arXiv:2304.02101v1 [cs.DC] 4 Apr 2023

MadEye：通过自适应摄像机配置提升实时视频分析准确率

Mike Wong[1], Murali Ramanujam[1],Guha Balakrishnan[2],Ravi Netravali[1]

[1]普林斯顿大学 [2]莱斯大学

摘要

摄像机的方向（即旋转和缩放）决定了摄像机在给定场景中捕获的内容，进而对实时视频分析流水线的准确率产生重大影响。然而，现有的视频分析方法忽略了这一关键的自适应调节手段，而只调整如何对固定方向下捕获的图像进行编码、传输和分析。我们提出 MadEye，这是一种摄像机—服务器系统，可自动且持续地调整摄像机方向，以针对当前工作负载和资源约束最大限度地提高准确率。为利用商用云台变焦（PTZ）摄像机实现这一目标，MadEye 集成了：（1）一种搜索算法，可快速探索庞大的方向空间，及时识别出一组有效方向；（2）一种新颖的知识蒸馏策略，仅利用摄像机资源便能高效地选择可最大限度提高工作负载准确率的方向。在多种工作负载上的实验表明，在资源使用量相同的情况下，MadEye 可将准确率提高 2.9–25.7%；或者在准确率相同的情况下，将资源成本降低 2–3.7× 。

1 引言

随着摄像机部署持续增长，以及用于视觉任务（例如分类或检测）的深度神经网络（DNN）不断进步， [3, 17, 40, 59, 63], 实时视频分析流水线已日益普及。这些流水线持续将摄像机传来的视频流传输至处理服务器（位于边缘端 [4, 7, 69, 74, 100] 或云端 [28, 51, 60, 110]），并在输入帧上运行 DNN，为不同的应用查询提供低延迟、高准确率的结果；应用查询即任务、DNN 和感兴趣对象的组合。典型应用包括自动驾驶、客流追踪、交通协调和商业分析等 [1, 5, 8, 20, 22, 26, 35, 36, 82, 83]。

鉴于实时视频分析流水线在实践中的重要性，许多研究致力于提升其资源效率和准确率。现有解决方案包括针对准确率调优推理配置、编码方式或外观调节参数 [29, 52, 77, 110], 过滤冗余内容 [21, 28, 48, 60],使用成本更低的模型变体 [4, 84], 改进作业调度 [74, 88, 110], 等等。然而，这些工作都假设摄像机可观测到的内容无法改变，只能通过不同方式进行编码、流式传输或分析。从本质上说，它们专注于优化固定的、预设的摄像机部署。

遗憾的是，为分析任务部署摄像机本身就是一项艰巨的工作。受实际条件限制（例如安装支架、电源），针对感兴趣场景，操作人员必须确定要部署的摄像机数量，以及每台摄像机采用的方向（即旋转和缩放倍率的组合）。可选方向众多，更改这些设置需要人工干预。然而，我们发现这样做可能带来显著收益：在不同工作负载和场景中，随时间动态调整方向，相较于最佳 固定朝向方案，可将准确率提升 21.3–35.3%，且不会增加资源使用量。此外，仅通过部署更多固定摄像头来同时覆盖更多方向，并不能获得这些收益：大多数方向在较短的累计时段内才是“最佳”选择（每段 10 分钟视频的中位数为 6 秒），这会大幅降低该方法的效率；在运行视频分析的资源受限环境中，这一问题尤为突出 [4, 61, 69, 89]。

另一种策略是利用云台摄像机提供的软件库来调节方向，从而以合理的方式实现上述准确率提升。事实上，尽管云台摄像机已经问世近二十年，但近年来其受欢迎程度大幅上升（ 2020 [16]年全球市场价值达30亿美元），这主要归功于价格下降，如今其成本已可与固定摄像机相媲美 [44, 86]。然而，在实时分析中的应用仍面临多项挑战（§2.3）。首先，由于查询目标各异（例如任务）、模型固有偏差（模型如何感知场景和物体）以及场景动态变化（物体所在位置不同），查询对朝向参数的敏感方式也各不相同；针对一种工作负载优化方向调节，可能会使另一种工作负载的中位数准确率提升潜力损失高达25.1%。其次，方向“网格”规模庞大，但可选空间稀疏，且在任意时刻，从最佳方向到其他方向的准确率都会急剧下降。第三，最佳方向变化迅速，例如，85%的变化发生在上次变化后的 ≤1 秒内。

为克服这些问题，我们提出了**MadEye**，这是一种摄像头-服务器系统，可自动持续调整云台摄像机朝向，最大限度地提高当前场景和工作负载下的分析准确率。Mad-Eye 的核心洞见是，商用云台摄像机切换

朝向的速度（即每秒超过 600°，且几乎可即时进行数字变焦）远远快于应用程序获取分析结果的频率（通常为每秒 1–30 帧，即每隔 33–1000 毫秒）。因此，MadEye 无需采用仅依赖先前探索来判断朝向重要性的常见非平稳多臂老虎机策略 [57, 73, 98]，而是可以采用基于当前场景内容、更有依据的策略。具体而言，在每个时间步（30 帧率时为 33 毫秒），并视网络/计算资源可用性而定，MadEye 摄像机会探索多个朝向，并迅速确定哪种朝向最能提高工作负载准确率，值得传输到后端进行完整推理。然而，要在实践中实现这一策略，还需要应对若干技术挑战。

首先，为了能够快速在摄像头端评估不同方向的重要性，MadEye 采用了一种定制的知识蒸馏 [43]策略，使用边缘级超压缩神经网络模型。为应对这些模型可能有限的预测能力，我们只让它们对查询敏感度建模，目标是根据各方向对工作负载准确率的影响，准确地排序各方向；精确结果则交由后端服务器处理。即便采用这种宽松的任务设定，MadEye 仍须进行多项优化，才能达到足够的排序准确率。最值得注意的是，MadEye 使用一种通用抽象——对感兴趣对象进行检测——来训练边缘模型；这种抽象反映了捕捉常见任务的敏感度和偏差所需的最少信息。边缘模型无需内置特定任务的语义，这些语义可以通过后处理融入生成的结果。

边缘模型会在 MadEye 后端持续训练，使用最新和历史工作负载结果，旨在缓解数据偏斜问题，避免模型过度偏向近期选定的方向（因为接下来会选择什么方向尚不确定）。重要的是，为了平衡资源成本和准确率，每个边缘模型只涵盖单个查询，但涵盖所有方向。其直觉在于：虽然模型结果可能存在显著差异，但对于同一场景，不同方向之间的特征级方差要小得多，通常也小于典型预训练数据集中的方差 [6, 10, 30, 56], [62]。因此，MadEye 会在不同查询间冻结预训练的特征提取层，并将这些权重缓存在摄像头上，从而降低重新训练和（下行链路）模型更新开销。

其次，我们提出一种新颖的摄像头端搜索策略，旨在探索各个方向，以便在每个时间步捕获准确率方面最佳的方向。我们的搜索由三项关键的经验观察所引导：（1）尽管时间变化迅速，最佳方向之间的转换在空间维度上却变化缓慢；（2）最佳方向通常呈空间聚类；（3）具有重叠区域的相邻方向在效能方面呈现高度相关的趋势。

基于这些观察，MadEye 会在每个时间步探索形状灵活的连续方向，并且只考虑转向那些效能能够可靠预测的相邻方向。保留或移除方向的决策同时取决于响应速率（及相应的时间预算），以及对近期边缘模型结果的相对 比较。对于前者，MadEye 使用一种高效的启发式方法，在时间预算内判断路径可行性（这是 NP难旅行商问题的一种变体 [42]）。对于后者，MadEye 会在探索（即形状大小）与网络使用量（即向后端发送更多方向进行推理）之间灵活权衡，以限制边缘模型误差的影响，并在满足所需响应速率的同时最大限度地提高准确率。

为了评估 MadEye，我们构建了据我们所知首个支持在每个时刻调整旋转和缩放的数据集，方法是从公开可用的 360°视频中剪辑出感兴趣的场景。借助该数据集，我们在多种网络条件和工作负载下评估了 MadEye，这些工作负载包含多种视觉深度神经网络和查询任务：分类、计数（逐帧和汇总）以及检测。在这些设置下，与预知最优固定朝向策略相比，MadEye 在不增加资源使用量的情况下将准确率提升了 2.9-25.7%；这一表现与预知最优动态策略相差不超过 1.8-13.9%。换一种说法，与采用（多个）固定朝向摄像头的最佳策略相比，MadEye 在实现上述准确率提升的同时，资源占用降低了 2-3.7× 。此外，MadEye 的表现优于近期的云台俯仰变焦跟踪算法 [85, 90]（提升 2.0-3.8×）和多臂老虎机方案 [97] （提升 5.8×）。我们将公开发布 MadEye 和我们的数据集。本研究不会引发任何伦理问题。

2 背景与动机

我们首先概述实时视频分析的部署情况（§2.1）。接着展示测量结果，说明根据工作负载和场景动态调整摄像机朝向的重要性（§2.2），并介绍在实践中实现这些优势所面临的挑战（§2.3）。

2.1 实时视频分析概述

在实时视频分析部署中，一个或多个摄像头会持续将视频帧流传输到服务器进行处理。服务器可以是远程（但性能强大）的云端机器 [88, 110]，也可以是附近（但性能较弱）的边缘设备 [4, 69, 74],，其任务是在输入帧上运行查询，以支持不同的应用程序。查询通常涉及对单帧运行深度神经网络（ DNN）推理，旨在定位并描述场景中的各种物体，例如十字路口。此外，不同应用程序的查询在所执行的任务、所关注的物体、所使用的深度神经网络（不同的架构和权重）以及所需的响应速率方面可能各不相同。例如，商业分析中的客流追踪

会统计经过某个区域的人数，所需响应速率为 1 帧率或更低 [8]。相比之下，智能驾驶或体育分析应用程序会检测汽车或人的具体位置，所需响应速率可达 30 帧率以上 [83]。

本文聚焦于近期文献 [19, 28, 53, 54, 60]和实际部署 [67, 74]中普遍存在的以下四种查询任务（及其对应的准确率指标）。我们指出，这些查询类型也是构建复杂应用程序和其他任务的基础，例如，跟踪查询依赖于对象检测。

• **二分类**：询问视频帧中是否存在任何感兴趣对象。视频整体的准确率以二元判断正确的视频帧所占比例来衡量。

• **计数**：统计每个视频帧中的感兴趣对象数量。每个视频帧的准确率以返回计数与真实标注计数之间的百分比差异来衡量。

• **检测**：查找视频帧中感兴趣对象的精确边界框坐标。每个视频帧的准确率使用mAP [33], 衡量，该指标评估每个返回框与其对应真实标注框之间的重叠程度。

• **聚合计数**：统计场景中出现的唯一 感兴趣对象。每段视频的准确率以返回计数与真实标注计数之间的百分比差异来衡量。

随着时间推移，视频分析部署将面对其所管理的视频流上运行的各种工作负载，每种负载的查询组成和规模各不相同 [6, 74]。然而，总体目标始终不变：在资源约束下，以期望的响应速率提供低延迟结果，并实现尽可能高的准确率。

2.2 调整摄像头方向带来的机遇

现有的视频分析优化方法（§6）假设固定摄像头的方向（旋转和缩放）以及它从目标场景中采集的内容都是固定不变、无法调整的。为量化这一限制的重要性，我们在包含 50 段视频的数据集和多种工作负载上开展实验，这些工作负载涵盖 4 种模型架构、§2.1 中的 4 项任务，以及人和汽车；两者的详细信息见 §5.1。每段视频都支持调整旋转角度（水平范围为 150°、步长为 30°；垂直范围为 75°、步长为 15°）和缩放（1-3×）；我们还在 §5.4 中考察了其他粒度。

对于每段视频，我们将每个工作负载的查询运行于全部 75 个方向上，从而获得逐视频帧（此处为 15帧/秒）的结果。随后，我们将准确率定义为相对于

最佳 方向，即在每个视频帧上使工作负载逐帧准确率最高的方向。例如，对于计数，某一方向在任意时刻的准确率等于其关注对象数量除以该时刻所有方向中的最大计数。采用此方法，我们比较三种方案：(1)一次性固定 ，选择attime=0时的最佳方向，并在整个视频期间保持不变；(2)最佳固定 ，利用预知信息选择单一最佳方向，使视频的平均工作负载准确率最高

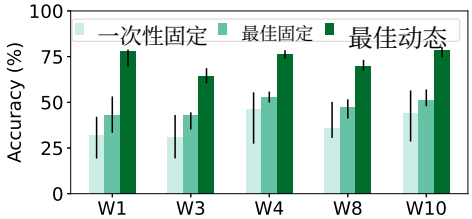


图 1：采用不同程度的朝向自适应时，5 种代表性工作负载的准确率。柱形图列出中位数视频的结果，误差线表示第 25 至第 75 百分位数。

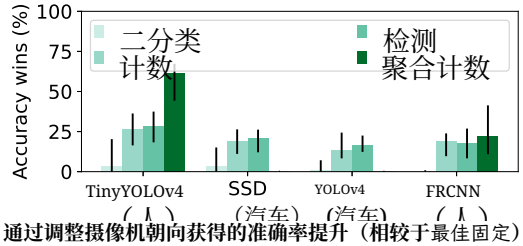


图 2：通过调整摄像机朝向获得的准确率提升（相较于最佳固定）会随着查询具体性的提高而增大。柱形图列出中位数视频的结果，误差线表示第 25 至第 75 百分位数。由于多目标跟踪器的限制（§5.1），我们排除了聚合计数+汽车。

；(3) 最佳动态，在视频的每个视频帧上选择最佳方向。

如图 1 所示，调整摄像机朝向能显著提升准确率，同时不会增加资源使用量，也就是说，传输和处理的视频帧数量保持不变：采用最佳动态方案时，中位数准确率相较于一次性固定方案提升 30.4%–46.3%，相较于最佳固定方案提升 21.3%–35.3%；后者是任何固定朝向方案所能达到的上限。图 2 按查询任务细分了这些结果。值得注意的是，随着查询类型变得更加具体，调整朝向的重要性也随之提高。例如，对于 YOLOv4 和汽车，相较于最佳固定方案，二分类、计数和检测的中位数准确率分别提升 1.2%、13.4% 和 16.4%。这是因为较粗略的查询会掩盖不同朝向之间的某些差异。例如，当场景中存在许多目标对象时，只要某个朝向能捕捉到一个目标对象，二分类就能达到最高准确率；而计数则更偏向于能捕捉到最多目标对象的朝向。

PTZ 摄像机简介。 云台变焦（PTZ）摄像机提供了一种直观的机制来实现这种适应。PTZ 摄像机分为两种：传统型 [31, 78] 和电子式（ePTZ）[46, 79], , 两者都支持通过软件调节水平旋转（水平旋转）、垂直旋转（垂直旋转）和缩放。两种类型的关键区别在于其调节机制。传统型 PTZ 摄像机内置物理电机，旋转速度可远超每秒 360°，并支持光学变焦（即，

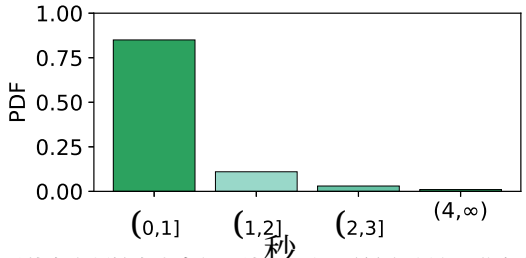


图3：最佳方向频繁发生变化。结果列出了所有视频和工作负载下，最佳方向之间切换间隔时间的概率密度函数（按1秒分箱）。

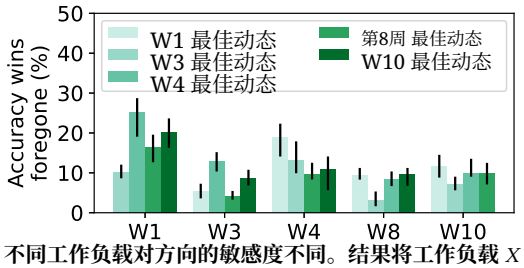


图 4：不同工作负载对方向的敏感度不同。结果将工作负载 X（图例）的最佳方向应用于工作负载 Y（x轴），并绘制因未对工作负载 Y使用最佳方向而损失的准确率提升（相对于 Y的最佳固定方向）。柱状图列出中位数；误差线表示第25至第75百分位数。

不会降低分辨率）。相比之下，电子云台摄像机拍摄宽视场，并通过近乎即时的数字旋转和缩放聚焦于场景的特定部分。电子云台摄像机改变方向的速度更快，价格也更低，但覆盖的旋转范围更小（150°，而非 360°），且数字缩放会降低图像质量。PTZ 摄像机在机载计算资源方面可与传统摄像机媲美，近期推出的产品还搭载了边缘级 GPU [71]。

2.3 挑战

尽管使用云台摄像机调整摄像机朝向具有潜在优势，但在实践中，三项根本性挑战使这种方法复杂化。下面依次介绍这些挑战。

C1：最佳方向随时间快速变化。 如图 3 所示，由于视频内容具有动态特性，最佳方向切换十分频繁：85% 的切换发生在上次切换后的 ≤1 秒内。

C2：不同工作负载对缩放和旋转的敏感度各异。 在任意时刻，针对不同查询和工作负载，最佳方向都可能不同。图 4 展示了这一点：为了使某一工作负载的准确率最大化而调整摄像机朝向，可能会导致其他工作负载损失 3.2–25.1% 的潜在（中位数）准确率提升。

图 5 从查询层面突显了这一点，显示不同模型、对象和任务都会影响方向选择。模型差异会影响推理过程中场景中可辨识的内容，以及适用的方向。例如，在人数统计任务中，为

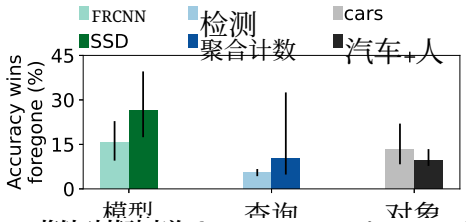


图 5：将针对基础查询 {YOLOv4, counting, people} 的最佳方向应用于仅修改基础查询中一个元素的查询 Y；我们将其准确率提升（相对于最佳固定方向）与使用针对 Y 的最佳方向时的准确率提升进行比较。柱形表示中位数；误差线表示第25至第75百分位数。

使用 YOLOv4 的查询选择最佳方向，相比使用 SSD 执行同一任务（即使在同一数据集上训练），会错失 26.3% 的中位数准确率提升。相比之下，任务决定了收集结果所需的具体程度；例如，使用 YOLOv4 优化人数统计，而非使用同一模型进行总人数统计，会放弃 10.2% 的潜在收益。最后，感兴趣对象决定了不同区域的重要性，这取决于物体密度，也决定了检测相关对象所用的特征及其难度（较小的对象通常更难辨识 [80]）。因此，不出所料，若将汽车而非人作为查询对象，针对 YOLOv4 人数统计查询进行优化将放弃 13.3% 的收益。

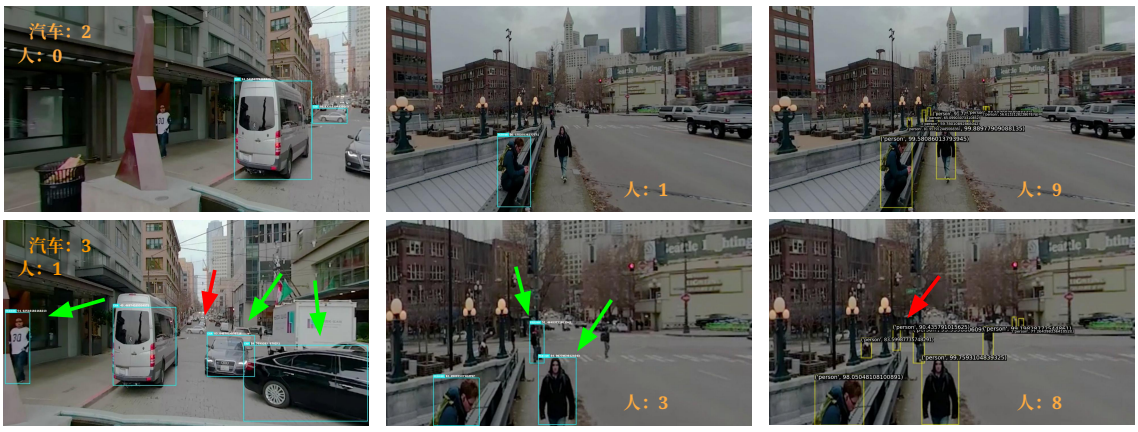
图 6 展示了示例截图，以说明改变方向的益处与弊端。重要的是，调整方向并非只是让新的物体进入视野；它还极大地影响模型检测已在视野中的物体的能力。

C3：庞大（但稀疏）的搜索空间。 方向空间在空间和时间维度上都具有显著的稀疏性。就空间维度而言，在任一时刻的 75 个方向中，只有 1 个（或存在并列时的多个）是最佳方向；其他方向的准确率则持续下降，例如，从最佳方向到第 2 和第 5 佳方向的准确率中位数分别下降 4.8% 和 20.7%。就时间维度而言，对于每个视频中的大多数方向，其成为最佳方向的累计时长都很短；各工作负载的时长中位数为 5–6 秒（图 7）。

3 设计

图 8 展示了 MadEye 的端到端运行过程。MadEye 的核心思路是利用云台俯仰变焦的快速旋转速度，在每个时间步（即两次需要帧率结果的时刻之间）探索多个方向，然后根据这些方向的当前内容，选择能够在资源约束下使工作负载准确率最高的一个或多个方向。与仅依赖过去方向效能的先前搜索算法（§5.3）相比，这种方法旨在减少“猜测”。

与其他视频分析系统一样 [28, 52, 60, 74], , 用户会向后端代理（运行于边缘端或云服务器上）注册查询，指定目标场景、所用模型、感兴趣的对象以及任务。为了满足摄像头的计算限制，MadEye 随后会训练适用于边缘端的（即



汽车 + 人 (SSD)　　人（SSD）　　人 (FRCNN)

图 6：展示旋转和缩放对不同查询产生（多样化）影响的截图。每列展示同一时间点拍摄的两张图像，使用的旋转或缩放各不相同。底部一行中，绿色箭头表示新捕获到的对象，红色箭头表示方向变化后新漏检的对象。左：旋转使一个新对象进入场景，帮助检测到 2 个先前可见的对象，但也丢失了一个先前已检测到的对象。中：放大有助于检测到新的人。右：切换模型后，中间列相同的缩放实际上减少了检测到的人数。

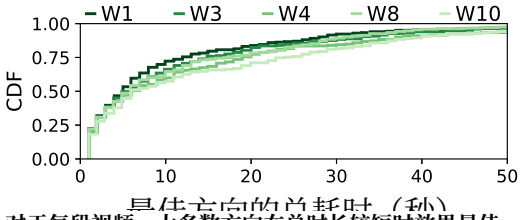


图 7：对于每段视频，大多数方向在总时长较短时效果最佳。结果涵盖每个工作负载中的所有方向-视频对。

高度压缩的）模型（§3.1），这些模型并非要取代原始的（更准确的）查询模型（如典型的知识蒸馏 [43]），而是大致提取视频帧中对每个查询而言重要的信息。换言之，近似模型专门用于估计每个查询固有的敏感性（§2.3 中的 C2）。

为应对方向空间庞大以及最佳方向快速变化的问题（§2.3 中的 C1 和 C3），MadEye 采用高效的摄像头端搜索策略（§3.3），在避免结果出现帧率违规的同时，尽可能探索有潜力的有效方向。随后，摄像头会在每个时间步对所有已捕获方向运行近似模型，并利用结果来（1）对方向进行排序，以评估其最大化整体工作负载准确率的可能性；（2）确定下一时间步要考虑的方向集合。网络能够支持的排序最高的方向会被发送到后端进行完整的工作负载推理；新结果则用于持续调整近似模型，以适应当前场景（§3.2）。

3.1 近似模型的设计

MadEye 的近似模型的主要目标是量化各个方向对于工作负载中查询的相对重要性。然而，这需要在摄像机计算约束下，捕捉每个查询对不同方向和场景动态的敏感性。考虑到工作负载查询可能十分复杂，我们不采用基于局部梯度的嘈杂（且有限的）视觉特征 [25, 66]，而是选择使用压缩模型进行知识蒸馏 [43]。不过，我们对这种方法做了几处调整，以更好地平衡排序准确率和资源效率。

我们使用一种通用抽象来设计近似模型，以反映充分排序各个方向所需的最少信息量。核心思路是，查询敏感度的关键要素在于模型如何查找和描述物体，而非任务如何对这些结果进行后处理。因此，MadEye 的近似模型纯粹采用超轻量级检测器的形式，用于检测感兴趣对象；这种策略也避免了为每项任务开发压缩模型的复杂工作。具体而言，我们使用边缘端 EfficientDet 系列中最小的变体 [94]，EfficientDet-D0（390 万个参数，在 Jetson 边缘 GPU 上的 >150 帧率）。虽然也可以使用更复杂的检测器，但摄像机的 GPU 内存有限，[60, 74]，而推理延迟会降低 MadEye 探索不同方向的能力（§3.3）。

为什么要使用检测器？我们考虑了两种近似模型的替代方案：直接估计图像中的物体数量，以及直接输出多张图像之间的排名顺序。然而，我们在实证中观察到，这两种方法的错误率都很高。这主要是因为，这些方法只能通过针对整张图像（或多张图像）的全局回归，将特征的存在与物体联系起来，无法利用基于边界框预测的局部回归来提高精确率。虽然图像级深度神经网络物体计数器

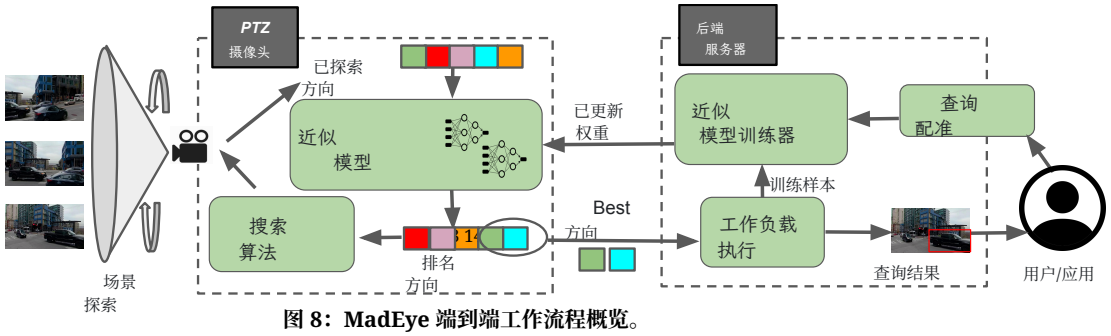


图 8：MadEye 端到端工作流程概览。

确实存在 [91, 104, 109, 113], , 但它们专注于人群密集的场景。相比之下，在任一时刻，某个方向上的感兴趣对象通常很少（§2.3），因此排名顺序对计数预测中的微小误差极为敏感。

MadEye 为每个查询使用一个近似模型，而不是为每个工作负载或对象分别使用模型。虽然这样效率更高，但我们没有采用按工作负载或对象划分的近似模型，因为我们（与其他研究者一样 [6]）发现，即使面对相同的对象类别，不同的深度神经网络也可能表现出截然不同的响应特征，这些差异源于尺度和分辨率等与对象无关的因素 [45]。此外，众所周知，在截然不同的数据集上训练的深度神经网络会继承不同的算法偏差 [10, 30, 56, 72, 92, 102]。

不过，出于两个原因，每个近似模型都配置为支持所有方向。首先，方向数量众多（§2.3），因此在摄像机端 GPU 上为每个方向分别建立近似模型并不切实际。其次，相邻方向之间存在大量重叠；而且，由于我们只考虑给定场景中的方向，背景内容、光照、阴影等方面的差异很小。事实上，我们测得同一场景中不同方向图像的感知距离 [81]（LPIPS）为 0.30。作为参考，用于成功预训练许多视觉模型（包括 EfficientDet）的热门数据集 MS-COCO 和 Pascal VOC，其对应数值分别为 0.46 和 0.41。

估算工作负载准确率。MadEye 对所有近似模型生成的边界框进行后处理，以计算用于视角排序的预测工作负载准确率。为此，MadEye 遵循 §2.1 中各任务的准确率指标，但以相对方式计算每个方向的预测准确率，并与当前测试的其他方向进行比较。例如，在计数任务中，会计算每个方向的物体数量与该时间步已探索方向集合中的最大物体数量之比；而检测任务则在此基础上纳入对象面积大小（依据 mAP 分数）。最后，聚合计数会调整计数分数，以优先考虑探索较少的方向（这些方向可能存在尚未发现的对象）。

3.2 持续训练近似模型

MadEye 服务器会针对每个新查询训练一个新的近似模型，目标是兼顾速度（因为训练会阻塞部署）和准确性（即对方向重要性进行排序）。初始训练使用来自目标场景的一小组 1000 张历史图像，随后使用已注册查询中的深度神经网络（DNN）在线标注；标签生成耗时 7–90 秒，具体取决于该 DNN。为加快这一流程，MadEye 以在 Pascal VOC 上预训练的 EfficientDet 版本为基础，并冻结负责特征提取与融合的骨干网络和 BiFPN 层。仅对最后 3 个边界框和类别预测层的权重进行微调，以模拟目标查询的行为。其依据是，随着层深度增加，模型特征会逐渐从通用特征（如纹理、梯度）转变为特定于任务的特征（如对象预测） [12, 107, 108]。初始微调持续 40 个训练轮次（≈25 分钟）。

即使经过初始微调，近似模型仍可能无法适应不断变化的场景动态 [93], , 导致准确率下降。为应对此类数据漂移，MadEye 采用持续学习（每 400–500 毫秒一次），利用发送到服务器进行完整工作负载推理的方向所对应的最新查询结果更新模型权重。尽管持续学习已应用于边缘视频分析 [4, 68],，MadEye 仍需对以往方法进行多项调整。主要挑战在于，在每个重新训练窗口内，只有 MadEye 的相机侧组件近期访问过并判定值得进行后端推理的方向，才有可用样本。由于各方向通常只有在累计时长较短时才表现最佳（§2.3），新训练样本所涵盖的方向往往严重失衡。例如，在排序完全准确的情况下，平均每个 2 分钟窗口中，只有 9.3% 的方向会被发送到后端。这可能导致模型对某些方向过拟合，并对其他可能很快获得较高排序的方向产生灾难性遗忘 [55] 。

为了解决这一问题，MadEye 会从每个方向中提取最近的历史训练样本，并用它们来平衡数据集。正如我们将在 §3.3 中讨论的，我们发现方向变化通常局限于局部空间，而距离较远的方向变化所需的时间更长。因此，MadEye 会为相邻方向的数据样本补齐数量（最多延伸到距最新方向 3 个方向的位置），使其与重新训练窗口中最常见方向的计数相匹配。

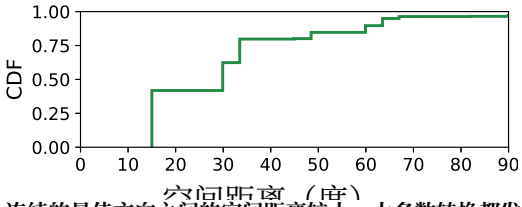


图 9：连续的最佳方向之间的空间距离较小，大多数转换都发生在相邻方向之间。结果汇总了所有视频和工作负载在 15帧/秒下的数据。

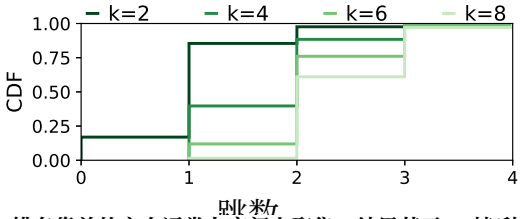


图10：排名靠前的方向通常在空间上聚集。结果基于 15帧/秒，汇总了所有工作负载和视频的数据，并显示每个时间步排名前 k 个方向之间的最大距离。

remaining 或i向 i采用指数 ially declini样本数量b ，具体取决于它们与最新方向之间的距离。

3.3 探索与排序方向

MadEye 的相机端组件主要目标是高效探索庞大的方向空间（或其子集），以便在每个时间步捕获最佳方向。要实现这一目标颇具挑战，原因有三。首先，MadEye 只能看到近期探索过的方向，但其他方向的内容和重要性随时可能变化。其次，即使对于近期探索过的方向，MadEye 对大多数方向也只能获取近似模型给出的粗略结果（即只能准确反映相对重要性）。第三，每个时间步不仅用于探索，还要（1）对已探索的方向运行近似模型，（2）对选定方向进行编码并发送到服务器，以及（3）在已发送的图像上运行工作负载。

MadEye 不依赖每个方向上先前（且可能已过时）的观测结果（§5.3），而是采用一种由 3 项实证观察指导、信息更充分的策略。

- 尽管最佳方向会随时间快速变化（§2.3），但这些变化在空间维度上要缓慢得多。图 9 展示了这一点：连续两个最佳方向之间的空间距离的中位数和第90百分位数分别为 30° 和 63.5°，这相当于在我们的默认网格（§5.1）中仅跨越 1 或 2 个方向。

- 任意时刻表现最佳的方向（准确率方面）通常在空间上呈聚类分布（图 10）。具体而言，在我们的整个数据集中，每个时间步排名前 k 的方向之间距离的第75百分位数，对于 k 值为 2 和 6 的情况，分别为 1 个和 2 个方向。

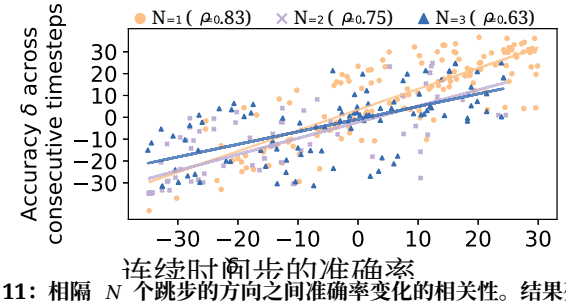


图 11：相隔 N 个跳步的方向之间准确率变化的相关性。结果列出了皮尔逊相关系数，涵盖 3 个代表性视频和工作负载（15 帧率）。

- 相邻方向的准确率往往会同步变化。事实上，如图 11 所示，直接邻居的准确率变化相关系数为 0.83；直观而言，考虑相隔 2 跳的邻居时（其内容重叠较少），这一数值会降至 0.75。

综合来看，这些发现促使我们采用一种搜索策略：在每个时间步考虑由连续方向构成的灵活形状，并且仅将前一形状中表现不佳的方向替换为其邻居方向，因为我们能够稳健地预测这些邻居方向在下一时间步的趋势。我们首先介绍一种不考虑缩放或资源约束的算法，之后再纳入这些因素。共同特点包括：仅使用近似模型结果的相对比较，充分利用这些模型的所有输出（包括边界框），并且搜索决策完全在本地（即摄像头上）进行，以保持快速响应。

MadEye 从一个矩形种子形状开始，该形状反映了时间预算内可覆盖的最大区域，从而最大化早期探索；每当在某个形状中发现 0 个感兴趣对象时，我们就会重置为该形状。随后，利用近似模型捕获并分析相应的方向，以计算每个方向的预测工作负载准确率（§3.1）。将排名最高的 k 个方向发送到服务器进行工作负载推理后，MadEye 必须利用这些先前的结果，确定下一时间步要探索的方向集合。

为此，MadEye 会为上一个时间步的每个方向标注一个值，表示该方向在下一个 时间步中产生有效结果的可能性。具体而言，我们会对近期（10 个）时间步中的指数加权移动平均值进行组合，分别针对：(1) 已计算出的预测准确率值，以及 (2) 这些值之间的变化量。采用加权平均值可以增强鲁棒性，避免深度神经网络的结果在连续帧之间出现不一致 [6, 76], ；使用 MadEye 的压缩近似模型时，这种不一致尤为明显。

根据这些标签，MadEye 现在必须确定在即将到来的时间步中要移除和添加哪些方向。为此，MadEye 根据标签值对方向排序，形成有序列表。MadEye 使用指向列表头部 H（最大标签值）和尾部 T（最小标签值）的指针，迭代地

比较方向，并判断：是否应移除 T处的方向，以便为 H添加一个邻居？具体而言，MadEye 计算 H/T的标签值之比。如果（1）该比值超过某个阈值（表明 H和 T的潜力存在显著差异），（2） H有尚未位于形状中的邻居，并且（3）移除 T不会破坏连通性，那么我们就移除 T 处的方向，并递增指针。随后，流程继续尝试为 H添加另一个邻居，此时使用更大的阈值，以考虑添加更多邻居所带来的额外不确定性。若无法添加邻居，则递减 H；当连一个邻居都无法为 H 添加时，流程结束。

每次迭代为H添加邻居时，MadEye 都会分析近似模型在上一个时间步生成的边界框，从 H的邻居中进行选择。对于每个候选邻居，我们计算两个值的比率：到 H中心的标准距离，以及到 H中所有边界框质心的标准距离。值 <1 越大， H 的物体在下一个时间步移动到该候选邻居的可能性越低。对于候选邻居与上一形状存在非零重叠的其他所有方向，我们都会重复此过程。候选邻居得分是这些比率的加权和（权重根据重叠程度确定），并选择得分最高的候选邻居。

可达性与路径选择。 到目前为止，搜索算法尚未考虑云台摄像机能否在给定时间预算内充分覆盖所选形状。形式上，方向的形状可表示为一个全连通无向图，边权重表示在给定旋转速度的情况下，两个相邻方向之间移动所需的时间。我们的目标是确定该形状能否在给定时间预算内覆盖；如果可以，则确定最短路径。方向之间的路径满足三角不等式性质 [95], ，因此可以将此问题建模为 NP难旅行商问题（旅行商问题） [14]的一个变体。鉴于时间预算紧张，MadEye 采用最小生成树（MST）启发式方法 [42], ，并对其进行优化以尽量减少在线延迟。具体而言，由于方向网格是静态的，我们会预先计算成对距离和整个最小生成树。在线运行时，对于给定形状，我们会快速提取相应子图并对其执行先序遍历，以获得最短路径。这使该启发式方法的复杂度降至线性复杂度（相对于方向数量）；每次路径计算耗时 14 μs，所得路径长度不超过最优值的 92%。若失败，MadEye 会贪心地移除潜力最低且不会破坏连通性的方向，然后重新检查可达性。

平衡搜索规模与网络/计算延迟。Mad-Eye 将方向探索与在每个方向上运行近似模型流水化处理。然而，向后端进行网络传输以及在后端进行工作负载推理，都不会与方向探索重叠。原因是

因此，传输由每个时间步中所有已探索方向的全局排名决定。因此，在每个时间步中，我们都面临着一个权衡：是探索更多方向，还是向后端发送更多方向。

MadEye 根据其近似模型准确排列所考虑方向的预期难度来化解这一矛盾，而这又决定了探索更多方向（并减少发送以获取真实结果的方向数量）所伴随的风险。直观而言，当所考虑方向预计能带来相近的准确率时，近似模型面临的难度最大（因为排名之间的差距会缩小）。MadEye 首先根据近似模型的训练准确率（由后端提供）以及上一个时间步中预测准确率值的方差，选择要发送的视频帧目标数量，以此确定恰当的平衡。例如，训练准确率为 85%、方差为 25% 时，至少发送 2 帧。随后，MadEye 会计算用于探索的目标形状大小，并考虑网络传输延迟（过去 5 次传输的调和平均数 [106]）、后端计算延迟、摄像头旋转速度以及近似模型推理延迟。

处理缩放。选定要访问的方向后，搜索算法必须确定每个方向使用的缩放倍率。难点在于，过去的准确率不足以确定缩放的精细程度，因为 MadEye 无法知道不进行放大或缩小时会漏掉哪些物体。因此，我们依靠近似模型提供的边界框来判断放大的风险。将某个方向加入形状时，我们会从最低缩放倍率开始，以观察其全部内容。在每个时间步，我们计算每个边界框与所有边界框质心之间的平均距离；距离越小，表示物体聚集程度越高，放大风险越低。我们将这些数值与每种缩放倍率覆盖的面积进行比较，以选定合适的倍率；为避免漏掉进入该方向的新物体，MadEye 会在 3 秒后自动缩小。

传输图像。在每个时间步结束时，MadEye 必须将选定的图像传输到服务器，以供工作负载推理使用。与标准流式传输不同，MadEye 会从每个方向的视频流中发送互不重叠的图像集。为了降低带宽成本，MadEye 会维护每个方向最近一次共享图像的列表，并使用函数式编码器 [34] 计算相对于该图像的差异。

4 实现

MadEye 的核心组件由 9.1k 行 Python 代码编写，后端和摄像机上的所有训练与推理任务均在 PyTorch 中运行。我们使用 TensorRT [2]加速后端上的推理，并采用 Nexus [88] 的一个变体作为轮询调度器，用于摄像机上的近似模型推理。方向首先表示为旋转

值，投影到 360° 空间中，然后使用内部开发的等距柱状投影到直线投影图像转换器（以 C++ 编写）进行转换，以匹配新款云台摄像机 [13]提供的 API。对于需要从全局（即涵盖所有方向）视角计算真实标注准确率（§5.1），以确定物体位置及其唯一性的任务，除了使用 ByteTrack 多物体跟踪器 [105]关联同一视频不同方向上的物体外，我们还使用 cv2 和 scikit-image 提取图像特征（例如尺度不变特征变换），以关联不同方向上的物体。

5 评估

我们在多种工作负载、网络设置和视频上评估了 MadEye。主要发现如下：

• 与预知最优固定朝向策略相比，MadEye 将工作负载准确率的中位数提高了 2.9-25.7%（资源用量相同）；其准确率提升与预知最优动态策略相差不超过 1.8-13.9%。

• 若要用 1 台云台摄像机达到 MadEye 的准确率提升，需要使用效果最佳的 4-6 台固定朝向摄像头，这会使资源成本增加 2-3.7× 。

• MadEye 的表现比以往的云台俯仰变焦算法高出 2.0-5.8×，其准确率分别比 Panoptes [90], 跟踪 [85], 和多臂老虎机算法 [97]高 31.1%、46.8% 和 52.7%。

• 即使资源减少、响应速率上升，MadEye 仍能灵活平衡摄像机端探索与方向传输，以最大化准确率。

5.1 方法

用于云台摄像机分析的视频数据集。 据我们所知，目前不存在供用户调节旋转和缩放参数的公开云台摄像机视频数据集；现有的云台摄像机数据集反映的都是预先确定的参数设置。因此，为了评估 MadEye，我们自行构建了一个数据集。具体而言，我们从丰富的 360度视频数据集中选取素材，使用了 YouTube 上的 50 个 360 度视频，这些视频包含与先前研究相似的场景 [4, 28, 60], , 例如交通路口、步道和购物中心。每段视频时长为 5-10 分钟。

我们从每段视频中截取感兴趣的场景，形成水平视野为 150°、垂直视野为 75° 的区域。随后，将这些区域划分为不同方向的网格，以模拟近期云台摄像机产品的配置 [46]（水平旋转和垂直旋转的粒度分别为 30° 和 15°；我们在 §5.4 中探讨其他网格划分方式），并为每个方向提取一段完整视频。由于我们使用的是预先拍摄的视频，缩放方面采用数字变焦（1-3×），通过裁剪图像并缩放回原始图像尺寸来实现。

模型与工作负载。 我们考虑了 4 种用于视觉任务的常见架构：采用 ResNet-50 主干网络的 SSD [64] 和 Faster RCNN [87] ，以及采用 CSPDarknet53 主干网络的 YOLOv4 和 Tiny-YOLOv4 [99]。每种模型我们都考虑两个版本分别在 Pascal VOC 和 MS-COCO 上训练，但由于两者的趋势相似，我们仅展示后者的结果。在构造查询时，我们采用近期工作（基于生产部署） [74]中的相同方法。每个模型都可以执行 §2.1 中的四种任务，重点关注人或汽车。我们枚举查询数量在 2–20 之间的所有可能工作负载，并随机挑选 10 个。附录详细介绍了各项工作负载。我们在所有视频上运行这些工作负载，并考察 1–30 帧率的响应速率。

硬件和网络。 摄像头上的计算在边缘级 Jetson Nano [71] 上运行，该设备配备 128 核 Maxwell GPU、主频为 1.43 GHz 的四核 ARM CPU，以及 4 GB 内存。我们将默认摄像头旋转速度设为每秒 400°；并在 §5.4 中研究这一参数。工作负载推理和近似模型的训练在服务器上运行，该服务器配备 NVIDIA GTX 1080 GPU（8 GB RAM）和 18 核 Intel Xeon 5220 CPU（2.2 GHz；125 GB RAM）。摄像头和服务器组件通过模拟的 Mahimahi 网络 [70]连接，使用固定容量（24-60 兆比特每秒；5-20 ms）和真实世界的移动网络轨迹。

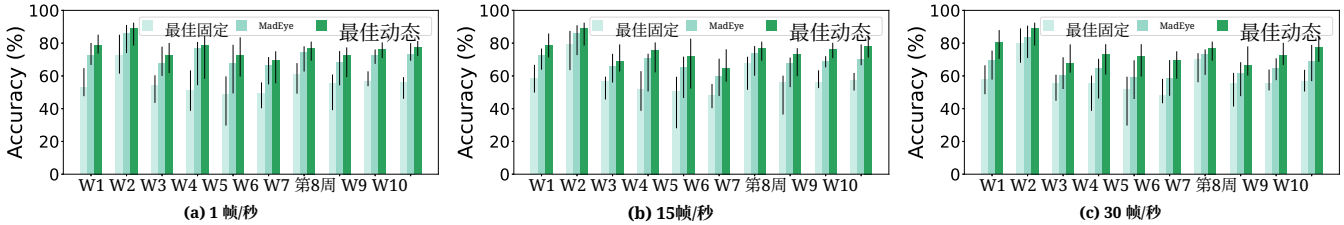
指标。 我们的主要评估指标是每个视频的平均工作负载准确率。对于每个视频帧，按照 §2.1 中的准确率定义，我们针对每个查询计算各方向上的准确率，并以当时能够达到最高准确率的方向作为参照。对每个时刻各查询的准确率取平均，得到每视频帧的工作负载准确率；再对这些准确率取平均，得到视频的工作负载准确率。

对于二分类和计数，计算这些值较为直接；而检测和聚合计数则需要稍作调整。对于检测，mAP 分数取决于特定物体的边界框坐标，因此无法通过直接比较不同方向下的结果来衡量。为此，我们将各方向的边界框整合为全局视图，并采用去重方法 [75]消除重叠区域中的冗余物体。随后，我们以全局场景为参照，计算各方向的 mAP 分数，并将每个方向的准确率设为其 mAP 分数与所有方向中最高 mAP 分数的比值。

聚合计数查询直接在整个视频范围内进行评估（而非逐视频帧评估）。因此，我们计算系统所选方向下唯一目标的数量与视频中唯一目标总数的比值。请注意，ByteTrack（§4）无法稳定支持汽车跟踪，因此我们不对汽车进行聚合计数。

5.2 总体结果

我们首先在不同的网络和帧率设置下，将 MadEye 与 §2.2 中的两种基线方法进行比较：最佳固定和 最佳动态。这两种基线方法都不切实际地依赖于对视频内容和工作负载准确率的预知信息，即选择最佳



(a) 1 帧/秒　(b) 15帧/秒　(c) 30 帧/秒

图 12：在 {24 Mbps, 20 ms} 网络下，通过改变帧率，比较 MadEye 与所有视频和工作负载中最佳的固定与自适应方向方案。柱形图列出中位数，误差线范围为第25至第75百分位数。

每个视频或每个时间步的方向，分别是指能够最大限度提高目标工作负载视频准确率的方向。尽管如此，它们仍为了解 MadEye 的性能提供了有用背景。请注意，MadEye 会根据网络延迟和响应速率自动调整其探索和传输的帧数（§3.3）。对于 最佳固定，我们通过增加固定摄像头的数量（即最佳、第二佳等）来利用网络速度的提升，而不是仅从一个摄像头捕获更多（冗余）帧。最佳动态 不会因聚合计数以外的任何查询而改变；对于聚合计数，我们会发送网络所能支持的最大数量的有效方向。

我们的结果如图12至图13所示。在这些设置下，MadEye 的中位数准确率和第75百分位数准确率分别比最佳固定高 2.9-25.7% 和 1.6-20.7%，与最佳动态相差 1.8-13.9% 和 1.3-12.5%。进一步分析后，我们的结果显示出两个关键趋势。首先，当帧率降低时（网络固定），MadEye 的准确率以及相较于最佳固定的优势会扩大。例如，对于 {24 Mbps, 20 ms} 网络，当帧率从 15 降至 1 时，中位数优势从 5.8-13.3% 提高到 12.4-25.7%。原因在于，较低的帧率会带来更长的时间步（例如，1 帧/秒对应 1 秒，15 帧/秒对应 66.7 毫秒），从而能够进行更多探索和/或传输。其次，随着网络速度提高（帧率固定），同样的趋势仍然存在（因为每次网络传输都更快），但程度较轻。例如，在 {60 Mbps, 5 ms} 网络下，15 帧/秒时的中位数优势提高到 8.6-18.4%。

图14按任务和对象细分了 MadEye 相较于最佳固定摄像头的优势。根据 §2.2 中的原理，随着任务特异性增强，MadEye 带来的准确率提升也会增大：对于人的查询，从计数到检测再到聚合计数，中位数优势依次从 8.6% 增至 13.3% 和 22.1%。我们还发现，相较于汽车查询，人的查询始终能获得更大的准确率提升，这是因为人的运动模式结构性较弱（方向切换更频繁且更分散）。例如，在检测任务中，汽车的优势缩小至 6.7%。

迄今为止的结果主要关注准确率提升。然而，MadEye 的一个关键目标是在给定资源成本（即网络和后端推理开销）的情况下最大限度地提高准确率。表1列出了为达到 MadEye 不同版本所实现的准确率而需要的、经过最优配置的固定摄像头最少数量；每个版本在每个时间步发送的帧数各不相同。如表所示，它将

MadEye 变体	中位数准确率（%）	# 固定摄像头数量
MadEye-1	63.1	3.7
MadEye-2	66.3	5.5
MadEye-3	66.8	6.1

表1：达到 MadEye 准确率所需的经过最优配置的固定摄像头数量。MadEye-*k* 指的是一种受限版本的 MadEye，每个时间步仅向服务器发送排名前 *k* 的帧以进行工作负载推理。结果基于 {24Mbps; 20 ms} 网络、15帧/秒的响应速率以及所有视频-工作负载对。

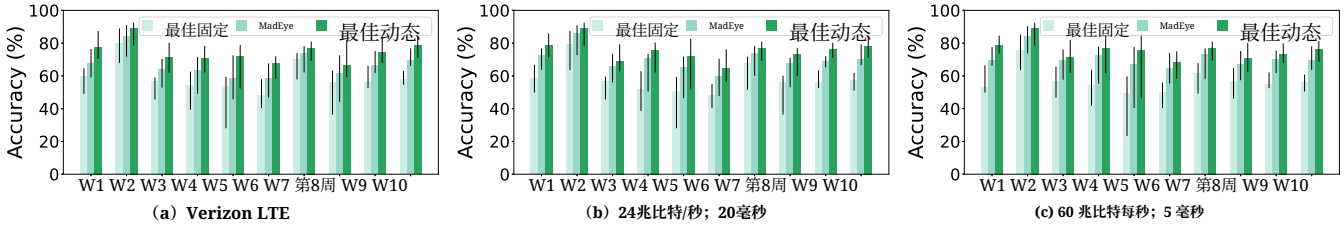
需要采用 3.7 个固定摄像头，才能达到 MadEye-1 实现的 63.1% 准确率，这意味着网络和后端计算用量可削减 3.7× 倍。要达到 MadEye-2 的效果，则需要 5.5 个固定摄像头；不过，在此情况下，资源削减倍数为 2.8×，因为 MadEye 每个时间步还会发送 2 帧。

5.3 与最先进方法的比较

我们将 MadEye 与 3 种用于自适应摄像头朝向的替代方法进行比较。图15展示了在 {24 Mbps; 20 ms} 网络和 15帧/秒条件下的结果；在所有其他场景中，趋势也保持一致。

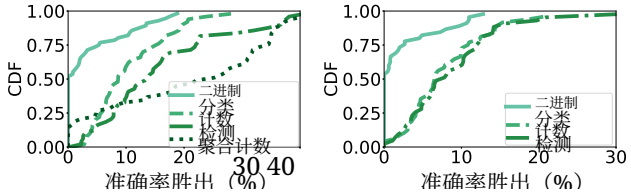
首先，我们考察 Panoptes——一种较新的云台变焦系统，它为应用程序的工作负载配置方向，每个工作负载都明确关注特定的方向。对于相关方向，Panoptes 会生成一个静态轮询调度，其权重取决于某个方向受到多少查询关注，以及历史上在该方向检测到的运动量；权重越高，表示在该方向停留的时间越长。随后，Panoptes 按照该调度在各方向之间切换，但有一个例外：如果朝向任何重叠关注方向的运动梯度超过阈值，Panoptes 就会切换到该方向并停留数秒，然后恢复轮询。Panoptes 未指定变焦策略，因此我们针对它访问的每个方向，考虑准确率方面最佳的缩放级别。

我们考虑 Panoptes 的两个版本，*Panoptes-all* 和*Panoptes-few*，其中每个工作负载查询分别关注所有方向，或仅关注其最佳固定方向。在这两种情况下，最大准确率均定义为相对于所考虑方向集合中最佳方向的准确率。如图15所示，MadEye 的表现优于 Panoptes-all，准确率提升了 3.8×，中位数准确率高出 46.8%。原因是，Panoptes 根据预先设定的调度和当前方向上的运动梯度在各个方向间轮换，而这两者都不足以表明当前时刻其他方向的重要性，例如，这些方向在大多数时候都不是最优的（§2.3）。相



（a）Verizon LTE　（b）24兆比特/秒；20毫秒　(c) 60 兆比特每秒；5 毫秒

图13：在固定帧率（15）和不同网络条件下（从左到右逐步改善），比较 MadEye 与所有视频和工作负载中可能实现的最佳固定方向方案和自适应方向方案。柱形表示中位数，误差线覆盖第25至第75百分位数。



(a) 人　**(b) 汽车**

图14：MadEye 相较于最佳固定方案在不同查询任务和对象上的准确率提升。结果涵盖所有视频和模型，并采用 15帧/秒和 {24兆比特/秒；20毫秒}。

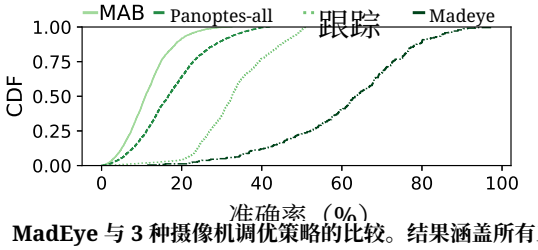


图15：MadEye 与 3 种摄像机调优策略的比较。结果涵盖所有工作负载和视频，采用 15帧/秒和 {24兆比特/秒；20毫秒}。

比之下，MadEye 在每个时间步都会考虑多个方向，并根据当前内容对它们进行排序。与 Panoptes-few 相比（由于准确率指标不同，结果未显示），MadEye 仍然表现更好，但优势较小（中位数为 40.5%），因为 Panoptes 需要考虑的无效方向更少。

接下来，我们考察如今大多数云台俯仰变焦摄像机都配备的跟踪算法 [85]。该算法从初始区域（在我们的实验中为最佳固定区域）开始，选择检测到的最大对象，并在不同方向上持续跟踪该对象，尽可能使其保持在中心位置。丢失跟踪对象后，算法会重置到初始区域。我们还考察了一种有利的变体：每个时间步探索的所有方向都会共享给后端，由后端选用准确率最高的方向。如图所示，与该跟踪方案相比，MadEye 的工作负载准确率提高了 2.0×（中位数提高了 31.1%）。主要原因仍然是，大对象的存在并不能很好地指示准确率的重要性，因为它无法捕捉更普遍的

系统	资源削减	中位数准确率
Chameleon [52]	2.4×	46.3%
Chameleon + MadEye	2.4×	56.1%

表2：MadEye 在保留近期系统资源节省的同时，提高了准确率。结果采用 15帧/秒、{24兆比特/秒；20毫秒}。

场景属性以及待测查询的敏感度。相比之下，MadEye 会直接估算查询敏感度，从而进行工作负载感知的明智方向选择。

最后，我们考察常见的 UCB1 多臂赌博机（MAB）算法 [97]。每种方向都被视为一条臂，其权重设为过去所有访问中观测到的平均准确率（我们用历史数据初始化该值）。算法会持续选择加权平均值与置信上界之和最高的方向进行访问（这会优先选择访问次数较少的方向）。与跟踪一样，我们将所有已访问的方向发送到后端，由后端在每个时间步选出最佳方向。MadEye 的中位准确率比该方案高 52.7%，即提高了 5.8×。与上述方案不同，MAB 在选择方向时确实会考虑工作负载准确率。然而，它的自适应仅考虑历史效能（而非当前内容），等到它更新模式时，场景动态早已发生变化。

与其他优化的兼容性。 MadEye 通过调整之前未调优的参数（旋转和缩放）来提高准确率，因此与此前旨在优化资源开销的工作基本兼容。为说明这一点，我们考虑 Chameleon [52] 的一个变体：该变体动态调整流水线参数（分辨率和帧率），在不损害准确率的前提下降低网络和后端推理资源成本；我们通过暴力搜索，为每个视频帧选择最佳固定方向。随后，我们在 Chameleon 选择的帧率和分辨率配置上运行 MadEye，并发送相同数量的网络数据。如表2所示，与发送所有最高分辨率视频帧的朴素方案相比，Chameleon 将资源成本降低了 2.4×；MadEye 在保留这些效率优势的同时，将准确率提高了 9.8%。

5.4 深入分析结果

旋转速度。 我们通过考察每秒 {200, 400, 500, infinite}° 的取值，以及固定网络（{24

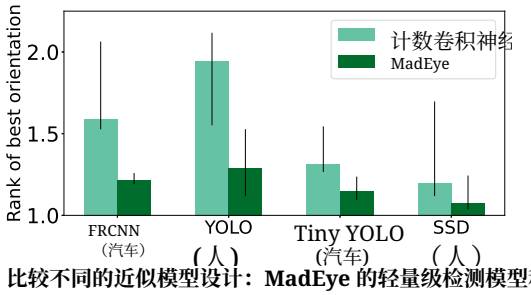


图16：比较不同的近似模型设计：MadEye 的轻量级检测模型和压缩计数模型（计数卷积神经网络）。结果使用所有视频、{24 Mbps; 20 ms}、15帧/秒，并列出每个时间步分配给最佳已探索方向的中位排名（误差线表示第25至第75百分位数）。

兆比特每秒；20 毫秒}) 和 15帧/秒。直观而言，随着旋转速度提高，准确率也会提升，例如，旋转速度从每秒 200° 增至 500° 时，准确率从 54.2% 升至 64.9%。这是因为更快的旋转可以探索更多方向，或者在较少见的情况下进行额外传输。值得注意的是，收益会逐渐趋于平稳，因为只要 MadEye 在每个时间步找到最佳方向，除聚合计数外的大多数查询在准确率方面就已完全满足。

网格粒度。为了了解网格粒度的影响（其他设置保持不变），我们聚焦于水平平移维度（因为它更宽），并考察 {15, 30, 45, 60}° 的步长。总体而言，随着网格变得更加细粒度（方向数量增加），MadEye 的准确率收益会缩小；例如，当水平平移步长从 45° 降至 15° 时，中位数准确率从 67.5% 降至 51.8%。这是因为，尽管时间预算内的探索由旋转速度而非网格粒度决定，但相同距离（以°计）的探索会在更多方向上进行更多近似模型推理，从而缩减每个时间步的探索预算。

开销。在 MadEye 的后端，主要开销来自初始化近似模型以及持续向摄像头共享模型更新。在各项工作负载中，我们发现启动延迟的中位数为 27 分钟（包括标注和初始微调）。下行流传输在中位数实验中消耗 3.2 兆比特每秒。请注意，MadEye 的微调策略（§3.2）可降低这两项开销。在摄像机上，主要开销来自选择要探索的方向以及运行近似模型；对于中位数工作负载-视频对，每个时间步执行各项任务的延迟分别为：在 15帧/秒和 {24 Mbps; 20 ms} 条件下为 17µs 和 6.7 毫秒。前一项受益于预计算的可达性分析（§3.3）。

微基准测试。MadEye 的性能主要由两项任务决定：(1) 使用近似模型对方向进行排序；(2) 选择要探索的方向，以便在每个时间步找到最佳方向。对于前者，图16显示，MadEye 的近似模型在每个时间步为已探索的最佳方向赋予的中位排序为 1.1–1.3，显著优于直接依据图像计数的变体。对于后者，在 {24 Mbps; 20 ms} 和 15帧/秒的设置下，对于中位数工作负载-视频对，MadEye 有 89.3% 的时间会探索最佳方向，其中 6.8% 的错误源于我们采用的保守变焦策略（§3.3）。

6 相关工作

调整视频分析参数。VideoStorm [111] 针对每个工作负载选择输入参数配置（例如帧率、分辨率），以降低资源成本并便于在后端服务器上进行作业调度。Chameleon [52]将此类配置调优扩展为动态调整，以应对不断变化的场景动态，同时保持较低的资源成本。如 §5.3 所示，MadEye 专注于调整摄像机朝向（而非后端流水线参数），为这些工作带来了互补优势：在保留其资源效率优势的同时提升准确率。其他工作则像 MadEye 一样关注摄像头侧参数。例如，Cam-Tuner [77] 使用 SARSA 强化学习，通过自动调节摄像头通常不会自动调整的采集参数（如亮度、对比度和锐度）来提高准确率。AccMPEG [29] 预测宏块编码设置对服务器端深度神经网络的影响，并调整编码以最大限度地提高准确率。MadEye 与这些工作目标相同，即通过调整摄像头参数来提高工作负载准确率，但关注的是互补参数，也就是摄像机朝向。

视频帧过滤与结果复用。许多先前的研究通过过滤掉需要网络传输和处理的视频帧，并相应地复用结果，来利用视频数据中的时间冗余 [9, 18, 21, 24, 27, 37, 38, 58, 60, 76, 103, 112, 115]。Spatula [48] 将这一方法扩展到多摄像头场景，在网络中的摄像机之间进行选择。这些优化在逻辑上与 MadEye 类似，后者同样旨在最大限度地提高单位网络使用量的准确率。不过，这些技术在很大程度上是互补的：可以在已探索的方向中进行过滤决策，以最大限度地增加传输中的新内容。

计算与网络优化。一些研究通过识别轻量级模型变体 [15, 23, 39, 43, 47, 65, 84, 114],、在推理过程中共享模型层 [50, 74]，或采用更智能的作业调度策略 [88, 111]，来降低计算开销。其他系统则通过智能压缩传输的视频帧来降低网络开销，使其能够在服务器上还原，或避免对准确率产生负面影响 [28, 32, 101]。MadEye 与这两个方向完全互补，因为它只专注于为应用提供的模型（该模型可以经过压缩）审慎选择任意时刻要处理的图像（即方向）；MadEye 不受所选视频帧在后端的传输或处理方式影响。

无人机协同。许多工作旨在调整无人机飞行计划（从而调整机载摄像头拍摄的内容），以

最大限度地提高分析准确率或场景覆盖率 [11, 41, 49,96]。然而，这些系统专注于在地理上分散的区域内识别感兴趣事件（例如野火、物体），以满足预设应用的需求。相比之下，MadEye 专注于调整单一场景中的摄像机朝向，以适应工作负载差异并最大限度地提高准确率。

7 结论

本文介绍了 MadEye，这是一种通过持续调整云台摄像机朝向，在给定分析工作负载和资源配置下最大化准确率的系统。MadEye 的核心包括一种快速算法，可在每个时刻搜索庞大的朝向空间；以及一种新的近似迁移学习策略，能够高效地从已探索的朝向中选出最有望提高准确率的朝向。在多种视频、工作负载和资源条件下，对于相同的资源使用量，MadEye 可将准确率提高 2.9-25.7%；或者在达到相同准确率的同时，将资源成本降低 2.0-3.7×

参考文献

[1] 2020. 视频分析流量研究为变革奠定基础。https : / / www . govtech . com / analytics / Video-Analytics-Traffic-Study-Creates-Baseline-for-Change.html. [2] 2021. NVIDIA TensorRT。https://developer.nvidia.com/tensorrt [3] 2021. PyTorch-YOLOv3。https://github.com/eriklindernoren/PyTorch-YOLOv3. [4] 2022. Ekya：边缘计算服务器上的视频分析模型持续学习。发表于第 *19* 届 *USENIX* 网络系统设计与实现研讨会（*NSDI 22*）。USENIX协会，华盛顿州伦顿，119–135。https://www.usenix.org/conference/nsdi22/ presentation/bhardwaj [5] 2022. 视频分析市场。https://www.fortunebusinessinsights.com/industry-reports/video-analytics-market-101114. [6] Neil Agarwal 和 Ravi Netravali。 [n.d.]. Boggart：迈向回溯式视频分析的通用加速。[7] Ganesh Ananthanarayanan、Victor Bahl、Landon Cox、Alex Crown、Shadi Nogbahi 和 Yuanchao Shu。2019年。视频分析——边缘计算的杀手级应用。发表于第 *17* 届移动系统、应用程序与服务国际年会论文集（大韩民国首尔）*(MobiSys'19)*。计算机协会，美国纽约州纽约市，695–696。https: //doi.org/ 10 1145 3307334 3328589 . / . Larry Anderson。n.d. . 零售领域的视频分析应用——Beyond Security。[8] [ ] https://www.securityinformed.com/insights/co-2603-ga-co-2214-ga-co-1880-ga.16620.html/. [9] Kittipat Apicharttrisorn、Xukan Ran、Jiasi Chen、Srikanth V. Krishnamurthy 和 Amit K. Roy-Chowdhury。2019年。节能跟随：面向移动增强现实的低功耗目标检测与跟踪。发表于第 *17* 届嵌入式网络传感器系统会议论文集（纽约，纽约）*(SenSys '19)*。计- 算机协会，美国纽约州纽约市，96–109。https://doi.org/ 10 1145 3356250 3360044 . / . Guha Balakrishnan、Yuanjun Xiong、Wei Xia 和 Pietro Perona。[10] 2021年。迈向人脸分析算法偏差的因果基准测试。发表于基于深度学习的人脸分析。施普林格，327–359.[11] Favyen Bastani、Songtao He、Ziwen Jiang、Osbert Bastani 和 Sam Madden。2021年。SkyQuery：无人机航拍视频感知平台。发表于*2021*年 *ACM SIGPLAN* 国际研讨会论文集关于新思想、新范式与编程和软件反思（美国伊利诺伊州芝加哥）（*Onward! 2021*）。计算机协会，美国纽约州纽约市，56–67。https://doi.org/10.1145/3486607.3486750[12] David Bau、Bolei Zhou、Aditya Khosla、Aude Oliva 和 Antonio Torralba。2017年。网络剖析：量化深层视觉表征的可解释性。载于*IEEE*计算机视觉与模式识别会议论文集。6541–6549。[13] Francois Beaufort。[无日期]。控制摄像头的水平旋转、垂直旋转和缩放——摄像头上的云台变焦功能终于可以在网络上控制了。https://web.dev/camera-pan-tilt-zoom/。[14] Michael A Bender 和 Chandra Chekuri。1999年。参数化三角不等式下旅行商问题的性能保证。载于算法与数据结构：第*6*届国际研讨会 *WADS'99*，加拿大温哥华，*1999*年*8*月*11*日至*14*日，论文集第*6*卷。施普林格，80–85。[15] Davis Blalock、Jose Javier Gonzalez Ortiz、Jonathan Frankle 和 John Guttag。2020年。神经网络剪枝的现状如何？*arXiv*预印本 *arXiv:2003.03033*（2020年）。[16] Business Research Insights。 [无日期]。全球云台摄像机市场研究报告（2020年）。h t t p s:// w w w.businessresearchinsights.com/market-reports/ptz-cameras-market-100130。[17] Zhaowei Cai、Mohammad Saberian 和 Nuno Vasconcelos。2015年。为深度行人检测学习复杂度感知级联。载于*2015*年*IEEE*国际计算机视觉会议论文集（*ICCV' 15*）。IEEE计算机学会，美国华盛顿哥伦比亚特区，3361 3369 10 1109 2015 384 –。https://doi.org/。[18] / ICCV.。Christopher Canel、Thomas Kim、Giulio Zhou、Conglong Li、Hyeon-taek Lim、David G. Andersen、Michael Kaminsky 和 Subramanya R. Dulloor。2019年。在受限边缘节点上扩展视频分析。载于第*2*届*SysML*会议。[19] Frank Cangialosi、Neil Agarwal、Venkat Arun、Junchen Jiang、Srinivas Narayana、Anand Sarwate 和 Ravi Netravali。2022年。Privid：实用的隐私保护视频分析查询。载于第*19*届*USENIX*网络系统设计与实现会议论文集（美国华盛顿州伦顿）（*NSDI'22*）。USENIX协会，美国加利福尼亚州伯克利。[20] David Cassel。[无日期]。我们准备好使用人工智能驱动的安全摄像头了吗？https : / / thenewstack . io / are-we-ready-for-ai-powered-security-cameras/。[21] Tiffany Yu-Han Chen、Lenin Ravindranath、Shuo Deng、Paramvir Bahl 和 Hari Balakrishnan。2015年。Glimpse：移动设备上的连续实时目标识别。载于第*13*届*ACM*嵌入式网络传感器系统会议论文集。155–168。[22] Mary Collins。 [无日期]。Hudl算法：将视频转化为球员追踪数据。https://www.maryecollins.com/hudl-tracking。[23] Matthieu Courbariaux、Itay Hubara、Daniel Soudry、Ran El-Yaniv 和 Yoshua Bengio。2016年。二值化神经网络：使用权重与激活值受限为+ 1 或-1的深度神经网络进行训练。*arXiv*预印本 *arXiv:1602.02830*（2016年）。[24] Daniel Crankshaw、Xin Wang、Guilio Zhou、Michael J. Franklin、Joseph E. Gonzalez 和 Ion Stoica。2017年。Clipper：低延迟在线预测服务系统。载于第*14*届*USENIX*网络系统设计与实现研讨会（*NSDI 17*）。USENIX协会，美国马萨诸塞州波士顿，613–627。h t t p s:// w w w.usenix.org/conference/nsdi17/technical-sessions/presentation/crankshaw [25] Navneet Dalal 和 Bill Triggs。2005年。用于人体检测的方向梯度直方图。载于*2005*年*IEEE*计算机学会计算机视觉与模式识别会议（*CVPR'05*），第1卷。IEEE，886–893。[26] Sokemi Rene Emmanuel Datondji、Yohan Dupuis、Peggy Subirats 和 Pascal Vasseur。2016年。基于视觉的道路交通监控综述

道路交叉口。*IEEE*智能交通系统汇刊17卷10期（2016年10月），2681–2698。https://doi.org/10.1109/TITS.2016.2530146[27] Utsav Drolia、Katherine Guo、Jiaqi Tan、Rajeev Gandhi 和 Priya Narasimhan。2017年。Cachier：面向识别应用程序的边缘缓存。载于*2017*年*IEEE*第*37*届国际分布式计算系统会议（*ICDCS*）。276–286。https://doi.org/10.1109/ICDCS.2017.94[28] Kuntai Du、Ahsan Pervaiz、Xin Yuan、Aakanksha Chowdhery、Qizheng Zhang、Henry Hoffmann 和 Junchen Jiang。2020年。用于深度学习推理的服务器驱动视频流传输。载于*ACM*数据通信特别兴趣小组年会暨计算机通信的应用、技术、架构与协议会议论文集（线上活动，美国）（*SIG-COMM '20*）。计算机协会，美国纽约州纽约市，557–570。https://doi.org/10.1145/3387514.3405887[29] Kuntai Du、Qizheng Zhang、Anton Arapin、Haodong Wang、Zhengxu Xia 和 Junchen Jiang。2022年。AccMPEG：优化视频编码以实现准确的视频分析。载于机器学习与系统论文集，D. Marculescu、Y. Chi 和 C. Wu（编），第4卷。450–466。https://proceedings.mlsys.org/paper/2022/file/98f13708210194c475687be6106a3b84-Paper.pdf[30] Mengnan Du、Fan Yang、Na Zou 和 Xia Hu。2020年。深度学习公平性：计算视角。*IEEE*智能系统36卷4期（2020年），25–34。[31] Jinlong E、Lin He、Zhenhua Li 和 Yunhao Liu。2023年。WiseCam：通过巧妙调节无线云台摄像机，实现高性价比的运动目标跟踪。载于*IEEE INFOCOM 2023——IEEE*计算机通信会议。IEEE。[32] John Emmons、Sadjad Fouladi、Ganesh Ananthanarayanan、Shivaram Venkataraman、Silvio Savarese 和 Keith Winstein。2019年。揭开深度神经网络黑箱：跨越摄像机—云端边界的深度神经网络视频分析。载于*2019*年视频分析与智能边缘热点专题研讨会论文集（墨西哥洛斯卡沃斯）（*HotEdgeVideo'19*）。计算机协会，美国纽约州纽约市，27–32。https://doi.org/10.1145/3349614.3356023[33] Mark Everingham、Luc Gool、Christopher K. Williams、John Winn 和 Andrew Zisserman。2010年。Pascal视觉对象类别（VOC）挑战赛。国际计算机视觉期刊88卷2期（2010年6月），303–338。https://doi.org/10.1007/s11263-009-0275-4[34] Sadjad Fouladi、John Emmons、Emre Orbay、Catherine Wu、Riad S. Wahby 和 Keith Winstein。2018年。Salsify：通过视频编解码器与传输协议的紧密集成实现低延迟网络视频。载于第*15*届*USENIX*网络系统设计与实现会议论文集（美国华盛顿州伦顿）（*NSDI'18*）。USENIX协会，美国，267–282。[35] Isha Ghodgaonkar、Subhankar Chakraborty、Vishnu Banna、Shane Allcroft、Mohammed Metwaly、Fischer Bordwell、Kohsuke Kimura、Xinxin Zhao、Abhinav Goel、Caleb Tung 等。2020年。通过大规模计算机视觉分析全球社交距离。*arXiv*预印本 *arXiv:2008.12363*（2020年）。[36] Grand View Research。[无日期]。全球体育分析市场规模报告，2021—2028年。https://www.grandviewresearch.com/industry-analysis/sports-analytics-market。[37] Peizhen Guo、Bo Hu、Rui Li 和 Wenjun Hu。2018年。FoggyCache：跨设备近似计算复用。载于第*24*届年度移动计算与网络国际会议论文集（印度新德里）（*MobiCom '18*）。计算机协会，美国纽约州纽约市，19–34。https://doi.org/10.1145/3241539.3241557[38] Peizhen Guo 和 Wenjun Hu。2018。Potluck：面向计算密集型移动应用程序的跨应用近似去重。*SIGPLAN* 通讯53卷2期（2018年3月），271–284。https://doi.org/10.1145/3296957.3173185[39] Seungyeop Han、Haichen Shen、Matthai Philipose、Sharad Agarwal、Alec Wolman 和 Arvind Krishnamurthy。2016年。MCDNN：一种用于资源约束下深度流处理的基于近似的执行框架。载于第*14*届年度移动系统、应用与服务国际会议论文集（新加坡，新加坡）（*MobiSys '16*）。计算机协会，美国纽约州纽约市，123–136。https://doi.org/10.1145/2906388.2906396[40] Kaiming He、Georgia Gkioxari、Piotr Dollár 和 Ross B. Girshick。2017年。Mask R-CNN。*CoRR* abs/1703.06870（2017年）。arXiv:1703.06870[41] http://arxiv.org/abs/1703.06870。Songtao He、Favyen Bastani、Arjun Balasingam、Karthik Gopalakrishna、Ziwen Jiang、Mohammad Alizadeh、Hari Balakrishnan、Michael Cafarella、Tim Kraska 和 Sam Madden。2020年。BeeCluster：通过预测优化实现无人机编排。载于第*18*届移动系统、应用与服务国际会议论文集（加拿大安大略省多伦多）（*MobiSys '20*）。计算机协会，美国纽约州纽约市，299–311。https://doi.org/10.1145/3386901.3388912[42] Richard M Karp。1970年。旅行商问题与最小生成树。《运筹学》18卷，6期（1970年），1138–1162页。[43] Geoffrey Hinton、Oriol Vinyals 和 Jeff Dean。2015年。在神经网络中提炼知识。*arXiv*预印本 *arXiv:1503.02531*（2015年）。[44] Honey Optics。[n.d.]。云台摄像机的价格是多少？https://honeyoptics.com/how-much-do-ptz-cameras-cost/。[45] Jonathan Huang、Vivek Rathod、Chen Sun、Menglong Zhu、Anoop Korattikara、Alireza Fathi、Ian Fischer、Zbigniew Wojna、Yang Song、Sergio Guadarrama 等。2017年。现代卷积目标检测器的速度与准确率权衡。载于*IEEE*计算机视觉与模式识别会议论文集。7310–7311页。[46] HuddleCamHD。[n.d.]。了解电子云台变焦与云台俯仰变焦的区别。https://huddlecamhd.com/eptz-and-ptz/。[47] Benoit Jacob、Skirmantas Kligys、Bo Chen、Menglong Zhu、Matthew Tang、Andrew Howard、Hartwig Adam 和 Dmitry Kalenichenko。2018年。面向高效仅整数运算推理的神经网络量化与训练。载于*IEEE*计算机视觉与模式识别会议论文集。2704–2713页。[48] Samvit Jain、Xun Zhang、Yuhao Zhou、Ganesh Ananthanarayanan、Junchen Jiang、Yuanchao Shu、Victor Bahl 和 Joseph Gonzalez。2020年。Spatula：大型摄像头网络上的高效跨摄像头视频分析。载于*ACM/IEEE*边缘计算研讨会（*SEC 2020*）。[49] Sagar Jha、Youjie Li、Shadi Noghabi、Vaishnavi Ranganathan、Peeyush Kumar、Andrew Nelson、Michael Toelle、Sudipta Sinha、Ranveer Chandra 和 Anirudh Badam。2021年。Visage：为无人机影像实现及时分析。载于第*27*届年度移动计算与网络国际会议论文集（路易斯安那州新奥尔良）（*MobiCom '21*）。计算机协会，美国纽约州纽约市，789–803页。https://doi.org/10.1145/3447993.3483273[50] Angela H. Jiang、Daniel L.-K. Wong、Christopher Canel、Lilia Tang、Ishan Misra、Michael Kaminsky、Michael A. Kozuch、Padmanabhan Pillai、David G. Andersen 和 Gregory R. Ganger。2018年。Mainstream：面向多租户视频处理的动态主干共享。载于*2018*年*USENIX*年度技术会议（*USENIX ATC 18*）。USENIX协会，美国马萨诸塞州波士顿，29–42页。https://www.usenix.org/conference/atc18/presentation/jiang[51] Junchen Jiang、Ganesh Ananthanarayanan、Peter Bodik、Siddhartha Sen 和 Ion Stoica。2018年。Chameleon：视频分析的可扩展适配。载于2018年计算机

*ACM*数据通信特别兴趣小组（匈牙利布达佩斯）（*SIGCOMM '18*）。ACM，美国纽约州纽约市，253–266页。https://doi.org/10.1145/3230543.3230574[52] Junchen Jiang、Ganesh Ananthanarayanan、Peter Bodik、Siddhartha Sen 和 Ion Stoica。2018年。Chameleon：视频分析的可扩展适配。载于*2018*年*ACM*数据通信特别兴趣小组会议论文集（匈牙利布达佩斯）（*SIG- COMM '18*）。计算机协会，美国纽约州纽约市，

253 266 10 1145 3230543 3230574 – 。

[53] 2019 https://doi.org/ 。/ 。Daniel Kang、Peter Bailis 和 Matei Zaharia。 。BlazeIt：面向神经网络视频分析的声明式聚合与限制查询优化。《*VLDB*基金会会刊》13卷，4期（2019年12月），533–546页。https://doi.org/10.14778/3372716.3372725[54] Daniel Kang、John Emmons、Firas Abuzaid、Peter Bailis 和 Matei Zaharia。2017年。NoScope：大规模优化视频上的神经网络查询。《*VLDB*基金会会刊》10卷，11期（2017年8月），1586–1597页。https://doi.org/10.14778/3137628.3137664[55] Ronald Kemker、Marc McClure、Angelina Abitino、Tyler Hayes 和 Christopher Kanan。2018年。测量神经网络中的灾难性遗忘。*AAAI*人工智能会议论文集32卷，1期（2018年4月）。https://doi.org/10.1609/aaai.v32i1.11651[56] Aditya Khosla、Tinghui Zhou、Tomasz Malisiewicz、Alexei A Efros 和 Antonio Torralba。2012年。消除数据集偏差造成的损害。载于《计算机视觉——*ECCV 2012*：第*12*届欧洲计算机视觉会议，意大利佛罗伦萨，*2012*年*10*月*7*日至*13*日，论文集，第一部分 *12*》。施普林格，158–171页。[57] Judy Goldsmith Kshitija Taywade、Brent Harrison。2022年。利用非平稳多臂老虎机，在具有非平稳需求的重复古诺博弈中进行学习。*arXiv*预印本 *arXiv:2201.00486* （2022 年）。[58] Adarsh Kumar 、 Arjun Balasubramanian、Shivaram Venkataraman 和 Aditya Akella。2019年。通过冻结加速深度学习推理。载于第*11*届*USENIX*云计算热点研讨会（*HotCloud 19*）。USENIX协会，华盛顿州伦顿。https: // www.usenix.org/conference/hotcloud19/presentation/kumar [59] H. Li、Z. Lin、X. Shen、J. Brandt 和 G. Hua。2015年。用于人脸检测的卷积神经网络级联。载于*2015*年*IEEE*计算机视觉与模式识别会议（*CVPR*）。5325–5334页。[60] Yuanqi Li、Arthi Padmanabhan、Pengzhan Zhao、Yufei Wang、Guo- qing Harry Xu 和 Ravi Netravali。2020年。Reducto：摄像头端过滤，实现资源高效的实时视频分析。载于*ACM*数据通信特别兴趣小组年会论文集：计算机通信的应用、技术、架构与协议（线上活动，美国）（*SIG- COMM '20*）。计算机协会，美国纽约州纽约市，359–376页。https://doi.org/10.1145/3387514.3405874[61] Zhuqi Li、Yuanchao Shu、Ganesh Ananthanarayanan、Longfei Shangguan、Kyle Jamieson 和 Victor Bahl。2021年。Spider：用于实时视频分析的多跳毫米波网络。载于*ACM/IEEE*边缘计算研讨会。ACM/IEEE。https://www.microsoft.com/en-us/research/publication/spider-a-multi-hop-millimeter-wave-network-for-live-video-analytics/ [62] Michael Lin、Tsung-Yiand Maire、Serge Belongie、James Hays、Pietro Perona、Deva Ramanan、Piotr Dollár 和 C. Lawrence Zitnick。2014年。Microsoft COCO：上下文中的常见物体。载于计算机视觉——*ECCV 2014*。740–755页。[63] T. Lin、P. Dollár、R. Girshick、K. He、B. Hariharan 和 S. Belongie。2017年。用于目标检测的特征金字塔网络。载于*2017*年*IEEE*计算机视觉与模式识别会议（*CVPR*）。936–944页。https://doi.org/10.1109/CVPR.2017.106[64] Wei Liu、Dragomir Anguelov、Dumitru Erhan、Christian Szegedy、Scott Reed、Cheng-Yang Fu 和 Alexander C. Berg。2016年。SSD：单次多框检测器。收录于计算机视觉——*ECCV 2016*。施普林格国际出版，21–37。[65] Zhuang Liu、Jianguo Li、Zhiqiang Shen、Gao Huang、Shoumeng Yan 和 Changshui Zhang。2017年。通过网络瘦身学习高效卷积网络。收录于*IEEE*国际计算机视觉会议论文集。2736–2744。[66] David G Lowe。2004年。基于尺度不变关键点的独特图像特征。《国际计算机视觉期刊》60（2004年）。[67] Yao Lu、Aakanksha Chowdhery 和 Srikanth Kandula。2016年。Optasia：用于高效大规模视频分析的关系型平台。收录于第七届 *ACM* 云计算研讨会论文集（美国加利福尼亚州圣克拉拉）（*SoCC '16*）。ACM，美国纽约州纽约市，57–70。https://doi.org/10.1145/2987550.2987564[68] Mehrdad Khani、Ganesh Ananthanarayanan、Kevin Hsieh、Junchen Jiang、Ravi Netravali、Yuanchao Shu、Mohammad Alizadeh、Victor Bahl。2023年。RECL：面向视频分析的响应式资源高效持续学习。收录于第*20*届 *USENIX* 网络系统设计与实现研讨会（*NSDI 23*）。USENIX协会，马萨诸塞州波士顿。[69] 微软 Azure。2021年。Azure Stack Edge。https://azure.microsoft. com/en-us/services/databox/edge/ [70] R. Netravali、A. Sivaraman、K. Winstein、S. Das、A. Goyal、J. Mickens 和 H. Balakrishnan。2015年。Mahimahi：针对 HTTP 的准确记录与回放（*ATC '15* 论文集）。USENIX。[71] 英伟达。2021年。NVIDIA JetsonNano。https://www.nvidia. com/en-us/autonomous-machines/embedded-systems/jetson-nano/product-development/ [72] Kemal Oksuz、Baris Can Cam、Sinan Kalkan 和 Emre Akbas。2020年。目标检测中的类别不平衡问题：综述。《*IEEE*模式分析与机器智能汇刊》43，10（2020年）。[73] Omar Besbes、Yonatan Gur、Assaf Zeevi。2014年。具有非平稳奖励的随机多臂老虎机问题。收录于*NeurIPS*。[74] Arthi Padmanabhan、Neil Agarwal、Anand Iyer、Ganesh Ananthanarayanan、Yuanchao Shu、Nikolaos Karianakis、Guoqing Harry Xu 和 Ravi Netravali。2022年。GEMEL：在边缘端进行内存高效、实时视频分析的模型合并。https://doi.org/10. 48550/ARXIV.2201.07705[75] Xunyu Pan 和 Siwei Lyu。2010年。使用 SIFT特征检测图像区域重复。收录于*2010 IEEE*国际声学、语音与信号处理会议。IEEE，1706–1709。[76] Sibendu Paul、Utsav Drolia、Y. Charlie Hu 和 Srimat T. Chakradhar。2021年。AQuA：用于优化视频分析系统的分析式质量评估。https://doi.org/10.48550/ARXIV.2101.09752[77] Sibendu Paul、Kunal Rao、Giuseppe Coviello、Murugan Sankaradas、Oliver Po、Y. Charlie Hu 和 Srimat T. Chakradhar。2023年。通过实时自动摄像头参数调优提升视频分析准确率。收录于第*20*届 *ACM* 嵌入式联网传感器系统会议论文集（马萨诸塞州波士顿）（*SenSys '22*）。计算机协会，纽约州纽约市，291–304。https://doi.org/ 10 1145 3560905 3568527 . / .

[78] [ ] PTZ Optics。无日期。PTZ Optics 云台摄像机。https://ptzoptics.com/products/. [79] PTZ Optics。 [无日期]。什么是电子云台变焦，它与真正的云台俯仰变焦有何区别？https://ptzoptics.com/what-is-eptz/。[80] Joseph Redmon 和 Ali Farhadi。2018年。YOLOv3：渐进式改进。*arXiv*预印本 *arXiv:1804.02767* （2018年）。[81] Richard Zhang、Phillip Isola、Alexei A. Efros、Eli Shechtman、Oliver Wang。2018年。深度特征作为感知度量的非凡有效性。收录于*CVPR*。[82] Mohammed Rijas。 [无日期]。在物联网世界中以人工智能驱动边缘端。https://www.forbes.com/sites/forbestechcouncil/2020/04/06/powering-the-edge-with-ai-in-an-iot-world/。

[83] Alberto Rizzoli。2022年。体育产业中的 7 项改变游戏规则的人工智能应用。https://www.v7labs.com/blog/ai-in-sports。[84] Francisco Romero、Qian Li、Neeraja J. Yadwadkar 和 Christos Kozyrakis。2021年。INFaaS：自动化无模型推理服务。收录于*2021*年 *USENIX*年度技术会议（*USENIX ATC 21*）。USENIX协会，397–411。https://www.usenix.org/conference/atc21/presentation/romero [85] SCW。 [无日期]。PTZ自动跟踪详解。https://www.getscw. com/knowledge-base/auto-tracking-explained。[86] SecurityBros。 [无日期]。哪款廉价室外WiFi PTZ IP摄像机最好？Boavision对比 Inqmega。https://securitybros.com/which-cheap-outdoor-wifi-ptz-ip-camera-is-best-boavision-vs-inqmeg

a/.

[87] Shaoqing Ren、Kaiming He、Ross Girshick、Jian Sun。2015年。Faster R-CNN：借助区域提议网络实现实时目标检测。发表于*NeurIPS*。[88] Haichen Shen、Lequn Chen、Yuchen Jin、Liangyu Zhao、Bingyu Kong、Matthai Philipose、Arvind Krishnamurthy 和 Ravi Sundaram。2019年。Nexus：用于加速基于深度神经网络的视频分析的 GPU 集群引擎。发表于第*27*届 *ACM*操作系统原理研讨会论文集 （加拿大 Ontario 的 Huntsville）（*SOSP '19*）。计算机协会，纽约州纽约市，美国，322–337。https://doi.org/ 10 1145 3341301 3359658 . / 。[89] Weisong Shi、Jie Cao、Quan Zhang、Youhuizi Li 和 Lanyu Xu。2016年。边缘计算：愿景与挑战。*IEEE*物联网期刊 3, 5（2016年），637–646。[90] Shubham Jain、Viet Nguyen、Marco Gruteser、Paramvir Bahl。2017年。Panoptes：使用可转向摄像机同时服务多个应用程序。发表于*IPSN*。[91] Vishwanath A Sindagi 和 Vishal M Patel。2017年。使用上下文金字塔卷积神经网络生成人群高质量密度图。发表于*IEEE*国际计算机视觉会议论文集。1861–1870。[92] Ryan Steed 和 Aylin Caliskan。2021年。通过无监督预训练学习到的图像表征包含类人偏见。发表于*2021*年 *ACM* 公平、问责与透明度会议论文集。701–713。[93] Abhijit Suprem、Joy Arulraj、Calton Pu 和 Joao Ferreira。2020年。ODIN：视频分析中的自动漂移检测与恢复。*VLDB*基金会学报 13, 12（2020年7月），2453–2465。https://doi.org/ 10.14778/3407790.3407837[94] Mingxing Tan、Ruoming Pang 和 Quoc V. Le。2019年。EfficientDet：可扩展且高效的目标检测。*CoRR*abs/1911.09070（2019年）。arXiv: 1911 09070 1911 09070 . http://arxiv.org/abs/ 。[95] Amos Tversky 和 Itamar Gati。1982. 相似性、可分离性与三角不等式。心理学评论 89, 2（1982年），123。[96] Deepak Vasisht、Zerina Kapetanovic、Jong-ho Won、Xinxin Jin、Ran-veer Chandra、Ashish Kapoor、Sudipta N. Sinha、Madhusudhan Su-darshan 和 Sean Stratman。2017年。Farmbeats：用于数据驱动农业的物联网平台。发表于第*14*届 *USENIX* 网络系统设计与实现会议论文集 （美国马萨诸塞州波士顿）（*NSDI'17*）。USENIX协会，美国，515–528。[97] Lluís Martínez；Margarita Cabrera-Bean；Josep Vidal。2021年。用于非平稳无线网络选择的多臂老虎机模型。发表于*2021*年 *IEEE Globecom Workshops*。IEEE。[98] Vidit Saxena、Joakim Jaldén、Joseph E. Gonzalez、Mats Bengtsson、Hugo Tullberg、Ion Stoica。2019年。用于蜂窝网络链路自适应的上下文多臂老虎机。发表于*NetAI*。[99] Chien-Yao Wang、Alexey Bochkovskiy 和 Hong-Yuan Mark Liao。2021年。Scaled-YOLOv4：跨阶段部分网络的尺度扩展。arXiv:2011.08036 [cs.CV]

[100] Junjue Wang、Ziqiang Feng、Shilpa George、Roger Iyengar、Padman- abhan Pillai 和 Mahadev Satyanarayanan。2019年。《迈向可扩展的边缘原生应用》。载于第*4*届 *ACM/IEEE* 边缘计算研讨会论文集（弗吉尼亚州阿灵顿）（*SEC '19*）。计算机协会，纽约，美国纽约州纽约市，152–165。https://doi.org/10.1145/3318216.3363308[101] Yiding Wang、Weiyan Wang、Junxue Zhang、Junchen Jiang 和 Kai Chen。2019年。《突破边缘云壁垒，实现实时高级视觉分析》。载于第*11*届 *USENIX* 云计算热点研讨会（*HotCloud 19*）。USENIX协会，伦顿，华盛顿州。https: // www.usenix.org/conference/hotcloud19/presentation/wang[102] Zeyu Wang、Klint Qinami、Ioannis Christos Karakozis、Kyle Genova、Prem Nair、Kenji Hata 和 Olga Russakovsky。2020年。《迈向视觉识别公平性：偏差缓解的有效策略》。载于*IEEE/CVF*计算机视觉与模式识别会议论文集。8919–8928。[103] Mengwei Xu、Mengze Zhu、Yunxin Liu、Felix Xiaozhu Lin 和 Xu- anzhe Liu。2018年。《DeepCache：面向移动深度视觉的原理化缓存》。载于第*24*届年度移动计算与网络国际会议论文集（新德里，印度）（*MobiCom '18*）。计算机协会，纽约，美国纽约州纽约市，129 144 10 1145 3241539 3241563 – 。https://doi.org/ . / .[104] Yifan Yang 、 Guorong Li 、 Dawei Du 、 Qingming Huang 和 Nicu Sebe。2020年。《将透视分析融入用于人群计数的多列卷积神经网络》。*IEEE*图像处理汇刊30（2020），1395–1407。[105] Yifu Zhang、Peize Sun、Yi Jiang、Dongdong Yu、Zehuan Yuan、Ping Luo、Wenyu Liu、Xinggang Wang。2021年。《ByteTrack：通过关联每个检测框实现多目标跟踪》。arXiv:2110.06864 [cs.CV][106] Xiaoqi Yin、Abhishek Jindal、Vyas Sekar 和 Bruno Sinopoli。2015年。《一种基于控制理论的方法：通过 HTTP 实现动态自适应视频流》。*SIGCOMM Comput. Commun. Rev.* 45，4（2015年8月），325 338 10 1145 2829988 2787486 – 。https://doi.org/ . / .[107] Jason Yosinski、Jeff Clune、Yoshua Bengio 和 Hod Lipson。2014年。《深度神经网络中的特征可迁移性有多强？》。载于神经信息处理系统进展，Z. Ghahramani、M. Welling、C. Cortes、N. Lawrence 和 K.Q. Weinberger（编），第27卷。Curran Associates, Inc. https://proceedings.neurips.cc/paper/2014/file/375c71349b295fbe2dcdca9206f20a06-Paper.pdf[108] Matthew D Zeiler 和 Rob Fergus。2014年。《卷积网络的可视化与理解》。载于计算机视觉—*ECCV 2014*：第*13*届欧洲会议，苏黎世，瑞士，*2014*年*9*月*6*日至*12*日，论文集，第一部分 *13*。施普林格，818–833。[109] Anran Zhang、Lei Yue、Jiayi Shen、Fan Zhu、Xiantong Zhen、Xian- bin Cao 和 Ling Shao。2019年。《用于人群计数的注意力神经场》。载于*IEEE/CVF*国际计算机视觉会议论文集。5714–5723。[110] Haoyu Zhang、Ganesh Ananthanarayanan、Peter Bodik、Matthai Phili- pose、Paramvir Bahl 和 Michael J. Freedman。2017年。《利用近似和延迟容忍实现大规模实时视频分析》。载于第*14*届 *USENIX* 网络系统设计与实现会议论文集（美国马萨诸塞州波士顿）（*NSDI'17*）。USENIX协会，美国加利福尼亚州伯克利，377–392。http://dl.acm.org/citation. cfm?id=3154630.3154661[111] Haoyu Zhang、Ganesh Ananthanarayanan、Peter Bodik、Matthai Phili- pose、Paramvir Bahl 和 Michael J. Freedman。2017年。《利用近似和延迟容忍实现大规模实时视频分析》。载于第*14*届 *USENIX* 网络系统设计与实现研讨会（*NSDI 17*）。USENIX协会，波士顿，马萨诸塞州，377–392。https://www.usenix.org/conference/nsdi17/technical-sessions/presentation/zhang

[112] Tan Zhang、Aakanksha Chowdhery、Paramvir Bahl、Kyle Jamieson 和 Suman Banerjee。2015年。《无线视频监控系统的设计与实现》。426–438。https://doi.org/10. 1145/2789168.2790123[113] Kun Zhao、Bin Liu、Luchuan Song、Weihai Li 和 Nenghai Yu。2019年。《用于人群计数的级联残差密度网络》。载于*2019*年 *IEEE*国际图像处理会议（*ICIP*）。IEEE，2199– 2203。

[114] Chenzhuo Zhu、Song Han、Huizi Mao 和 William J Dally。2016年。训练式三值量化。*arXiv* 预印本 *arXiv:1612.01064*（2016年）。[115] Yuhao Zhu、Anand Samajdar、Matthew Mattina 和 Paul Whatmough。2018年。Euphrates：面向低功耗移动连续视觉的算法与片上系统协同设计。载于第*45*届国际计算机体系结构研讨会论文集（美国加利福尼亚州洛杉矶）（*ISCA '18*）。IEEE出版社，547–560。https://doi.org/10.1109/ISCA.2018.00052

A 附录

A.1 工作负载

模型	对象	Type
SSD	人	聚合计数
Faster RCNN	cars	二分类
SSD	人	计数
YOLOv4	人	检测
Faster RCNN	人	检测

表 3：工作负载 1（W1）

模型	对象	Type
YOLOv4	人	聚合计数
Tiny YOLOv4	人	聚合计数
Tiny YOLOv4	人	检测
YOLOv4	人	二分类
Tiny YOLOv4	人	聚合计数
Faster RCNN	人	计数
Faster RCNN	人	检测
Faster RCNN	car	计数
YOLOv4	人	聚合计数
YOLOv4	人	检测
YOLOv4	人	计数
Tiny YOLOv4	人	聚合计数
YOLOv4	car	计数
YOLOv4	car	检测
Tiny YOLOv4	car	计数
SSD	人	二分类
Faster RCNN	car	计数
SSD	car	计数

表 4：工作负载 2（W2）

模型	对象	Type
SSD	car	二分类
Faster RCNN	人	聚合计数
Faster RCNN	人	计数
Tiny YOLOv4	人	二分类
Tiny YOLOv4	人	二分类
Tiny YOLOv4	人	聚合计数
YOLOv4	人	计数
Faster RCNN	人	聚合计数
SSD	人	二分类
Faster RCNN	car	计数
SSD	car	计数

表 5：工作负载 3（W3）

模型	对象类型	
Tiny YOLOv4	car	计数
Faster RCNN	car	检测
Faster RCNN	人	聚合计数

表 6：工作负载 4 (W4)

模型	对象	Type
Tiny YOLOv4	car	计数
SSD	car	计数
Faster RCNN	人	聚合计数

表 7：工作负载 5（W5）

模型	对象	Type
Tiny YOLOv4	人	聚合计数
Tiny YOLOv4	人	二分类
SSD	car	计数
YOLOv4	人	聚合计数
Tiny YOLOv4	人	计数
Faster RCNN	car	二分类
SSD	人	检测
Faster RCNN	car	检测
Faster RCNN	人	聚合计数
YOLOv4	car	计数
Tiny YOLOv4	人	聚合计数
Faster RCNN	人	检测
SSD	人	聚合计数
YOLOv4	car	检测

表 8：工作负载 6（W6）

模型	对象	Type
YOLOv4	人	二分类
SSD	人	检测
Tiny YOLOv4	car	二分类
Tiny YOLOv4	人	检测
SSD	人	二分类
SSD	人	聚合计数
Tiny YOLOv4	人	检测
SSD	car	计数
SSD	人	计数
Faster RCNN	人	计数
YOLOv4	人	计数
Faster RCNN	人	二分类
Tiny YOLOv4	人	聚合计数
Faster RCNN	人	聚合计数
Faster RCNN	car	计数
YOLOv4	car	二分类

表 9：工作负载 7（W7）

模型	对象	Type
Faster RCNN	car	计数
Tiny YOLOv4	人	二分类
YOLOv4	人	聚合计数
YOLOv4	car	计数
Tiny YOLOv4	人	聚合计数
Faster RCNN	人	聚合计数
YOLOv4	人	聚合计数
Faster RCNN	car	计数
SSD	car	计数
Faster RCNN	car	计数
SSD	car	二分类
YOLOv4	car	二分类
SSD	car	二分类
SSD	人	计数
YOLOv4	人	计数
YOLOv4	car	二分类
Faster RCNN	人	聚合计数
SSD	car	检测

表10：工作负载 8（W8）

模型	对象	Type
Tiny YOLOv4	人	聚合计数
Faster RCNN	人	计数
Faster RCNN	人	计数
Tiny YOLOv4	car	检测
Tiny YOLOv4	人	二元分类
YOLOv4	人	检测
Faster RCNN	人	计数
YOLOv4	人	聚合计数
SSD	人	聚合计数

表11：工作负载9（W9）

模型	对象	Type
Faster RCNN	人	聚合计数
Faster RCNN	car	计数
Faster RCNN	人	计数

表12：工作负载10（W10）