串行胜于并行：学习多模态视觉对象跟踪与基准测试的持续统一

Zhangyong Tang
zhangyong_tang_jnu@163.com
Jiangnan University
Wuxi, China

Tianyang Xu
tianyang.xu@jiangnan.edu.cn
Jiangnan University
Wuxi, China

Xue-Feng Zhu
xuefeng_zhu95@163.com
Jiangnan University
Wuxi, China

Chunyang Cheng
chunyang_cheng@163.com
Jiangnan University
Wuxi, China

Tao Zhou
taozhou.ai@gmail.com
Nanjing University of Science and Technology
Nanjing, China

Xiaojun Wu*
wu_xiaojun@jiangnan.edu.cn
Jiangnan University
Wuxi, China

Josef Kittler
j.kittler@surrey.ac.uk
University of Surrey
Guildford, United Kingdom

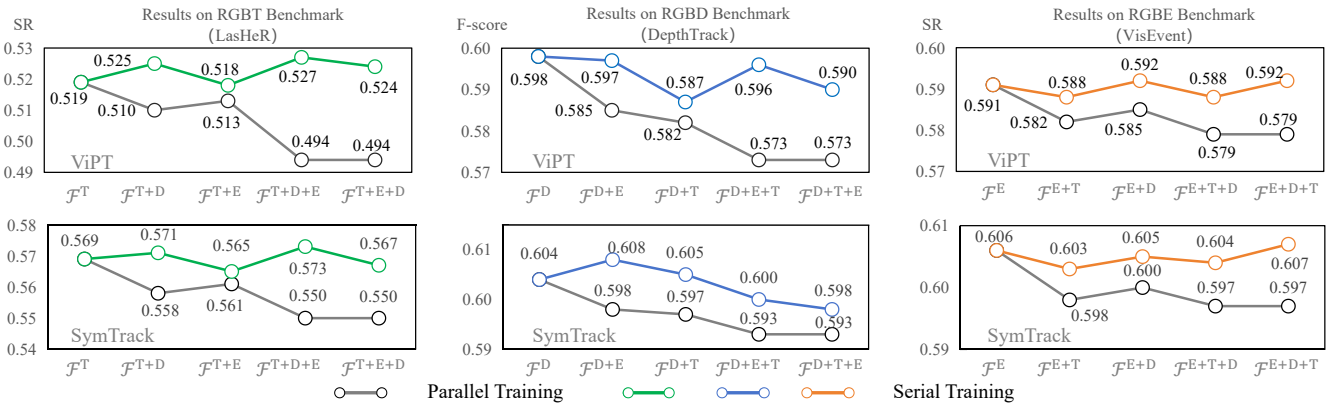

Figure 1: 比较串行训练和并行训练的统一追踪器。$\mathcal{F}^{\mathrm{T}}$、$\mathcal{F}^{\mathrm{D}}$ 和 $\mathcal{F}^{\mathrm{E}}$ 分别代表具有 RGBT（可见光（RGB）+ 热红外（T））、RGBD（RGB+ 深度（D））和 RGBE（RGB+ 事件（E））数据的任务。$\mathcal{F}^{\mathrm{T+D}}$ 表示一种情境，在这种情境下，RGBT 数据是之前可用的，随后引入了 RGBD 数据。在这种情境中，RGBT 和 RGBD 数据在并行训练中混合无序，而在串行训练中则是顺序引入的。

Abstract

由于不同模态在构建鲁棒跟踪系统中的互补特性，将多种多模态视觉目标跟踪（MMVOT）任务统一起来引起了越来越多的关注。现有的方法在单一的训练过程中融合所有的数据传感器类型，从数据中心的视角结构一个并行的范式，并旨在对所涉及任务的联合分布达到全局最优。然而，由于缺乏一个所有类型数据共存的统一基准，导致在分离的基准上进行评估，从而造成训练与测试之间的不一致，进而导致性能下降。为了解决这些问题，这项工作在两个方面进行了改进：❶ 引入一个名为 UniBench300 的统一基准，通过整合多任务数据来弥合不一致性，减少推理过程的次数从三次降到一次，并减少了 27 % 的时间消耗。❷ 统一过程被重新表述为一种串行格式，逐步整合新任务。通过这种方式，性能的下降可以被明确为先前任务的知识遗忘，这自然与持续学习（CL）的理念相符，进一步激励了将 CL 注入到统一过程中的探索。在两个基线和四个基准上进行的大量实验显示了 UniBench300 的重要性以及 CL 在支持稳定统一过程中的优越性。此外，在进行专门分析时，发现性能下降与网络容量呈负相关。此外，模态差异导致了跨任务的不同退化水平（在 MMVOT 中 RGBT > RGBD > RGBE），为未来的多模态视觉研究提供了有价值的见解。源码和所提出的基准可以在 https://github.com/Zhangyong-Tang/UniBench300 获取。

MM '25, Dublin, Ireland

ACM ISBN 979-8-4007-2035-2/2025/10
https://doi.org/10.1145/3746027.3754879

CCS Concepts

• **Computing methodologies** → **Tracking**.

Keywords

Multi-modal visual object tracking; unification; continual learning.

ACM Reference Format:
Zhangyong Tang, Tianyang Xu, Xue-Feng Zhu, Chunyang Cheng, Tao Zhou, Xiaojun Wu*, and Josef Kittler. 2025. 串行胜于并行：学习多模态视觉对象跟踪与基准测试的持续统一. In *Proceedings of the 33rd ACM International Conference on Multimedia (MM '25), October 27–31, 2025, Dublin, Ireland.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3746027.3754879

1 介绍

视觉对象跟踪专注于在视频序列中连续预测对象状态（位置和大小），通过利用各种数据模态的互补性正在迈入多模态时代。例如，与众所周知的 RGB 模态相比，T 模态增强了对环境变化的鲁棒性 [?]，D 模态提供了 3D 感知 [?]，而 E 模态捕捉到了细粒度的运动线索 [?]。目前有三种流行的 MMVOT 任务，RGBT、RGBD 和 RGBE 跟踪，以涉及的模态命名，并已被证明优于仅 RGB 跟踪 [???]。这一进展激发了统一多个 MMVOT 任务以结合其优势的兴趣，旨在在具有挑战的场景中提高鲁棒性。

如图 2 (a) 所示，现有的统一实践分为两类：具有统一架构的方法（Unified-A）[???????] 和具有统一模型的方法（Unified-M）[??]。由于 Unified-A 方法在不同任务中使用了相同的架构，它们仍然需要特定任务的适配，导致训练出多个独立的模型，而非一个整合了所有数据模态优势的统一模型。因此，进一步的讨论主要集中在第二类（Unified-M）。

动机：实现一个统一的模型从最基本的一步——数据准备开始。如图 2 (b) 所示，现有的具有统一模型的方法通常将所有类型的数据混合到一个统一的数据集合 [??] 中。在训练过程中，来自不同模态的数据并行加载，将多样的多模态知识注入到统一的模型中，以在联合分布上找到一个全局最优。然而，这与测试阶段相矛盾，在测试阶段，统一模型是在分开的基准上进行评估的，这与独立训练的方法相同 [????]。这反映出局部最优更受偏爱，而这种偏倚导致训练和测试之间的不一致，最终导致性能下降，如图 1 和表 2 所示。

为了解决这些问题，我们的工作做出了两个关键进展：❶ 如图 2 (d) 所示，为了解决不一致的问题，引入了 UniBench300 作为 MMVOT 的第一个统一基准。它包含了 300 个视频序列，包括 100 个 RGBT 序列，100 个 RGBD 序列，和 100 个 RGBE 序列，总共 368.1K 帧。通过形成多模态数据的联合分布，UniBench300 对训练和测试范式进行了对齐，从而弥合了不一致。此外，如表 4 所量化的，UniBench300 通过减少 27 % 的推理时间并且只需要进行一次评估（与之前的公式相比，无需进行三次单独评估），提供了便捷而高效的评估。❷ 从数据中心的角度来看，统一可以以并行或串行的方式进行。尽管现有的工作 [??] 采取并行统一，但它们存在性能下降的问题，因此需要探索串行统一。如图 2 (c) 所示，串行统一逐步整合新任务，将性能下降明确为先前任务的知识遗忘，这是 CL 领域的核心主题 [?]。在此基础上，以串行方式重新构建统一过程使得在 MMVOT 任务的统一中自然结合 CL 技术成为可能。如图 1 所示，两个基线和三个基准上的结果验证了 CL 在构建更稳定的统一过程中具有优越性。

备注：值得注意的是，我们专门分析中出现了两个有趣的现象：① 方法 1（ViPT）在所有任务中遭受比方法 2（SymTrack）更严重的性能下降；② 这两种方法在任务中的下降趋势相同（RGBT > RGBD > RGBE）。对于第一个观察，表 5 中的进一步调查表明，网络容量起着关键作用，其中较大的网络经历较少的性能下降。对于第二个，数据分布的质量分析在图 6 中以及跨任务的数量化验证在图 7 中揭示了相比于 T，D 和 E 模式在分布上更相似，且 D 与 T 之间的差异大于 E 与 T 之间的差异。这解释了为什么 RGBT 跟踪（结合两个较远的模式，D 和 E）承受了比 RGBD 和 RGBE 任务更大的负面影响（结合的单一相对较远的模式是 T，而 T 与 E 相对比 D 更接近）。

总之，我们工作的主要贡献包括：

为了弥合当前任务统一的训练和测试模式之间的不一致性，提出了首个统一的基准测试 UniBench300，它整合了 RGBT、RGBD 和 RGBE 数据。它提供了一个更加方便和高效的方法评估平台（在 UniBench300 上推理一次需要 93 分钟，而在 LasHeR、DepthTrack 和 VisEvent 上推理三次需要 127 分钟），显著减少了 27

从数据中心的角度出发，• 当前的并行统一过程被重新表述为一个串行过程，在此基础上可以自然地促进持续学习技术的整合。这通过减轻以往任务的知识遗忘显著提高了统一的稳定性。

通过研究揭示了统一化后性能下降的两个根本原因，并对未来研究提出潜在建议：① 性能下降与网络容量呈负相关。② 模态差异导致不同任务出现不同程度的性能下降。

在两个基线和四个基准上进行的大量实验证明了所提出基准的重要性以及 CL 在稳定 MMVOT 任务统一性方面的优越性。

在认识到多模态数据在增强不利场景下的跟踪性能的优势后，引入了各种基准以推进 MMVOT 研究，例如 RGBT234、LasHeR、RGBD1K、DepthTrack、VisEvent 和 COESOT。尽管这些基准在很大程度上加深了对 MMVOT 任务的研究，但它们都专注于单一类型的多模态数据。例如，LasHeR、DepthTrack 和 VisEvent 分别仅包含 RGBT、RGBD 和 RGBE 数据。然而，这一限制阻碍了统一跟踪器的发展，因为这些跟踪器需要一个涵盖多个 MMVOT 任务的基准。结果，现有的统一方法在混合数据上进行了训练，但在每个单独任务上分别进行评估，导致训练和测试之间的不一致。为了应对这一问题，我们引入了 UniBench300，这是第一个整合 RGBT、RGBD 和 RGBE 数据的基准，填补了这一空白，促进了进一步的统一跟踪研究。

1.1 统一多模态视觉目标跟踪

认识到实际场景的复杂性，当前的研究正在努力通过设计统一的跟踪器来结合多种 MMOVT 任务的优点。现有方法根据训练范式可分为两类：（1）Unified-A：这一类别的方法在不同任务上实现时保持网络架构相同，但参数重新训练 [????????]。因此，它们产生多个特定于任务的模型，而不是一个真正统一的模型。这限制了它们处理多样化挑战的能力，使它们与单独训练的方法面临相同的限制，如图 2 (a) 所示。（2）Unified-M：这些方法采用一个能够支持多项任务的单一模型 [??]，这使它们更适合部署。然而，它们直接混合所有数据类型，然后在训练过程中并行加载，而没有考虑不同模式之间的独特性。这导致与单独训练的模型相比，性能下降。为了解决这一问题，我们从数据角度重新审视统一过程，并提出一种序列统一范式，逐步整合新任务的知识，通过借鉴 CL 技术来缓解性能下降。

1.2 多模态跟踪中的 CL

CL 着重于来自不同任务 [??] 知识的稳定整合，并已成功应用于多个领域，例如多模态图像融合 [???]、大语言模型 [?] 和单模态视觉目标跟踪 [????]。然而，其在多模态跟踪中的应用，特别是在 MMVOT 任务统一方面，仍然未被充分探索。如

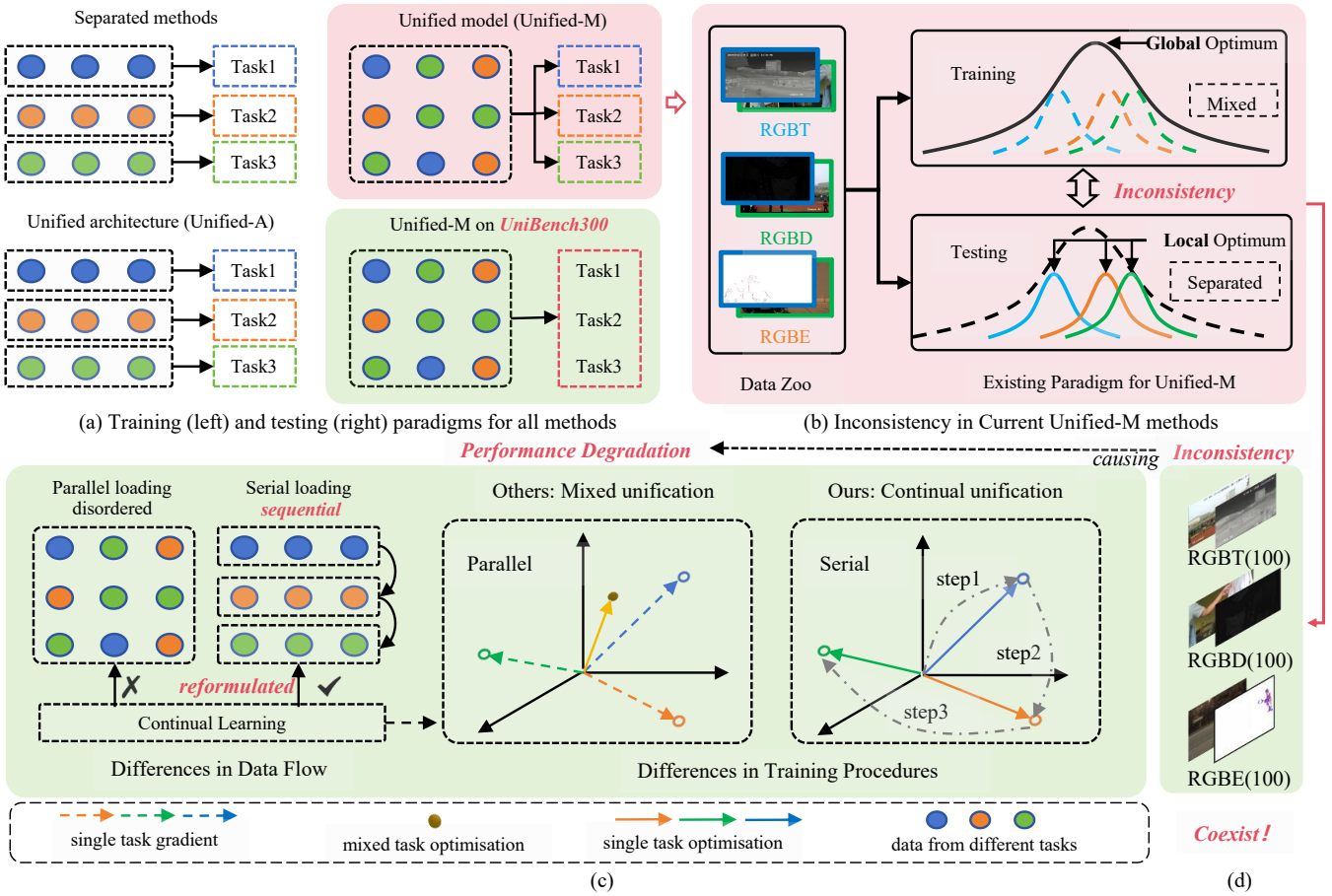

Figure 2: (a) 所有方法的训练（左）和测试（右）范例；(b) 在统一方法（Unified-M）中，当前训练和测试范例（全局对比局部）之间的不一致导致在分离的基准中性能下降。为了解决这些问题，(d) 提出了 UniBench300 作为第一个统一基准以弥补不一致，并且 (c) 统一过程被重新表述为一个串行过程，从而促进 CL 的注入以减轻性能下降。

图 2 (c) 所示，主要限制在于当前的训练模式，其中来自不同任务的多模态数据被无序且并行地使用，未能形成结构化的多任务方案，这是应用 CL 的前提。相比之下，我们的工作将统一过程重新表述为一个串行的过程，其中任务数据逐步出现。这一转变弥合了 MMVOT 统一与 CL 之间的差距，为进一步探索 CL 技术以增强构建统一模型的稳定性铺平了道路。

2 UniBench300

数据收集：针对当前统一跟踪器 [? ?] 的训练和测试范式不一致的问题，提出了 UniBench300 以同时支持多种跟踪任务来弥补这一差距。与现有基准不同，正如表 1 所示，UniBench300 是第一个包含 RGBT、RGBD 和 RGBE 序列的基准。它总共包括 300 个序列，包括来自 LasHeR 的 100 个 RGBT 序列 [?] 、来自 VisEvent 的 100 个 RGBE 序列 [?] 、来自 DepthTrack 的 50 个 RGBD 序列 [?] 和来自 RGBD1K 的 50 个 RGBD 序列 [?] 。重要的是，各任务间的序列数量是平衡的，以防止潜在的偏差。

关于数据收集的原则，我们坚持构建一个具有挑战性的基准，以更好地推动 MMVOT 社区的发展。有鉴于此，以 RGBT 序列为例，我们首先根据几种先进方法（包括 TBSI [?] 、GMMT [?] 和 BAT [?] 的每个序列的平均交并比（IoU）表现）对 LasHeR 中的所有序列进行排名。然后选择表现最差的 100 个序列作为 UniBench300 的 RGBT 分支。同样，RGBE 分支是从 VisEvent 中获得的，使用 TENet [?] 和 SeqTrackV2 [?] 作为代表性的

Table 1: 现有的 MMVOT 基准测试和 UniBench300。

Benchmark	Sequence Number	Total Frames	X Modality
GTOT [?]	50	7.8 K	T(100 %)
RGBT234 [?]	234	116.7 K	T(100 %)
LasHeR-test [?]	245	220.7 K	T(100 %)
VTUAV-ST-test [?]	176	631.4 K	T(100 %)
MV-RGBT [?]	124	89.9 K	T(100 %)
CDTB [?]	80	102.0 k	D(100 %)
DepthTrack-test [?]	50	76.4 K	D(100 %)
RGBD1K-test [?]	50	117.9 K	D(100 %)
ARKittrack-test [?]	50	64.3 K	D(100 %)
FE108-test [?]	32	59.7 K	E(100 %)
VisEvent-test [?]	320	106.8 K	E(100 %)
COESOT-test [?]	528	176.6 K	E(100 %)
Ours(UniBench300)	300	368.1K	T(33.3 %)/D(33.3 %)/E(33.3 %)

RGBE 跟踪器。最后，由于 DepthTrack 和 RGBD1K 的规模有限（它们的测试集均只包含 50 个序列），因此它们被合并以形成 RGBD 分支。总而言之，UniBench300 作为现有 RGBT、RGBD 和 RGBE 基准的一个具有挑战性的分支，其中同时包含所有模态。这样的独特组合给予 UniBench300 显著的优势，相较于其他基准，它在评估方面提供了更大的便利性和效率，详情如表 4 所示。

评估指标：UniBench300 采用精度率（PR）和成功率（SR）作为评估指标，与在 VisEvent [?] 和 LasHeR [?] 等已建立的基准中使用的指标保持一致。因此，关于 PR 和 SR 的详细介绍保留在补充材料中。

3 方法论

统一多种数据模态的优点是解决现实世界中高动态挑战的一个有前景的方法。然而，现有方法不加区分地混合所有数据，忽视了它们的独特特性，导致性能下降。在这项工作中，我们从数据混合分布的角度重新审视统一过程，并对现有的并行范式重新制定了一种串行解决方案。此转换使得性能下降可以进一步解释为先前任务的知识遗忘，这与 CL 的理念一致。因此，在两个基线上的实验验证了 CL 在促进更稳定的统一过程中的有效性。

3.1 统一多模态跟踪器

跟随广泛使用的范式 [?]，统一多模态追踪器的训练过程被表述为最小化预测和真实值边界框之间的距离：

$$\arg\min_{\boxtimes} \text{Loss}(f(\text{mix}(d_1, d_2, ..., d_n), \boxtimes), g) \quad (1)$$

其中 f 和 θ 分别是追踪器及其参数。d_n 表示来自 n^{th} 任务的训练数据，g 表示相关数据批的真实值边界框。$\text{mix}(d_1, d_2, ..., d_n)$ 指的是所有任务数据的混合无序组合，然后并行加载进行训练，旨在找到所有任务联合分布的全局最优。但是，训练后，模型在不同任务上进行 n 次评估，每次都期望在特定基准测试中达到局部最优。这揭示了训练和测试范式间的固有不一致性，最终导致统一方法相比那些为单个任务分别训练的方法性能下降。至于损失函数的详情，请参考 ViPT [?] 。

3.2 持续统一

训练和测试范式之间的不一致导致的性能下降促使我们深入研究统一过程。基本上，从数据中心的角度来看，统一可以通过两种方式实现：并行和串行。目前的实践 [? ?] 通过无序地混合和加载数据来遵循并行范式，这导致性能下降。因此，我们通过以串行方式结合来自多个任务的知识来重新制定统一过程。在这种范式下，性能下降可以进一步解释为以前任务的知识遗忘，这是 CL 中的一个基本挑战 [?] 。这自然引导我们探索 CL 技术以增强统一过程，该过程被称为持续统一过程，如图 2 (c) 所示。具体来说，我们采用重放，这是一种被广泛采用的 CL 技术 [?] ，其数学形式如下：

$$\arg\min_{\boxtimes_i} \text{Loss}(f(\text{mix}(d_1, ..., d_i), \boxtimes_{i-1}), g), i \in [1, ..., n] \quad (2)$$

其中 θ_{i-1} 和 θ_i 是分别从 $(i-1)^{th}$ 和 i^{th} 训练步骤中学习到的参数。在开始时，$i = 1$ ，θ_0 被随机初始化，仅使用任务 1 的训练数据（d_1），这与为任务 1 训练特定模型相同。当新的任务到来时，此前训练的模型 θ_{i-1} 被保留作为接下来训练步骤的初始化器，并且之前任务的训练数据也被保留。这确保了任务通过串行方式逐步实现统一。此外，通过整合 CL，最终模型

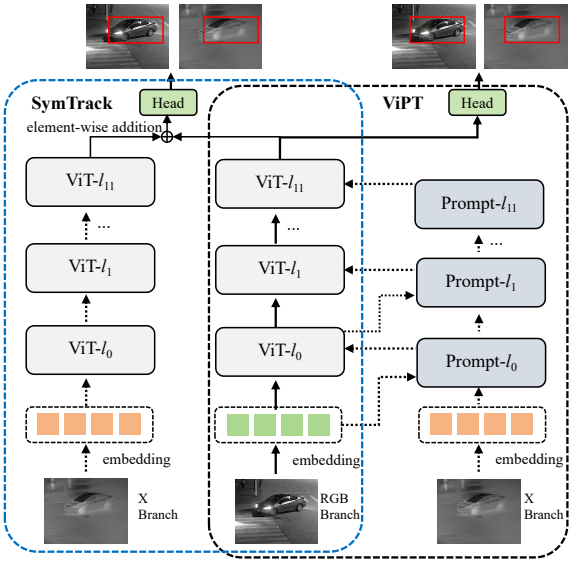

Figure 3: SymTrack 和 ViPT 的架构共享一个标准的 RGB 分支，因为这两种方法对 RGB 数据的处理方式相同。

$f(\theta_n)$ 保持了与分别训练的模型相媲美的性能，显著优于以并行方式训练的方法，如图 1 所示。

讨论：方程 2 表示在持续统一过程中，预训练模型和以前任务的数据都被重放。这背后的理由有两个：① 必须保留预训练模型；否则，当 $i = n$ 时，持续统一过程与原始并行范式相同，受到性能下降的影响；② 应该结合以前任务的数据；否则，模型将遭受灾难性遗忘，正如在持续学习的研究 [? ?] 中所强调的那样。

3.3 训练范式的差异

图 2 直观地展示了我们提出的持续统一方法与广泛使用的范式之间的区别。在现有范式中，所有类型的数据通常在每个批次中混合，导致在梯度计算后出现多种优化方向（混合统一）。随后，梯度仅根据批次大小 [?] 或任务特定的样本数量 [?] 进行简单平均，忽略了更深层次的任务差异，例如每个任务更新的适当幅度。结果是，最终的优化（黄色线）次优，导致统一不足。相比之下，提出的持续统一过程采用了多步训练策略。当引入新任务时，CL 技术被激活，使模型在学习新知识的同时更好地保持对先前学习任务的性能。总的来说，这两种范式在一个关键方面存在差异：从数据中心的角度来看，现有范式以并行方式加载训练数据，而提出的方法则以顺序方式处理数据。

图 2 (c) 展示了数据流的差异。显然，仅仅通过将统一过程重新构造成串行范式，它就与 CL [?] 领域中研究的典型场景（多任务）相一致。通过 CL 实现的显著改进，如图 1 所示，进一步验证了从数据中心角度重新审视统一过程的正确性。

如图 3 所示，所提出的多模态视觉目标追踪（MMVOT）任务的连续统一实施在两个基线上：ViPT [?] ，这是一种被许多统一追踪器采用的知名非对称方法；以及 SymTrack，一个由于其对称结构而命名的自设计基线。两种方法都遵循标准的“嵌入-骨干网络-头部” 流程，其中输入图像首先被转换为嵌入空间，由骨干网络处理，然后通过追踪头投射到任务空间。虽然 ViPT 和 SymTrack 在处理 RGB 数据时采用相同的处理策略，

但在处理 X 数据（X 可以是 T、D 或 E）时有所不同。具体而言，两种方法都在 RGB 分支中使用了预训练的 ViT-B [?]，具有十二个自注意力层。对于 X 分支，ViPT 结合了十二个轻量级提示层，每层由三个卷积层组成。相比之下，SymTrack 采用对称结构，其中 RGB 和 X 分支共享参数。此外，在 ViPT 中，只有 RGB 分支的输出被送入追踪头，而在 SymTrack 中，RGB 和 X 分支的输出都被利用。

4 实验

4.1 实现细节

训练细节：本研究在配有 NVIDIA RTX 3090 GPU 和 AMD R9 7950X CPU 的平台上实现。具体来说，在训练过程中，epoch 和 batch 大小分别设置为 60 和 32。至于其他超参数，对于 ViPT [?]，我们遵循其官方实现，而对于 SymTrack，我们保留与 MPLT [?] 相同的配置。数据集：为了确保公平比较，我们的训练集与其他统一跟踪器保持一致，包括 VisEvent [?]、LasHeR [?] 和 DepthTrack [?]。这些数据集分别用于训练“统一-A”变体，并同时用于“统一-M”类别。随后，在这些数据集的测试集以及提出的 UniBench300 上进行评估。评估指标：著名的 PR [?]、归一化精度率（NPR）[?] 和 SR [?] 用于评价 LasHeR、VisEvent 和 UniBench300，而精度（Pr）[?]、召回率（Re）[?] 和 F-分数 [?] 则用于 DepthTrack。一般来说，PR、NPR、Pr 和 Re 测量预测的准确性。SR 评估成功跟踪帧的比例。F-分数结合了精度（Pr）和召回率（Re），提供了一个综合的排名指标。

4.2 UniBench300 的重要性

统一基准：表 1 显著突出了 UniBench300 作为第一个整合多任务数据的统一基准的优势。这使得在该基准上对分离方法和统一方法进行无缝评估成为可能。图 4 展示了先进分离方法和统一方法的追踪结果，其中采用统一模型的方法显然优于分离方法和采用统一结构的方法，强调了开发统一模型以提高鲁棒性的重要性。

方便且高效：表 4 比较了 ViPT* 和 SymTrack 在现有基准测试和 UniBench300 上的评估时间。例如，ViPT* 在 LasHeR、DepthTrack 和 VisEvent 等独立基准上分别需要三次单独评估，耗时分别为 65、24 和 38 分钟（总共 127 分钟）。相比之下，它在 UniBench300 上仅需执行一次，耗时 93 分钟，推论时间节省了 26.77 %。同样，SymTrack 在 UniBench300 上评估时显示出时间消耗减少了 27.03 %，从直观上强调了 UniBench300 提案带来的便利性和效率。

具有挑战性：图 5 展示了 UniBench300 上的基准测试结果。很明显，UniBench300 上的结果显著低于表 2 中其他基准测试的结果（0.395 对 0.570，低 30 %），这促进了一个更具挑战性的基准测试，并为推进 MMVOT 社区提供了更大的潜力。

4.3 持续统一的优越性

高级性能：表 2 比较了独立方法、统一架构的方法和跨多个基准（包括 LasHeR [?]、DepthTrack [?] 和 VisEvent [?]）的模型性能，其中“Prior”表示任务类型的先验信息。结果表明，独立方法或“统一-A”类别的方法通常比“统一-M”类别的方法表现更好。这种差异可归因于在统一过程中未能考虑模态差异。性能下降的进一步细节在表 3 中呈现。ViPT* 和 SymTrack 在所有涉及的任务中都显示出效率的降低。然而，当在串行范式下进行训练时，它们的性能提高，变得与其先前的变体相当甚至更好。具体而言，SymTrack+CL 在 LasHeR、DepthTrack 和 VisEvent 上分别比 SymTrack+mixed 高出 2.3 %、0.7 % 和 1.0

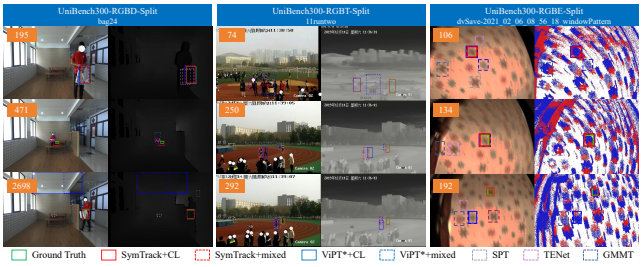

Figure 4: UniBench300 上的可视化。

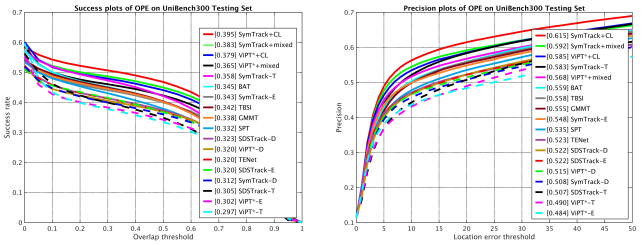

Figure 5: UniBench300 的基准测试结果。

%。值得注意的是，在 LasHeR 上，SymTrack+CL 在 PR 上比 SymTrack 高出 0.6 %（0.714 对比 0.708）、在 NPR 上高出 0.6 %（0.676 对比 0.670），在 SR 上高出 0.4 %（0.573 对比 0.569）。另外，图 5 展示了 UniBench300 上的基准测试结果，其中后缀“T”、“D”和“E”分别代表针对 RGBT、RGBD 和 RGBE 任务的变体。从图中可以看出，SymTrack+CL 在所有竞争对手中表现更佳，在 SR 上达到 0.395，在 PR 上达到 0.615。它在 SR 上比亚军（SymTrack+mixed）高出 1.2 %，在 PR 上高出 2.3 %，进一步凸显了 CL 在稳定统一过程中的有效性。

细粒度分析：与现有的并行加载多任务数据的方法相反，我们的持续统一过程逐步整合来自新任务的知识。这种方法允许进行细粒度分析，如图 1 所示。与其他仅针对一个或三个任务（共四个：T、D、E 以及 mix(T, D, E)）的变体进行评估的方法不同，我们在持续统一框架内尝试所有可能的排列和组合。这种全面的方法显著展示了 CL 在统一 MMVOT 任务方面的优越性，通过提高所有十五个变体的性能。

从表 2 可以看出，CL 在恢复 ViPT*+mixed 和 SymTrack+mixed 的性能方面显著受益。值得注意的是，观察到了两个现象：SymTrack 的恢复程度小于 ViPT*，而它们在 LasHeR 上的性能提升比在 DepthTrack 和 VisEvent 上更大。这个见解激发了对统一过程的进一步探索。为了更加直观，表 3 展示了从 ViPT* 和 SymTrack 到 ViPT*+mixed 和 SymTrack+mixed 的性能下降情况。该表显示，ViPT* (SymTrack) 分别在 LasHeR、DepthTrack 和 VisEvent 上出现了 3.30 % (2.20 %)、2.57 % (1.13 %) 和 1.2 % (0.80 %) 的下降。这表明在统一后，ViPT* 的性能下降比 SymTrack 更大（ViPT* > SymTrack）。此外，两种方法在 LasHeR 上显示出最大的性能下降，其次是 DepthTrack，VisEvent 上的下降最小（RGBT > RGBD > RGBE）。

退化：ViPT*>SymTrack：图 3 展示了 ViPT* 和 SymTrack 的架构。两者在 RGB 分支中使用相同的架构，但在 X 分支中有所不同。这导致一个合理的假设，即性能退化与 X 分支的大小有关，因为 ViPT* 使用仅具有三个卷积层的轻量级提示层，而 SymTrack 采用具有二次复杂度的 transformer 模块。这也符合

Table 2: 与多模态跟踪器在 VisEvent、DepthTrack 和 LasHeR 基准上的比较。"分离"表示该方法是为特定任务量身定制的。上标"*"表示该方法在本研究中进行了复现。

Type	Method	Venue	Prior		LasHeR			DepthTrack			VisEvent	
			Train	Test	PR ↑	NPR ↑	SR ↑	Pr ↑	Re ↑	F-score ↑	PR ↑	SR ↑
Separated	GMMT [?]	AAAI'2024	✓	✓	0.707	0.670	0.566	-	-	-	-	-
	BAT [?]	AAAI'2024	✓	✓	0.702	-	0.563	-	-	-	-	-
	SSLTrack [?]	PR'2024	✓	✓	-	-	-	0.565	0.491	0.525	-	-
	VADT [?]	ICASSP'2024	✓	✓	-	-	-	0.606	0.603	0.610	-	-
	eMoE-Tracker [?]	Arxiv'2024	✓	✓	-	-	-	-	-	-	0.764	0.613
	TENet [?]	NN'2025	✓	✓	-	-	-	-	-	-	0.765	0.601
Unified-A	ProTrack [?]	ACMMM'2022	✓	✓	0.509	-	0.421	0.583	0.573	0.578	0.617	0.474
	ViPT* [?]	CVPR'2023	✓	✓	0.645	0.614	0.519	0.587	0.611	0.598	0.754	0.591
	MixRGBX [?]	NEUCOM'2024	✓	✓	0.672	-	0.536	0.593	0.609	0.601	0.774	0.602
	EMTrack [?]	TCSVT'2024	✓	✓	0.659	-	0.533	0.580	0.585	0.583	0.724	0.584
	SDSTrack [?]	CVPR'2024	✓	✓	0.665	0.631	0.531	0.619	0.609	0.614	0.767	0.597
	OneTracker [?]	CVPR'2024	✓	✓	0.672	-	0.538	0.607	0.604	0.609	0.767	0.608
	SymTrack	-	✓	✓	0.708	0.670	0.569	0.592	0.617	0.604	0.770	0.606
Unified-M	Un-Track [?]	CVPR'2024	✓	✓	0.646	-	0.513	0.610	0.610	0.610	0.755	0.589
	XTrack [?]	Arxiv'2024	✓	✗	0.655	-	0.525	0.597	0.597	0.598	0.756	0.591
	ViPT*+mixed	CVPR'2023	✗	✗	0.609	0.576	0.494	0.562	0.584	0.573	0.743	0.579
	ViPT*+CL	-	✓	✗	0.652	0.618	0.527	0.584	0.608	0.596	0.758	0.592
					+4.3 %	+4.1 %	+3.3 %	+2.2 %	+2.4 %	+2.3 %	+1.5 %	+1.3 %
	SymTrack+mixed	-	✗	✗	0.682	0.649	0.550	0.581	0.605	0.593	0.763	0.597
	SymTrack+CL	-	✓	✗	0.714	0.676	0.573	0.587	0.613	0.600	0.771	0.607
					+3.2 %	+2.7 %	+2.3 %	+0.6 %	+0.8 %	+0.7 %	+0.8 %	+1.0 %

Table 3: 从统一的架构方法转换 ViPT* 和 SymTrack 到统一模型后的观察。

Type	Method	LasHeR			Mean	DepthTrack			Mean	VisEvent		Mean	
		PR ↑	NPR ↑	SR ↑		Pr ↑	Re ↑	F-score ↑		PR ↑	SR ↑		
Unified-A	ViPT*	0.645	0.614	0.519		0.587	0.611	0.598		0.754	0.591		Degradation:
Unified-M	ViPT*+mixed	0.609	0.576	0.494		0.562	0.584	0.573		0.743	0.579		
		-3.6 %	-3.8 %	-2.5 %	-3.30 %	-2.5 %	-2.7 %	-2.5 %	-2.57 %	-1.1 %	-1.3 %	-1.20 %	ViPT*
Unified-A	SymTrack	0.708	0.670	0.569		0.592	0.617	0.604		0.770	0.606		∨
Unified-M	SymTrack+mixed	0.682	0.649	0.550		0.581	0.605	0.593		0.763	0.597		SymTrack
		-2.6 %	-2.1 %	-1.9 %	-2.20 %	-1.1 %	-1.2 %	-1.1 %	-1.13 %	-0.7 %	-0.9 %	-0.80 %	
Degradation: RGBT > RGBD > RGBE													

Table 4: 时间消耗的比较。

Method	LasHeR	DepthTrack	VisEvent	UniBench300	Δ
ViPT*	65min	24min	38min	93min	-26.77 %
SymTrack	76min	28min	44min	108min	-27.03 %

直觉，即更大的网络有能力容纳更多知识。在此基础上，进行了一系列实验，通过逐步减少 X 分支的层数，同时保持 RGB 分支不变，结果如表 5 所示。当网络深度从 12 层减少到 2 层时，性能退化逐渐从 2.20 % 增加到 3.00 %。这支持了我们的假设，并得出结论，即较大的网络在统一后经历较少的性能退化。这也为为什么 ViPT* 比 SymTrack 感受到更大的退化提供了直观的解释，并为未来关于统一的研究提供了宝贵的见解。

劣化：RGBT>RGBD>RGBE：这些任务中最显著的差异在于 X 分支的数据类型。因此，数据模态的差异可能是这一现象的原因。根据结果，T/D/E 与其他两种模态之间距离的总和应该逐渐减少，因为较大的距离表明相似性较低，这通常导致多模态系统中的性能下降 [? ? ? ? ? ?]。图 6 通过可视化 T、D 和 E 数据的分布验证了这一点。为了量化这些差异，进一步引入了 C1、C2 和 C3。例如，C2 表示 T 和 E 之间的距离，C3 测量 T 和 D 之间的距离，而它们的组合（C3+C2）反映了 T 到 D 和 E 的总结距离。从这个图中，一个明确的模式出现：（C3+C2）>（C3+C1）>（C2+C1）。这意味着 T、D、E 相对于其他两种模态的距离依次减少，这与实验观察结果一致。

此外，为了消除图 6 中的主观性，例如用于可视化的数据选择或用于距离计算的中心点（浅蓝色点）的选择，进行了额外的交叉验证实验。结果显示在图 7 中。在左侧，对于 LasHeR，使用 RGBT 数据训练的模型在 SR（T->T）上达到了 0.569。使用 RGBD（D->T）和 RGBE（E->T）数据训练的模型分别达到 0.394 和 0.428。与 T->T 相比，性能分别在 D->T 和 E->T 上下降了 30.76 % 和 24.78 %。这表明 T 和 D 之间的差异大于 T 和 E

Table 5: 在 LasHeR 上的降解和网络深度。

Layers	LasHeR PR ↑	LasHeR NPR ↑	LasHeR SR ↑	Mean	Layers	LasHeR PR ↑	LasHeR NPR ↑	LasHeR SR ↑	Mean
	0.708	0.670	0.569			0.690	0.651	0.552	
12	0.682	0.649	0.550		6	0.658	0.623	0.528	
	-2.6 %	-2.1 %	-1.9 %	-2.20 %		-3.2 %	-2.8 %	-2.4 %	-2.80 %
	0.698	0.659	0.557			0.685	0.643	0.545	
10	0.667	0.637	0.537		4	0.650	0.615	0.522	
	-3.1 %	-2.2 %	-2.0 %	-2.43 %		-3.5 %	-2.8 %	-2.3 %	-2.87 %
	0.697	0.657	0.557			0.673	0.632	0.538	
8	0.664	0.631	0.535		2	0.637	0.602	0.514	
	-3.3 %	-2.6 %	-2.2 %	-2.70 %		-3.6 %	-3.0 %	-2.4 %	-3.00 %

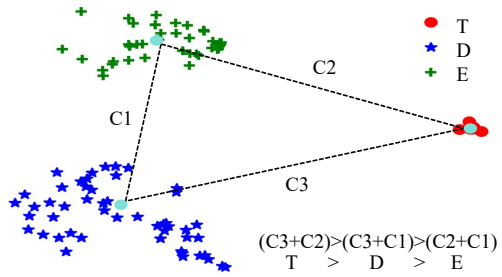

Figure 6: 数据 T、D 和 E 的分布。C1、C2 和 C3 表示模态之间的距离。

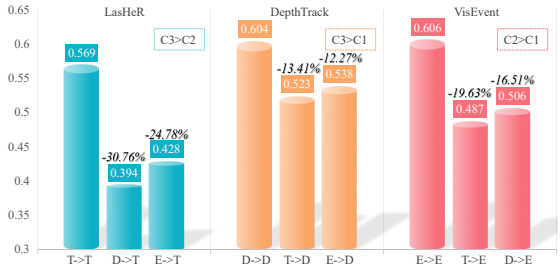

Figure 7: 模态差异的交叉验证。D->T 表示将使用 RGBD 数据训练的模型应用于 RGBT 基准。

之间的差异（C3 > C2）。同样地，同样的分析应用于中间和右边部分，得出结论（C3 > C1）和（C2 > C1）。基于这些，交叉验证结果证实了结论（C3 + C2）> （C3 + C1）> （C2 + C1），这与图 6 中的发现一致，进一步验证了我们分析的客观性和可靠性。

总之，我们的深入分析表明，统一后的性能下降与网络规模呈负相关，并且与任务有关（涉及更多异质数据的任务会遭受更大的降级）。

4.4 补充材料中包含什么

由于页面限制，我们将以下信息留在补充材料中：① 效率分析。② 所有之前任务的结果，补充了图 1（仅显示主要任务的表现）。③ 在嵌入空间中 CL 优势的定性分析。④ 所提出的持续统一过程的伪代码。⑤ UniBench300 的评价指标。⑥ 对连续统一中序列的见解。

在这项工作中，我们深入探讨了 MMVOT 任务的统一，并揭示了当前训练和测试范式之间的不一致性，这导致了性能下降。为了解决这些问题，我们推出了 UniBench300，这是第一个同时结合 RGBT、RGBD 和 RGBE 数据的多模态跟踪的统一基准。这个基准通过将推理与训练过程对齐来解决不一致性，提供了一个更便捷和高效的评估平台，只需进行一次推理（之前在分开的基准 LasHeR、DepthTrack 和 VisEvent 上需进行三次），减少了 27% 的时间。此外，从数据中心视角来看，原始的训练范式被重构为一个串行的方式，逐渐整合新任务，将性能下降视为 CL 中的一个关键挑战：先前任务的知识遗忘。这自然激发了将 CL 注入统一过程，从而显著增强了所有相关任务的稳定性。通过大量的实验，我们得出了两个重要的见解：统一后性能下降与网络规模负相关，并且依赖于任务，涉及更多异构数据的任务经历了更大的性能下降，为未来的研究提供了有价值的建议。补充材料可以在 https://github.com/Zhangyong-Tang/UniBench300/blob/main/Supplementary-material.pdf 找到。