功耗等方面表现出更高的效率和更低的资源消耗。这为在资源有限的边缘环境中部署深度学习模型提供可行性和可靠性保障。这对于在边缘设备上实现高效的病理图像分析和移动医疗应用提供有力的支持。

4 结 论

本研究通过对边缘设备进行实验,评估了TCP/IP协议数据传输时间、推理精确度、推理时间和资源利用等指标,并将其与云端计算进行比较。实验结果验证了此边缘计算方案的隐私安全性、推理高效和实时性、低成本等优势,而且可以针对不同的任务需求,选择不同的模型进行单独或个性化的部署,具有相当的适用性和可扩展性。此边缘计算方案显示出了巨大潜力,在病理图像分析方面具有很高的应用价值。相对于传统的云端计算模式,在多个方面呈现出明显优势。(1)保护隐私方面的优势。通过在边缘设备上进行本地数据处理,敏感数据可以避免传输到云端,有助于保护用户和患者的隐私。(2)相对于云端计算减少数据传输时间和延迟。边缘设备有效利用了本地计算资源,在设备附近进行数据处理

根据表4~表6数据可以观察到,在边缘设备上使用FP16或INT8精度的模型进行推理时,模型对内存和显存的占用相对较低,并且功率消耗也相对较低。边缘计算有效地利用了有限的内存资源,并通过优化显存管理,使得边缘设备能够高效地处理和存储病理图像数据。同时,在保持较高的推理精度的前提下,边缘设备降低内存和显存的需求。边缘设备在推理过程中能够有效管理功耗,降低设备的能耗。这对于边缘设备的能源管理和延长设备续航时间具有积极的影响。

在进行模型推理时,边缘设备会产生一定的热量,但温度的上升幅度相对较小,表7记录了设备从推理开始(t_1)至第2小时(t_2)的温度变化。这表明边缘计算在设备散热方面表现出较好的效果。

和分析,减少数据传输量和传输时间,提高实时性。这不仅降低数据传输和处理的成本,还节省带宽成本。(3)推理时间方面的优势。相对于将数据传输到云服务器进行处理的云端计算模式,边缘设备上的推理时间更短。这减少了结果反馈的延迟,有助于实现实时诊断和移动医疗的要求。(4)边缘计算减少对网络连接的依赖性。即使在网络不稳定或断开的情况下,边缘设备仍然能够进行数据处理和决策,提高系统的可靠性。可减少医学机构的服务器机房维护成本。(5)边缘设备的便携性也使得其适用于医学研究人员进行教学工作。传统的医学教学通常需要依赖大型服务器和计算机设备,这给教学过程中的移动性和灵活性带来了一定限制。

基于以上优势,边缘计算对推动医疗领域的智能化发展和边缘计算技术的应用提供有力支持,具有很高的应用价值,未来的研究可以进一步探索边缘计算在其他医疗应用领域的潜力,并优化其性能和效果,以进一步提升医疗服务的质量和效率。

推理在少数情况下,有个别大目标细胞未能被检测到,未来还可以进一步优化模型的推理算法和参数设置,以提高模型的检测准确度和鲁棒性。同时探索更先进的模型架构和训练技术,例如模型剪

枝技术,进一步提高模型性能和效率^[24]。通过持续的研究探索,期待能够进一步推动医学图像分析领域的发展,为实现智慧医疗做出更大贡献。

【参考文献】

- [1] Shi WS, Pallis G, Xu ZW. Edge computing [scanning the issue][J]. Proc IEEE, 2019, 107(8): 1474-1481.
- [2] Gezer V, Um J, Ruskowski M. An introduction to edge computing and a real-time capable server architecture[J]. Int J Adv Intell Syst, 2018, 11(1/2): 105-114.
- [3] Shin HC, Roth HR, Gao MC, et al. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning[J]. IEEE Trans Med Imaging, 2016, 35(5): 1285-1298.
- [4] 杨培伟,周余红,邢岗,等.卷积神经网络在生物医学图像上的应用进展[J]. 计算机工程与应用, 2021, 57(7): 44-58.
Yang PW, Zhou YH, Xing G, et al. Applications of convolutional neural network in biomedical image[J]. Computer Engineering and Applications, 2021, 57(7): 44-58.
- [5] 王茹. 基于深度学习的病理学图像可解释性分类[D]. 太原: 山西大学, 2023.
Wang R. Interpretative classification of pathological images based on deep learning[D]. Taiyuan: Shanxi University, 2023.
- [6] 周涛,董雅丽,霍兵强,等. U-Net网络医学图像分割应用综述[J]. 中国图象图形学报, 2021, 26(9): 2058-2077.
Zhou T, Dong YL, Huo BQ, et al. U-Net and its applications in medical image segmentation: a review[J]. Journal of Image and Graphics, 2021, 26(9): 2058-2077.
- [7] 张玮智,于谦,苏金善,等. 从U-Net到Transformer: 深度模型在医学图像分割中的应用综述[J]. 计算机应用, 2024, 44(S1): 204-222.
Zhang WZ, Yu Q, Su JS, et al. From U-Net to transformer: application review of deep models in medical image segmentation [J]. Journal of Computer Applications, 2024, 44(S1): 204-222.
- [8] 陈全,邓倩妮. 云计算及其关键技术[J]. 计算机应用, 2009, 29(9): 2562-2567.
Chen Q, Deng QN. Cloud computing and its key techniques[J]. Journal of Computer Applications, 2009, 29(9): 2562-2567.
- [9] 张建勋,古志民,郑超. 云计算研究进展综述[J]. 计算机应用研究, 2010, 27(2): 429-433.
Zhang JX, Gu ZM, Zheng C. Survey of research progress on cloud computing[J]. Application Research of Computers, 2010, 27(2): 429-433.
- [10] Satyanarayanan M. The emergence of edge computing[J]. Computer, 2017, 50(1): 30-39.
- [11] Zhou YQ, Tian L, Liu L, et al. Fog computing enabled future mobile communication networks: a convergence of communication and computing[J]. IEEE Commun Mag, 2019, 57(5): 20-27.
- [12] Qi YL, Tian L, Zhou YQ, et al. Mobile edge computing-assisted admission control in vehicular networks: the convergence of communication and computation[J]. IEEE Veh Technol Mag, 2019, 14(1): 37-44.
- [13] Shi Y. Edge computing in medical imaging: a review[J]. IEEE J Biomed Health Inform, 2020, 24(12): 3437-3449.
- [14] Zeng Z, Chen C, Veeravalli B, et al. Introduction to the special issue on edge intelligence: neurocomputing meets edge computing [J]. Neurocomputing, 2022, 472: 149-151.
- [15] 胡新平,张志美,董建成. 基于云计算理念与技术的医疗信息化[J]. 医学信息学杂志, 2010, 31(3): 6-9.
Hu XP, Zhang ZM, Dong JC. Medical informatization based on cloud computing concepts and techniques[J]. Journal of Medical Intelligence, 2010, 31(3): 6-9.
- [16] Scalcon FP, Tahar R, Ahrabi M, et al. AI-powered video monitoring: assessing the NVIDIA jetson Orin devices for edge computing applications[C]//2024 IEEE Transportation Electrification Conference and Expo (ITEC). Piscataway, NJ, USA: IEEE, 2024: 1-6.
- [17] Alqahtani DK, Cheema MA, Toosi AN. Benchmarking deep learning models for object detection on edge computing devices [C]//Service-Oriented Computing. Singapore: Springer Nature Singapore, 2025: 142-150.
- [18] NVIDIA. Developer guide: NVIDIA deep learning TensorRT documentation[EB/OL]. <https://docs.nvidia.com/deeplearning/tensorrt/developer-guide/index.html>.
- [19] Jeong EJ, Kim J, Tan S, et al. Deep learning inference parallelization on heterogeneous processors with TensorRT[J]. IEEE Embed Syst Lett, 2022, 14(1): 15-18.
- [20] 王学东,黄宏成. 复杂变道场景下的轻量化车道线检测算法研究[J]. 传动技术, 2023, 37(3): 3-15.
Wang XD, Huang HC. Research on lightweight lane detection algorithm in complex lane changing scenes[J]. Drive System Technique, 2023, 37(3): 3-15.
- [21] 黄靖淞. 基于嵌入式GPU的AI加速推理技术研究[D]. 北京: 中国科学院光电技术研究所, 2021.
Huang JS. Research on AI accelerated inference technology based on embedded GP[D]. Beijing: The Institute of Optics and Electronics, The Chinese Academy of Science, 2021.
- [22] Graham S, Jahanifar M, Vu QD, et al. CoNIC: colon nuclei identification and counting challenge 2022[EB/OL]. (2021-11-29). <https://arxiv.org/abs/2111.14485>.
- [23] 高威. 基于病理全切片图像分析的胰腺癌患者生存预测模型[D]. 南京: 南京信息工程大学, 2023.
Gao W. Survival prediction model of pancreatic cancer patients based on pathological whole section image analysis[D]. Nanjing: Nanjing University of Information Science and Technology, 2023.
- [24] 易啸,马胜,肖依. 深度学习加速器在不同剪枝策略下的运行优化[J]. 计算机工程与科学, 2023, 45(7): 1141-1148.
Yi X, Ma S, Xiao N. Running optimization of deep learning accelerators under different pruning strategies[J]. Computer Engineering & Science, 2023, 45(7): 1141-1148.

(编辑:薛泽玲)