YOLO Nano:一个高度紧凑的你只看一次卷积神经网络的目标检测

Alexander Wong[1,2], **Mahmoud Famuori**[1,2], **Mohammad Javad Shafiee**[1,2]
Francis Li[2], **Brendan Chwyl**[2], **and Jonathan Chung**[2]
[1]Waterloo Artificial Intelligence Institute, University of Waterloo, Waterloo, ON, Canada
[2]DarwinAI Corp., Waterloo, ON, Canada

Abstract

目标检测仍然是计算机视觉领域的一个活跃研究领域，通过设计用于处理目标检测的深度卷积神经网络，在这一领域取得了相当大的进展和成功。尽管取得了这些成功，但在边缘和移动场景中广泛部署此类目标检测网络的最大挑战之一是高计算和内存要求。因此，人们对设计适合边缘和移动使用的高效深度神经网络架构的研究兴趣越来越大。在本研究中，我们引入了一种高度紧凑的深度卷积神经网络YOLO Nano，用于目标检测任务。利用人机协作设计策略创建YOLO Nano，其中基于YOLO系列单镜头目标检测网络架构的设计原则的原则网络设计原型，与机器驱动的设计探索相结合，创建一个紧凑的网络，具有高度定制的模块级宏观架构和为嵌入式目标检测任务量身定制的微架构设计。所提出的YOLO Nano模型大小为∼ 4.0MB(分别比Tiny YOLOv2和Tiny YOLOv3小> 15.1 ×和> 8.3 ×)，需要4.57B次推理运算(比Tiny YOLOv2和Tiny YOLOv3低>34%和∼ 17%)。)，同时在VOC 2007数据集上仍然实现了∼69.1%的mAP(分别比Tiny YOLOv2和Tiny YOLOv3高∼ 12%和∼ 10.7%)。在Jetson AGX Xavier嵌入式模块上进行了不同功耗预算下的推理速度和功耗效率实验，进一步验证了YOLO Nano在嵌入式场景下的有效性。

1 简介

对象检测是计算机视觉领域的一个活跃领域，其目标不仅是在场景中定位感兴趣的对象，而且还要为每个感兴趣的对象分配一个类标签。最近在目标检测领域取得的相当大的成功源于深度学习[8, 7]的现代进步，特别是利用深度卷积神经网络。最初的重点是提高准确性，导致越来越复杂的目标检测网络，如SSD [11], R-CNN [2], Mask R-CNN [3]，以及这些网络的其他扩展变体[6, 9, 18]。虽然这种网络展示了最先进的目标检测性能，但由于计算和内存的限制，在边缘和移动设备上部署它们非常具有挑战性，如果不是不可能的话。事实上，即使是更快的变体，如faster R-CNN [15]，在嵌入式处理器上运行时，其推理速度也只有较低的个位数帧速率。这极大地限制了此类网络在无人机、视频监控、自动驾驶等需要本地嵌入式处理的广泛应用中的广泛采用。

为了解决实现嵌入式目标检测的这一挑战，人们对探索和设计更适合边缘和移动设备[12, 13, 14, 23, 4, 17]的高效深度神经网络架构越来越感兴趣。围绕效率设计的一个特别有趣的目标检测网络家族是YOLO系列神经网络架构[12, 13, 14]，它利用许多设计原则来创建单次架构，可以在高端桌面gpu上实现嵌入式目标检测性能。然而，这些网络架构对于许多边缘和移动场景来说仍然太大(例如，在YOLOv3架构的情况下，∼ 240MB)，并且由于计算复杂性(例如，在YOLOv3的情况下，> 65B操作)，它们在边缘和移动处理器上运行时的推理速度大大下降。为了解决这个问题，Redmon等人引入了Tiny YOLO网络架构家族，它以牺牲对象检测性能为代价大大减小了模型尺寸。

在本研究中，我们致力于探索一种人机协作设计策略，设计高度紧凑的深度卷积神经网络来完成目标检测任务，其中原则网络设计原型与机器驱动的设计探索相结合。更具体地说，我们在这种人机协作设计策略中利用YOLO系列单次目标检测网络架构的设计原则来创

Preprint. Under review.

建YOLO Nano，这是一个高度紧凑的网络，具有高度定制的模块级宏架构和微架构设计，专为嵌入式目标检测任务量身定制。

2 方法

在这项研究中，我们介绍了YOLO Nano，这是一种高度紧凑的深度卷积神经网络，用于嵌入式目标检测，采用人机协作设计策略[21]。YOLO Nano的人机协同设计策略包括两个主要设计阶段:1)原则网络设计原型，2)机器驱动设计探索。

2.1 原则网络设计原型

创建YOLO Nano的第一个设计阶段是有原则的网络设计原型阶段，我们创建了一个初始的网络设计原型(表示为φ)，基于人驱动的设计原则来指导机器驱动的设计探索阶段。更具体地说，我们基于YOLO系列单次架构[12, 13, 14]的设计原则构建了一个初始网络设计原型。YOLO系列网络架构的一个突出特点是，与基于区域提议的网络不同，基于区域提议的网络依赖于构建区域提议网络来生成场景中物体所在位置的提议，然后对所生成的提议进行分类，而是利用单一的网络架构来处理输入图像并生成输出结果。因此，单幅图像的所有目标检测预测都是在一次前向传递中完成的，相比之下，基于区域提议的网络需要执行数百到数千次才能获得最终结果。这使得YOLO系列网络架构的运行速度大大加快，因此更适合嵌入式对象检测。

本研究中使用的初始设计原型从YOLO系列网络架构中获得灵感，由一堆特征表示模块组成，模块之间具有[14]的快捷连接。此外，与[14]一样，特征表示模块的配置方式类似于特征金字塔网络[10]，因此它能够在三个不同的尺度上表示特征。这些特征表示模块之后是几个卷积层，输出是一个三维张量，用于编码三种不同尺度的边界框、对象和类预测。因此，这种初始设计原型架构设计允许高效的多尺度目标检测。

最终的YOLO Nano网络架构中各个模块和层的实际宏架构和微架构设计，以及网络模块的数量，都留给机器驱动的设计探索阶段，以自动确定给定的数据，以及人为指定的设计要求和约束，这些要求和约束专门针对计算和存储能力有限的边缘和移动场景设计。

2.2 机器驱动设计探索

使用初始网络设计原型(φ)、数据以及针对边缘和移动使用的人为指定的设计需求作为指导，然后利用机器驱动的设计探索阶段来确定所提议的YOLO Nano网络架构的模块级宏观架构和微架构设计。更具体地说，本研究以生成合成[22]的形式实现了机器驱动的设计探索，能够在人类指定的需求和约束下确定最终网络架构的最佳宏观架构和微架构设计。生成综合的总体目标是学习能够生成满足设计需求和约束的深度神经网络的生成机器，其描述如下。这在生成合成的概念中被表述为一个约束优化问题，用于确定一个生成器$\mathcal{G}$，给定一组种子S，可以生成网络$\{N_s|s \in S\}$最大化通用性能函数$\mathcal{U}$(例如[20])，同时满足通过指示函数$1_r(\cdot)$定义的要求和约束:

$$\mathcal{G} = \max_{\mathcal{G}} \mathcal{U}(\mathcal{G}(s)) \text{ subject to } 1_r(\mathcal{G}(s)) = 1, \ \forall s \in S. \tag{1}$$

由于方程1所提出的约束优化问题的全局最优解在计算上是难以解决的，鉴于可行区域的巨大，我们通过迭代优化来求解近似解$\hat{\mathcal{G}}$，其中初始解$\hat{\mathcal{G}}_0$由$\varphi, \mathcal{U}$和$1_r(\cdot)$引导，并逐步更新，使每个连续的近似解$\hat{\mathcal{G}}_k$比以前的近似解(即$\hat{\mathcal{G}}_1, \ldots, \hat{\mathcal{G}}_{k-1}$等)获得更高的$\mathcal{U}$，同时仍然受到$1_r(\cdot)$的约束。最后的近似解$\hat{\mathcal{G}}$用于创建所提出的YOLO纳米网络。

为了指导生成合成过程学习生成机器，生成边缘和移动场景的目标检测网络，不仅高效紧凑，而且还提供强大的目标检测性能，关键步骤之一是配置指标函数$1_r(\cdot)$以强制执行适当的设计要求和约束。在本研究中，指标函数$1_r(\cdot)$的设置如下:i)平均精度(mAP) $\geq 65\%$的VOC 2007, ii)计算成本$\leq$ 5B操作，iii) 8位权重精度。计算成本约束的设置使得所得的YOLO纳米网络的计算成本低于Tiny YOLOv3 [14]，后者是嵌入式目标检测中最流行的紧凑型网络之一。

3 YOLO纳米建筑设计

所提出的用于嵌入式目标检测的YOLO纳米网络的网络架构如图1所示，其中有几个有趣的观察结果值得在下面讨论。

3.1 残差投影-扩展-投影宏观结构

关于YOLO Nano网络体系结构的第一个值得注意的观察结果是，它与YOLO网络家族有很大的不同，它由具有独特的残差投影-扩展-投影(PEP)宏观体系结构的模块组成，此外还有像[16, 19, 1]中那样的扩展-投影(EP)宏观体系结构。剩余的PEP宏体系结构包括:I)一个有1个$\times$ 1个卷积的投影层，它将输出通道投射到一个低维的输出张量中ii)一个有1个$\times$ 1个卷积的扩展层，它将通道的数量扩展到一个更高的维度，iii)一个深度卷积层，它对来自扩

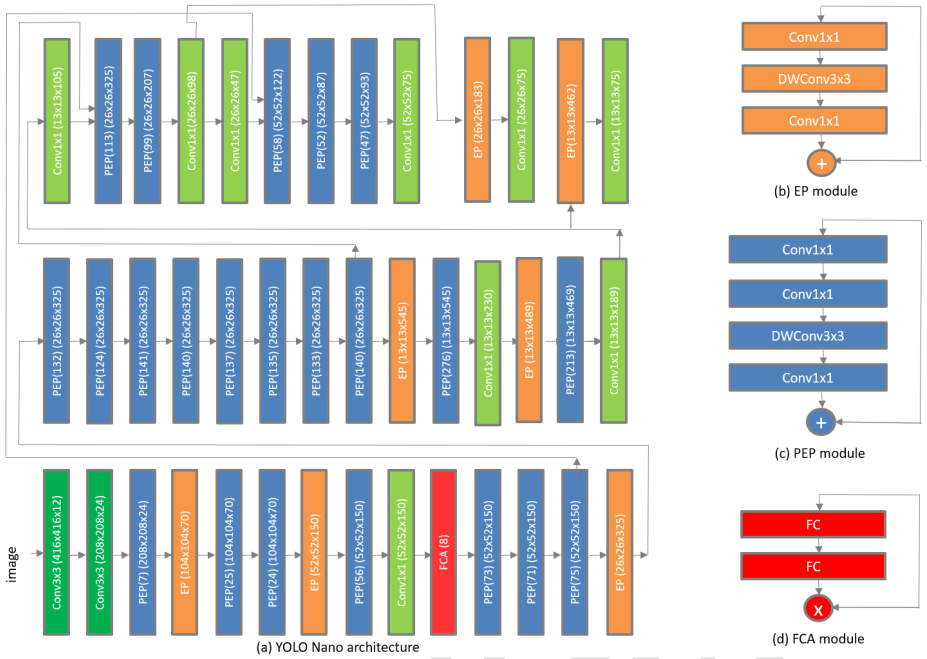

Figure 1: YOLO Nano网络架构。注意，PEP(x)表示残留PEP模块第一投影层的x通道，FCA(x)表示还原比x

展层的每个单独的输出通道使用不同的滤波器进行空间卷积，iv)一个有1个× 1个卷积的投影层，它将输出通道投影到一个低维的输出张量中。残余PEP宏观体系结构的使用可以显著降低体系结构和计算复杂性，同时保持模型的表达性。

3.2 全连接注意力宏观架构

关于YOLO纳米网络架构的第二个值得注意的观察是，通过机器驱动的设计探索过程，在网络中战略性地引入了轻量级全连接注意力(FCA)，这与其他设计探索方法中固定模块级的引入形成了对比[19]。与[5]一样，FCA宏观架构由两个完全连接的层组成，这些层学习信道之间的动态、非线性相互依赖关系，并产生调制权重，以便通过信道智能乘法重新加权信道。FCA的使用有助于基于全局信息的动态特征重新校准，从而更加关注信息特征，从而更好地利用可用的网络容量。这反过来又允许在降低的体系结构和计算复杂性与模型表达性之间取得强有力的平衡。

3.3 宏观架构和微架构异构性

关于YOLO纳米网络架构的第三个值得注意的观察是，不仅在宏观架构(PEP模块、EP模块、FCA以及单独的3 × 3和1 × 1卷积层的不同组合)方面具有高度的异质性，而且在单个特征表示模块和层的微架构方面也具有高度的异质性，网络中的每个模块或层都具有独特的微架构。在YOLO Nano网络体系结构中具有高微体系结构异构性的好处是，它允许对网络体系结构的每个组件进行独特的定制，从而在体系结构和计算复杂性以及模型表达性之间实现非常强的平衡。YOLO Nano中的这种架构多样性还展示了利用机器驱动的设计探索策略的优势，这种策略与生成合成一样灵活，这对于人类设计师或其他设计探索方法(如[19, 1])来说是不可能将网络架构定制到这种架构粒度级别的。

4 实验结果及讨论

为了研究YOLO Nano在嵌入式目标检测中的有效性，我们在PASCAL VOC数据集上考察了它的模型大小、目标检测精度和计算成本。出于比较的目的，Tiny YOLOv2网络[13]和Tiny YOLOv3网络[14]被用作基准参考，因为它们是用于嵌入式对象检测的最流行的紧凑深度神经网络之一，因为它们的模型尺寸小，计算复杂性低。VOC2007/2012数据集由20种不同类型的物体标注的自然图像组成。使用VOC2007/2012训练数据集对深度神经网络进行训练，并按照研究文献的标准做法，在VOC2007测试数据集上计算平均精度(mAP)来评估深度神经网络的目标检测精度。

表1给出了所提出的YOLO纳米网络以及Tiny YOLOv2和Tiny YOLOv3的模型尺寸和目标检测精度。首先，观察到YOLO Nano的模型大小为4.0MB，分别比Tiny YOLOv2和Tiny YOLOv3小> 15.1 ×和> 8.3 ×，这对于边缘和移动场景来说非常重要，因为内存限制。其次，尽管YOLO Nano的模型尺寸要小得多，但在VOC 2007测试数据集上的mAP值

为69.1%，分别比Tiny YOLOv2和Tiny YOLOv3高∼ 12%和∼ 10.7%。第三，YOLO Nano只需要45.7亿次运算来执行推理，比Tiny YOLOv2低b> 34%，比Tiny YOLOv3低∼ 17%。

Table 1: 在VOC 2007测试集上测试的紧凑网络的目标检测精度结果。所有测试网络的输入大小为416 × 416。最好的结果以粗体突出显示。

Model Name	Model size	mAP (VOC 2007)	computational cost (ops)
Tiny YOLOv2 [13]	60.5MB	57.1%	6.97B
Tiny YOLOv3 [14]	33.4MB	58.4%	5.52B
YOLO Nano	**4.0MB**	**69.1%**	**4.57B**

最后，为了研究YOLO Nano在嵌入式场景中的实际性能，我们评估了YOLO Nano在不同功耗预算下在Jetson AGX Xavier嵌入式模块上运行的推理速度和功耗效率。在15W和30W的功率预算下，YOLO Nano的推理速度分别为∼ 26.9 FPS和∼ 48.2 FPS，功率效率分别为∼ 1.97图像/秒/瓦和∼ 1.61图像/秒/瓦。这些实验结果表明，通过人机协作设计策略创建的YOLO纳米网络在精度、尺寸和计算复杂性之间提供了良好的平衡，使其非常适合边缘和移动场景的嵌入式目标检测。

References

[1] X. Chu, B. Zhang, R. Xu, and J. Li. Fairnas: Rethinking evaluation fairness of weight sharing neural architecture search. *arXiv preprint arXiv:1907.01845*, 2019.

[2] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[3] K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask r-cnn. *ICCV*, 2017.

[4] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[5] J. Hu, L. Shen, S. Albanie, G. Sun, and E. Wu. Squeeze-and-excitation networks. *IEEE TPAMI*, 2019.

[6] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, et al. Speed/accuracy trade-offs for modern convolutional object detectors. In *IEEE CVPR*, 2017.

[7] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[8] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 2015.

[9] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In *CVPR*, volume 1, page 4, 2017.

[10] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[11] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. SSD: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[13] J. Redmon and A. Farhadi. YOLO9000: better, faster, stronger. *arXiv preprint*, 1612, 2016.

[14] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[15] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[16] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4510–4520, 2018.

[17] M. J. Shafiee, B. Chywl, F. Li, and A. Wong. Fast YOLO: A fast you only look once system for real-time embedded object detection in video. *arXiv preprint arXiv:1709.05943*, 2017.

[18] A. Shrivastava, A. Gupta, and R. Girshick. Training region-based object detectors with online hard example mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 761–769, 2016.

[19] M. Tan, B. Chen, R. Pang, V. Vasudevan, and Q. V. Le. Mnasnet: Platform-aware neural architecture search for mobile. *arXiv preprint arXiv:1807.11626*, 2018.

[20] A. Wong. Netscore: Towards universal metrics for large-scale performance analysis of deep neural networks for practical usage. *arXiv preprint arXiv:1806.05512*, 2018.

[21] A. Wong, Z. Q. Lin, and B. Chwyl. Attonets: Compact and efficient deep neural networks for the edge via human-machine collaborative design. *arXiv preprint arXiv:1903.07209*, 2019.

[22] A. Wong, M. J. Shafiee, B. Chwyl, and F. Li. Ferminets: Learning generative machines to generate efficient neural networks via generative synthesis. *Advances in neural information processing systems Workshops*, 2018.

[23] B. Wu, F. Iandola, P. H. Jin, and K. Keutzer. Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. *arXiv preprint arXiv:1612.01051*, 2016.