E2EdgeAI：将低功耗边缘计算应用于在微型自主无人机上部署基于视觉的深度神经网络

Mozhgan Navardi[1], Edward Humes[2], and Tinoosh Mohsenin[3]

[1]Center for Real-time Distributed Sensing and Autonomy (CARDS)
[2]Computer Science and Electrical Engineering Department
[3]University of Maryland Baltimore County, USA

摘　要

人工智能（AI）和深度神经网络（DNNs）因其支持诸如视觉感知、视觉导航等应用而受到自动系统领域关注。虽然基于云端的解决方案已得到广泛应用，但 AI 和 DNNs 相结合这一方法因其具有更低延迟并能够避免数据向远程服务器传输过程中的种种安全隐患而受到持续关注。然而，由于边缘设备计算能力有限并且功耗（能效）问题至关重要，因此在类似设备上部署深度神经网络富有挑战。在这项工作中，我们提出了一种高能效边缘计算方法——E2EdgeAI，能够利用人工智能技术为微型自主无人机提供支持。该方法通过考虑内存访问和 CPU 核心利用率对无人机功耗的影响优化深度神经网络资源消耗。在实验部分，我们使用 Crazyflie 微型无人机进行验证，该无人机带有 AI-deck 扩展，其中包括八核 RISC-V 处理器。实验结果表明：我们所提出的方法可以将模型大小减小 14.4 倍，单次推理功耗减少 78%，并将能源利用率提升 5.6 倍。演示视频见此处：Video。

关键词：边缘计算; 自动系统; 障碍躲避; 无人机导航

1　引言和相关工作

如今，微型无人机 (UAV) 等边缘设备已被用于许多室内场景，如搜索救援、气体泄漏定位和实时监控等对人类太过危险或耗时的活动 [1-3]。由于微型无人机能够安全地在人类附近操控，有能力进入人类无法轻易到达的空间，并且它们身形小巧还能够配备各种传感器，这使其在如上应用场景中具有潜在优势。此外，前述的多传感器配置使无人机更易导航以及完成任务方案 [4]。

为使微型无人机变得智能和自主，可以在其上部署深度神经网络 [2-3,5] 或强化学习（RL）[6-10] 模型。DNNs 帮助处理无人机相机和其他传感器捕获的数据，以实现无人机自主导航并完成任务。然而，由于参数和计算量大大增加，DNNs 十分耗电且需要大量计算资源 [11]。为解决上述困难，云计算、无人机与云端间数据传输作为解决方案被提出。虽然云计算能够提供强大算力，但受带宽限制，边缘能够发送到云端的数据量有限。此外，安全问题也是云计算面临的另一个潜在挑战。因此，由于存在通信延迟和潜在安全问题，云计算不适合实时、低延迟或隐私敏感的应用。幸运的是，近期研究表明边缘计算能够支持在边缘设备上运行 DNN 模型，同时仍满足上述延迟和隐私约束 [12-13]。

边缘计算的主要思想是在网络边缘设备上处理捕获数据以减少不必要数据传输，提高能源效率 [11-16]。虽然边缘计算具有较低通信延迟等固有优势，但在边缘设备上部署 DNN 模型仍然具有挑战性，这主要受限于能耗和可用资源。因此，为边缘设备优化 DNN，如降采样图像 [17-18] 和 DNN 稀疏化 [19]，以实现低延迟、高能效部署十分重要。本文实现了将当前最先进（State-Of-The-Art aka SOTA）的视觉 DNN 模型转换成低能耗边缘计算模型。

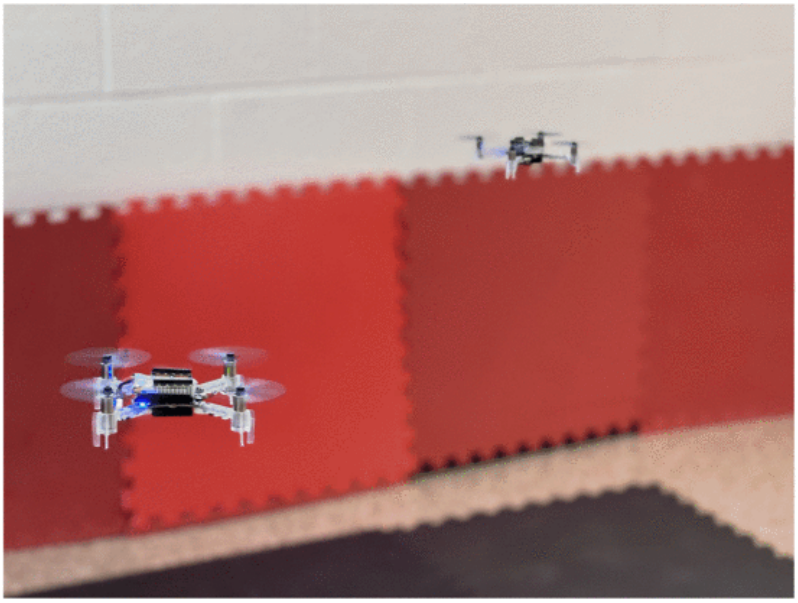

图 1: 配备 AI-deck 扩展的微型无人机 Crazyflie。AI-deck 采用 GAP8 处理器，具有 8 个能够并行计算的 RISC-V 核心。

为了在各种应用场景中利用无人机完成任务，实现自主无人机导航是一个初步问题。因此，在这项工作中，我们以无人机导航为目标，并利用深度神经网络模型和边缘计算来解决该问题。我们提出了 E2EdgeAI 方案，该方法能够在满足延迟和能效比约束的同时为无人机提供自主导航。本文主要贡献总结如下：

- 探讨云计算在自主无人机导航应用中实现实时决策所面临的挑战和瓶颈。
- 优化基于视觉的深度神经网络模型，减少模型大小和计算复杂度，实现高能效边缘计算。
- 实现端对端低功耗半载处理和边缘计算，同时满足延迟约束。
- 分析 DNN 部署在资源受限设备上时的功耗和延迟，包括内存访问和计算核心使用情况。
- 在延迟、吞吐量、功耗和能源效率方面优化配置无人机硬件。

为评估 E2EdgeAI，我们将优化后的模型部署在 Crazyflie[5] 边缘设备上，如图 1 所示。实验结果表明：与 SOTA 方法相比，E2EdgeAI 能够减少 4 倍延迟，提高 5.6 倍能量效率。

2 问题定义和研究动机

本节定义了本工作所讨论的问题陈述，之后通过一个示例展示边缘计算有效性。

2.1 问题定义

本文涉及自主系统中延迟和能量效率问题。E2EdgeAI 尝试通过降低模型大小和边缘计算来减小延迟。为实现这一目标，该方法优化了模型的内存使用情况以提高能量效率和模型性能。该问题可以表述为：

输入： 给定以下内容：

- SOTA 深度神经网络模型。
- 多核嵌入式系统，可并行运行 DNN 模型。

- 边缘设备从环境中采集的图像。

输出：确定一个在约束条件下经过优化的小型 DNN 模型。
约束条件：延迟、隐私和能量效率是主要约束条件。
目标：本工作主要目标是利用边缘计算来改善延迟和能效。

2.2 示例

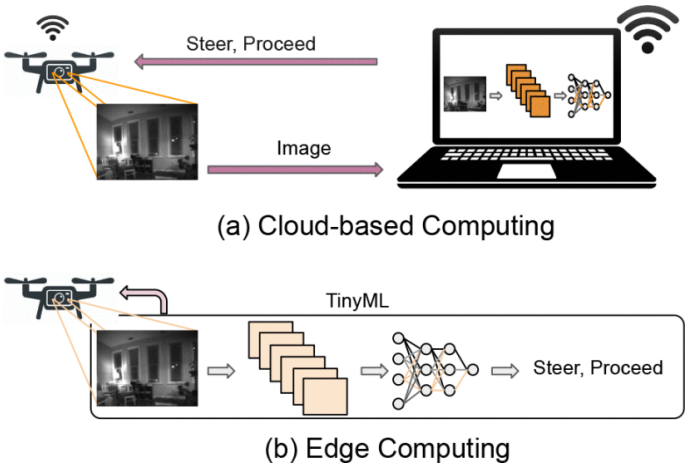

图 2: 基于云计算和边缘计算的无人机导航和避障示例。(a) 云计算方法云端进行模型推理。模型输入、输出分别由无人机端发送、接收。连接延迟、带宽限制以及数据安全问题是该种方法所面临的主要挑战。(b) 边缘计算将模型部署在边缘设备上、无需考虑通信速度和数据安全问题。

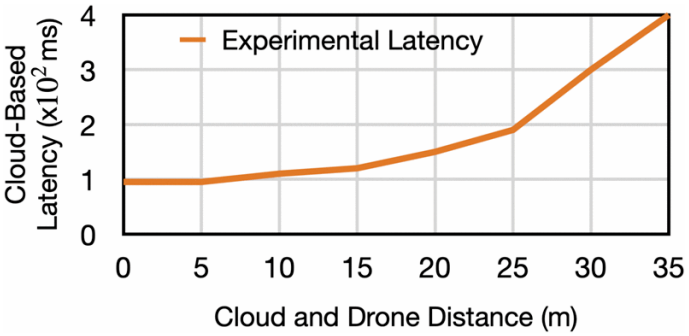

图 3: 云计算方法延迟测量实验。在该实验中，DNN 模型部署在云端笔记本电脑上，无人机采集图像并将其发送至云端，之后通过云端获得控制指令。

图 2 (a) 展示了基于云计算方法的系统概述。无人机采集到数据（本例中为图像）后通过 WiFi 传输到笔记本电脑，DNN 模型部署在无人机附近的笔记本电脑上，电脑端接收数据进行处理，并将结果发送回无人机。云端方法中通信延迟会随无人机与 WiFi 网络之间距离的增加而增加，最终无人机将与云端断开连接。图 3 显示了云计算方法测量推理阶段计算和通信延迟的实验结果：无人机与云端相隔 35 米时，延迟可达到 400 毫秒，并且距离大于 35 米后无人机与云端不再有连接。基于上述问题，边缘计算是一种更为安全可靠的方法，该方法中 DNN 模型将直接部署在无人机上，示意图详见图 2 (b)。

3 提出方法：E2EdgeAI

本节提出 E2EdgeA 方案，该方法优化了一个 DNN 模型用于高能效边缘计算。本节首先介绍了模型架构和量化方法，之后讨论了内存访问和核心使用如何影响机载模型的延迟和能

量效率。

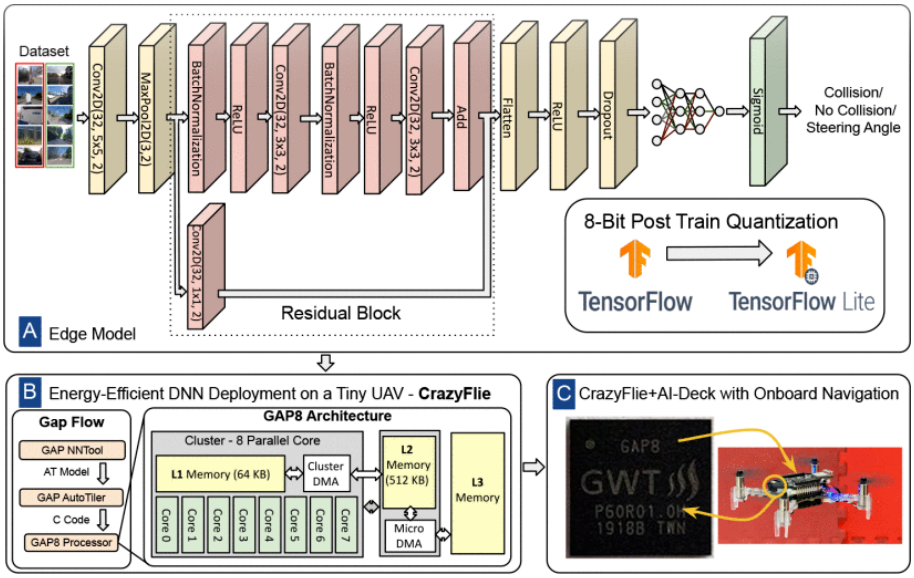

图 4: 边缘计算方法——E2EdgeAI：(a) 优化 DNN 模型 (b) 在边缘部署模型并分析内存访问情况与核心利用率 [5,20-21] (c) 无人机自主导航

3.1 基于视觉的 DNNs 模型优化

应用和数据集。本项工作针对微型自主无人机导航进行研究。为此，我们需要训练一个模型，使得无人机在飞行过程中能够避免碰撞。我们基于 SOTA 深度神经网络结构设计、训练模型，该模型可以预测碰撞概率，且能够基于单目灰度摄像头输出一个合适的转向角使飞机避开障碍物以实现自主飞行。无人机接收上述两个模型输出后据此完成导航：若第一个输出产生的碰撞概率高于某一阈值，无人机将基于第二个输出生成的角度进行转向。

本项工作共用到两个数据集训练模型，能够帮助实现无人机碰撞避免并根据原始灰度图像控制转向：(1) 该数据集 [22] 包含约 32000 张图像，并用 0/1 标签进行注释，分别表示有碰撞/无碰撞；(2) Udacity 项目数据集 [23] 包含 70000 张图像，采集了驾驶汽车时左、中、右三个视角的画面。

模型架构。由于物联网和嵌入式系统等边缘设备计算能力有限，所以边缘计算需要简化复杂模型，减少操作次数。因此，考虑到应用场景和硬件实现，我们提出了一种可以根据模型大小和计算量进行缩放的深度神经网络。在本项工作中，我们使用了四个可扩展模型（包括 MobileNet[24] 和 ResNet[25]）来训练 DNN 模型。与 [3] 类似，我们考虑了四个不同配置，并提取了精度、大小和计算次数。我们通过分配不同 Width Multiplier：0.25、0.5、0.75、1 来修改 MobileNet 模型复杂度；对于 ResNet，我们使用自定义 Resnet，并依次序列化 1、2、3、4 个残差块（Residual Blocks）。图 4(a) 展示了具有一个残差块的自定义 ResNet。类似于我们先前针对时间序列音频输入 [26] 和图像输入 [27] 分类的工作，以及在 [17-18] 研究工作中，我们认为输入图像尺寸是影响模型大小和操作次数的一个因素。

边缘模型。一个简单的差异点可能会产生颠覆性的影响。例如，大多数神经网络使用浮点数（FP）进行设计和训练，这也是计算机表示实数集合的方法。由于浮点数在现代软件中非常普遍，大多数处理器已针对浮点运算进行了优化，能够做到以处理整数运算相同或更快的速度执行浮点运算。然而，嵌入式处理器为满足大小、重量和功率约束，它们通常不具备 FP 处理单元，而只支持整数运算；即使含有 FP 单元，处理器执行浮点运算的速度仍要慢于整数运算。因此，为使嵌入式设备能够以任何合理的性能水平运行神经网络模型，我们需要将神经

网络模型中的浮点运算转换为低精度整数运算，这一过程被称为量化。目前有两种常见模型量化方法：后训练量化（PTQ）和量化感知训练（QAT）。我们利用 TensorFlow Lite 对优化模型进行 PTQ 量化操作，将其从 32 位 FP 模型转换为 8 位量化模型。经过上述过程，该模型不仅适用于边缘部署，且占用更少的内外存储空间。在下一节中，我们将分析优化模型的能量效率和推理延迟。

	下载/上传速度		模型功耗		
网络类型	下载速度（Mbps）	上传速度（Mbps）	α_u（mW/Mbps）	α_d（mW/Mbps）	β（mW）
无线网络	55	19	283	137	133

表 1: 无线网络平均下载速度、上传速度和功耗模型参数 [3,28]

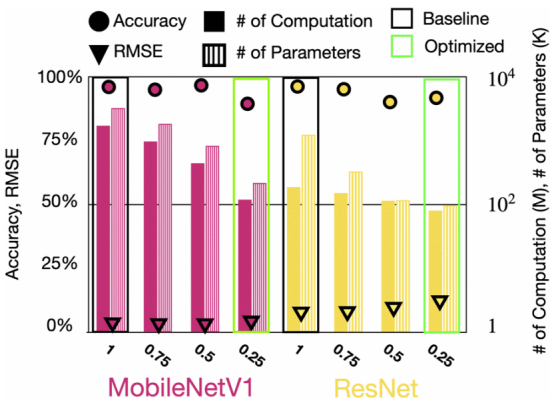

图 5: 参考 MobileNetV1[24] 和 ResNet[25-26] 网络结构，有四种不同图像输入尺寸（原始图像 324x244），训练基于视觉的障碍物检测 DNN 模型。与基准模型相比，ResNet 和 MobileNet 模型大小分别优化了约 14 倍和 12 倍 [3]。

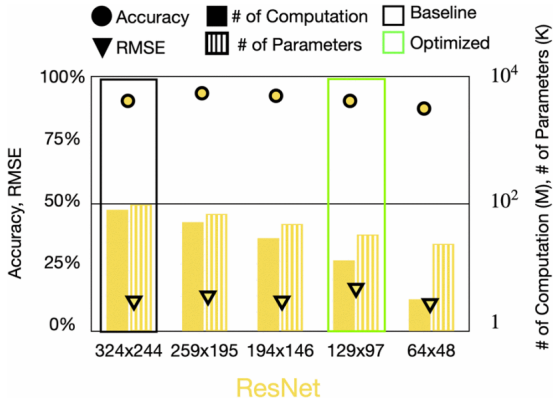

图 6: ResNet 模型在图像输入缩放比为 0.25 时取得最优结果，即当输入图像大小为 129x97 时，可以兼顾计算复杂度与模型精度，实验结果表明此时模型大小可缩小至原来的 1/3。

3.2 边缘计算资源优化

延迟和能耗分析。障碍物检测对于无人机自主导航实现碰撞避免具有关键作用，当前基于视觉的 DNN 模型是一个有前景的解决方案。然而，在无人机导航这一应用场景中，通信延迟、带宽和功耗是实现实时决策的重要参数。为获得以上参数，我们参考了无线网络的标准下载速度、平均上传速度以及通信功耗 [3,28]，借助表 1 数据，可以基于公式 1 计算出利用 WiFi 传输数据时延迟、功耗的理论值：

$$\begin{aligned} p_{uplink} &= \alpha_u \times upload\_speed + \beta, \\ p_{downlink} &= \alpha_d \times download\_speed + \beta \end{aligned} \tag{1}$$

公式 1 中 $upload\_speed$ 和 p_{uplink} 分别表示数据通过 WiFi 从无人机传输到云端的平均速度和上载功耗，$download\_speed$ 和 $p_{downlink}$ 则分别表示下载速度和下载功耗，α_u, α_d 和 β 为数据传输过程中的（包括上载和下载）功耗系数，单位是 mW/Mbps。为降低延迟和能耗，我们优化了 DNN 模型大小，减少运行模型所需计算量，使其可以部署在微型无人机等资源受限的设备上。

硬件架构。图 4 说明了 E2EdgeAI 在 GAP8[5,20-21] 等资源受限处理器上部署 DNN 模型的过程。图4 (b) 和图 4 (c) 显示 Crazyflie 有一个 GAP8 处理器，它具有基于 RISC-V 的

PULP 平台，其中包括两个计算域：（1）用于控制作业的结构控制器（Fabric Controller）和 512KB 二级缓存（L2），（2）用于并行计算高需求工作负载的 8 计算核心集群和 64KB 可直接访问一级缓存（L1）。在本项工作中，我们使用 GAPFlow 工具链，其中包括 NNTOOL 和 AutoTiler2 个程序。NNTOOL 负责对 DNN 架构进行调整，将模型转换为与 AutoTiler 兼容的格式，以及转换权重格式以便写入 GAP8；AutoTiler 负责利用算法确定针对 DNN 模型的最佳可实现内存布局，将模型操作转换为 C 代码以便 GAP8 进行编译。虽然上述过程大部分已实现自动化处理，但某些 DNN 模型通常需要进行一定调整才能正确转换，例如，某些深度神经网络会在栈空间分配过多数据，模型转换时需要调整结构控制器和计算核心集群的最大堆栈大小；同样，堆空间也需要针对特定 DNN 进行调整，AutoTiler 默认为 DNN 模型分配整个系统的一级缓存和二级缓存，这将不可避免地导致堆溢出，即在神经网络模型运行期间，某些数据结构和栈空间等内存数据将会被覆盖重写，造成严重后果。此外，GAP8 处理器使用实时操作系统（RTOS），它在 DNN 启动之前会频繁自分配堆空间，导致 DNN 可用内存空间减少。我们分析了 GAPFlow 和内存分配对不同 DNN 模型功耗和延迟的影响，并提出了一种适用于边缘计算的无人机导航高能效模型。在下一节中，我们将展示模型优化结果并基于 DNN 模型边缘实现进行分析。进一步，我们讨论了延迟和能效方面的改进办法。

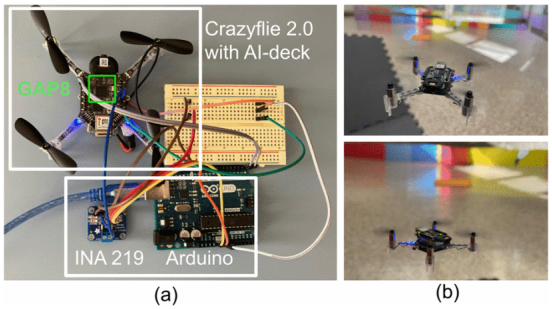

图 7: (a) AI-deck-GAP8 处理器连接在带有功率测量装置的 Crazyflie 上，INA219 和 Arduino 用于测量 GAP8 功耗 [6] (b) Crazyflie 运行基线模型以实现障碍物检测和规避。

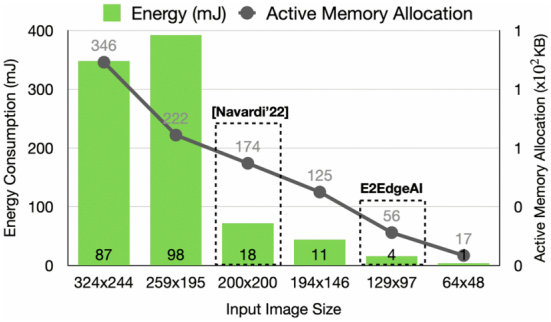

图 8: 各 ResNet 模型（输入图像大小不同）在推理阶段的平均功耗和活动内存分配，并将 E2EdgeAI 模型与 [3] 中方法作比较。

方法	[Navardi' 22] [3]	E2EdgeAI
推理延迟（ms）	40	10
吞吐量（推理/秒）	25	100
功耗（mW）	470	400
性能（GOPS）	1	1.3
能效（GOPS/W）	55	323
单次推理能耗（mJ）	18.8	4

表 2: E2EdgeAI 与 [3] 中硬件实现结果对比。实验利用最新 NINA 固件 [29] 在 GAP8 处理器上部署模型，并提取延迟和功耗数据。

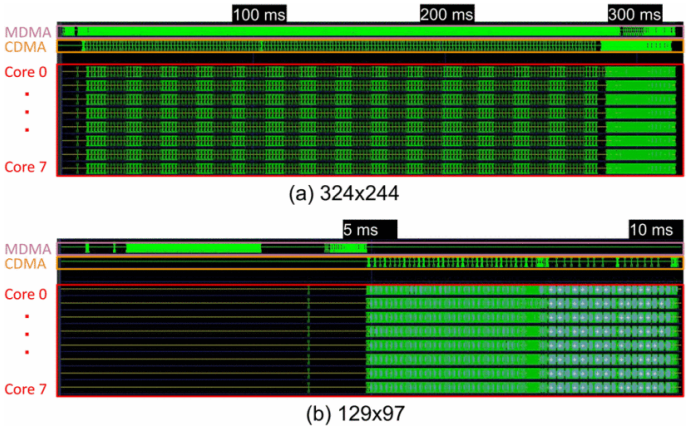

图 9: ResNet 模型推理阶段 VCD 轨迹 [30]。共有两种不同图像输入尺寸：（a）324x244，（b）129x97。

4 方法实施与实验结果

4.1 软件实验结果

本节展示了障碍检测和转向角 DNN 模型的训练结果。，如 3.1 所述，我们采用 ResNet 和 MobileNet 网络结构，并使用了四个不同模型缩放比和六种不同图像输入。图 5 表示通过减小两种网络模型大小，能够减少计算量和参数数量。由于优化后的 ResNet 模型具有较低计算复杂度（是基线模型的 1/14），最终我们选择该模型，并分析输入图像大小对模型精度的影响。图 5 显示当 ResNet 模型残差块数量为 1，图像输入大小为 129x97 是，复杂度较基线模型减小三倍以上，而精度几乎相同（92%）。我们针对不同图像输入大小的 ResNet 模型进行 PTQ 量化操作以分析 GAP8 处理器的延迟和能效。

4.2 GAP8 边缘设备实施结果和分析

为评估 E2EdgeAI 方法，我们将训练好的模型部署在 GAP8 处理器上。此外，我们比较了 E2EdgeAI 与 [3] 中方法的能效和延时。为保证公平，我们对 [3] 提出的边缘模型进行重复测量，以获得相同配置下的结果。图 7 （a）展示了 [3,6] 中使用的功率测量设置。E2EdgeAI 的延迟、功耗和能效如表 2 所示，结果表明：E2EdgeAI 比 [3] 中模型速度快 4 倍，能效高 5.8 倍。图 8 展示了输入大小对推理阶段活动内存分配和能耗的影响。为测量活动内存分配情况，我们提取了推理过程中任意给定时间最大内存使用量。由于活动内存分配大小大于 L1（64Kb），[3] 中结果表明它同时使用了 L1 和 L2，而 E2EdgeAI 能够将模型压缩为 56 Kb，这说明 E2EdgeAI 可以仅使用 L1 缓存。因此，可以证明 E2EdgeAI 能耗更低（大约 4mJ）。图 7 （b）展示了障碍物检测与避免的基线模型运行情况。

为展示不同 DNN 模型复杂度下能效如何变化，我们提取了其 VCD 轨迹 ***。基础 ResNet 模型（324x244）与优化后的 E2EdgeAI 模型（129x97）中 ClusterDMA（CDMA）、Micro-DMA（MDMA）和核心利用率的 VCD 轨迹 [30] 如图 9 所示。CDMA 引擎负责在 L1 和 L2 缓存之间移动数据，同时 L1 缓存可看做一个用于应用程序管理的划线板，而 MDMA 负责 GAP8 与其他外部设备之间数据传输。深度神经网络模型需要将权重信息从 L3 缓存中移动至内存，L3 缓存实际上是连接到超级总线（HyperBus）的额外 RAM，此类 RAM 速度比 L1/L2 缓存慢得多，耗能也更多，但存储能力更强 [6]。图 9 （a）中 MDMA 和 CDMA 单元几乎在整个推理阶段中都处于活动状态，所以处理器必须浪费时钟周期等待数据传输，而图 9 （b）显示模型优化后两个 DMA 单元都不再被频繁占用。此外，推理结束时 MDMA 单元完全停止使用，CDMA 单元访问频率大大降低，实现更低能耗。

5 总结

本文提出了 E2EdgeAI 方案用于在微型无人机上实现高能效边缘计算，并利用深度神经网络实现无人机自主导航。我们评估了 E2EdgeAI 模型的吞吐量和精度，并且在配备 AI 推理功能的无人机上（Crazyflie）成功部署了此模型。然后，我们将 E2EdgeAI 与 SOTA 方法在延迟、功耗、单次推理能耗、性能和能量效率等方面进行对比，实验结果显示：相比 SOTA 方法，我们的模型复杂度减少了 14.4 倍，能效提高 5.6 倍，单次推理能耗减少 78%。

6 致谢

感谢 Griffin Bonner、Haoran Ren、Aidin Shiri 和 Tejaswini Manjunath 参与初步讨论和实验。该项目由美国陆军研究实验室赞助，合作协议编号：W911NF2120076。

参考文献

[1] SHAKHATREH H, SAWALMEH A H, AL-FUQAHA A, et al. Unmanned Aerial Vehicles (UAVs): A Survey on Civil Applications and Key Research Challenges[J]. IEEE Access, 2019, 7: 48572-48634. DOI: 10.1109/ACCESS.2019.2909530.

[2] DUISTERHOF B P, KRISHNAN S, CRUZ J J, et al. Tiny Robot Learning (tinyRL) for Source Seeking on a Nano Quadcopter[C]//2021 IEEE International Conference on Robotics and Automation (ICRA). 2021: 7242-7248. DOI: 10.1109/ICRA48506.2021.9561590.

[3] NAVARDI M, SHIRI A, HUMES E, et al. An Optimization Framework for Efficient Vision-Based Autonomous Drone Navigation[C]//2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS). 2022: 304-307. DOI: 10.1109/AICAS54282.2022.9869975.

[4] MÜLLER H, NICULESCU V, POLONELLI T, et al. Robust and Efficient Depth-based Obstacle Avoidance for Autonomous Miniaturized UAVs[Z]. 2022. arXiv: 2208.12624 [cs.RO].

[5] PALOSSI D, LOQUERCIO A, CONTI F, et al. A 64-mW DNN-Based Visual Navigation Engine for Autonomous Nano-Drones[J]. IEEE Internet of Things Journal, 2019, 6(5): 8357-8371. DOI: 10.1109/JIOT.2019.2917066.

[6] SHIRI A, NAVARDI M, MANJUNATH T, et al. Efficient Language-Guided Reinforcement Learning for Resource-Constrained Autonomous Systems[J]. IEEE Micro, 2022, 42(6): 107-114. DOI: 10.1109/MM.2022.3199686.

[7] NAVARDI M, DIXIT P, MANJUNATH T, et al. Toward real-world implementation of deep reinforcement learning for vision-based autonomous drone navigation with mission [J]. UMBC Student Collection, 2022.

[8] PRAKASH B, WAYTOWICH N, OATES T, et al. Towards an Interpretable Hierarchical Agent Framework using Semantic Goals[Z]. 2022. arXiv: 2210.08412 [cs.LG].

[9] PRAKASH B, WAYTOWICH N, MOHSENIN T, et al. Automatic Goal Generation using Dynamical Distance Learning[Z]. 2021. arXiv: 2111.04120 [cs.AI].

[10] PRAKASH B, WAYTOWICH N, OATES T, et al. Interactive Hierarchical Guidance using Language[Z]. 2021. arXiv: 2110.04649 [cs.AI].

[11] LI E, ZENG L, ZHOU Z, et al. Edge AI: On-Demand Accelerating Deep Neural Network Inference via Edge Computing[J]. IEEE Transactions on Wireless Communications, 2020, 19(1): 447-457. DOI: 10.1109/TWC.2019.2946140.

[12] CHANG Z, LIU S, XIONG X, et al. A Survey of Recent Advances in Edge-Computing-Powered Artificial Intelligence of Things[J]. IEEE Internet of Things Journal, 2021, 8(18): 13849-13875. DOI: 10.1109/JIOT.2021.3088875.

[13] PREMSANKAR G, GHADDAR B. Energy-Efficient Service Placement for Latency-Sensitive Applications in Edge Computing[J]. IEEE Internet of Things Journal, 2022, 9(18): 17926-17937. DOI: 10.1109/JIOT.2022.3162581.

[14] SHI W, CAO J, ZHANG Q, et al. Edge Computing: Vision and Challenges[J]. IEEE Internet of Things Journal, 2016, 3(5): 637-646. DOI: 10.1109/JIOT.2016.2579198.

[15] MAZUMDER A N, MOHSENIN T. A Fast Network Exploration Strategy to Profile Low Energy Consumption for Keyword Spotting[Z]. 2022. arXiv: 2202.02361 [cs.LG].

[16] MAZUMDER A N, MENG J, RASHID H A, et al. A Survey on the Optimization of Neural Network Accelerators for Micro-AI On-Device Inference[J]. IEEE Journal on Emerging and Selected Topics in Circuits and Systems, 2021, 11(4): 532-547. DOI: 10.1109/JETCAS.2021.3129415.

[17] GENC H, ZU Y, CHIN T W, et al. Flying IoT: Toward Low-Power Vision in the Sky [J]. IEEE Micro, 2017, 37(6): 40-51. DOI: 10.1109/MM.2017.4241339.

[18] PALOSSI D, ZIMMERMAN N, BURRELLO A, et al. Fully Onboard AI-Powered Human-Drone Pose Estimation on Ultralow-Power Autonomous Flying Nano-UAVs[J]. IEEE Internet of Things Journal, 2022, 9(3): 1913-1929. DOI: 10.1109/JIOT.2021.3091643.

[19] LAMBERTI L, NICULESCU V, BARCIŚ M, et al. Tiny-PULP-Dronets: Squeezing Neural Networks for Faster and Lighter Inference on Multi-Tasking Autonomous Nano-Drones[C]//2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS). 2022: 287-290. DOI: 10.1109/AICAS54282.2022.9869931.

[20] FLAMAND E, ROSSI D, CONTI F, et al. GAP-8: A RISC-V SoC for AI at the Edge of the IoT[C]//2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP). 2018: 1-4. DOI: 10.1109/ASAP.2018.8445101.

[21] GreenWavesTechnologies. "Gap8 processor architecture[Z]. https://greenwaves-technologies.com/manuals/BUILD/HOME/html/index.html.

[22] LOQUERCIO A, MAQUEDA A I, DEL-BLANCO C R, et al. DroNet: Learning to Fly by Driving[J]. IEEE Robotics and Automation Letters, 2018, 3(2): 1088-1095. DOI: 10.1109/LRA.2018.2795643.

[23] Udacity. An open source self-driving car[Z]. https://www.udacity.com/. 2016.

[24] HOWARD A G, ZHU M, CHEN B, et al. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications[Z]. 2017. arXiv: 1704.04861 [cs.CV].

[25] HE K, ZHANG X, REN S, et al. Deep Residual Learning for Image Recognition[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2016: 770-778. DOI: 10.1109/CVPR.2016.90.

[26] REN H, MAZUMDER A N, RASHID H A, et al. End-to-end Scalable and Low Power Multi-modal CNN for Respiratory-related Symptoms Detection[C]//2020 IEEE 33rd International System-on-Chip Conference (SOCC). 2020: 102-107. DOI: 10.1109/SOCC 49529.2020.9524755.

[27] KHATWANI M, RASHID H A, PANELIYA H, et al. A Flexible Multichannel EEG Artifact Identification Processor Using Depthwise-Separable Convolutional Neural Networks[J/OL]. J. Emerg. Technol. Comput. Syst., 2021, 17(2). https://doi.org/10.1145 /3427471. DOI: 10.1145/3427471.

[28] ESHRATIFAR A E, PEDRAM M. Energy and Performance Efficient Computation Offloading for Deep Neural Networks in a Mobile Cloud Computing Environment[C/OL] //GLSVLSI '18: Proceedings of the 2018 on Great Lakes Symposium on VLSI. Chicago, IL, USA: Association for Computing Machinery, 2018: 111-116. https://doi.org/10.114 5/3194554.3194565. DOI: 10.1145/3194554.3194565.

[29] Bitcraze. Firmware running on the esp32 nina w102 module of the ai-deck[Z]. https://g ithub.com/bitcraze/aideck-esp-firmware. 2021.

[30] GreenWavesTechnologies. Gvsoc virtual platform for gap8 processor.[Z]. https://green waves-technologies.com/manuals/BUILD/GVSOC/html/index.html.