arXiv:2601.00352v1 [cs.CV] 1 Jan 2026

OmniVaT：多模态视觉-触觉学习的单域泛化

Liuxiang Qiu[1], **Hui Da**[2], **Yuzhen Niu**[2],**Tiesong Zhao**[†1],**Yang Cao**[3],**Zheng-Jun Zha**[3]

[1]福建省媒体信息智能处理与无线传输重点实验室，福州大学物理与信息工程学院，中国福州 350108 [2]福州大学计算机与数据科学学院，中国福州 350108 [3]脑启发智能感知与认知教育部重点实验室，中国科学技术大学，中国合肥 230026

[†]通讯作者。

摘要： 视觉-触觉学习（VTL）通过整合视觉（VIS）和触觉（TAC）传感器，使具身智能体能够感知物理世界。然而，VTL仍面临VIS与TAC图像之间的模态差异，以及由触觉传感器非标准化和数据采集流程不一致所导致的领域差距。我们将这些挑战定义为一项新任务，称为多模态VTL单域泛化（SDG-VTL）。本文提出OmniVaT框架，首次成功解决了这一任务。一方面，OmniVaT集成了多模态分数阶傅里叶适配器（MFFA），将VIS和TAC嵌入映射到统一嵌入-频率空间，从而无需多域训练数据或精心设计的跨模态融合策略，即可有效缓解模态差距。另一方面，该框架还引入了离散树生成（DTG）模块，通过层次树结构获取多样且可靠的多模态分数表示，从而增强其对未见领域中不断变化的域偏移的适应能力。大量实验表明，OmniVaT在SDG-VTL任务上具有出色的跨域泛化性能。

项目页面： 项目页面即将发布。

MIC@FZU

1 引言

近年来，视觉-触觉学习（VTL）利用视觉（VIS）和触觉（TAC）传感器增强机器人感知，已在物体理解领域得到广泛研究[14, 37, 12, 43]。然而，TAC 传感器设计缺乏标准化，导致不同系列传感器的成像机制存在显著差异，进而产生迥异的触觉表征。例如，真实世界中的 GelSight 传感器 [42] 使用摄像头和 LED照明，捕捉物体接触时的凝胶形变；模拟的 Taxim 传感器 [32]则采用多项式查找表，近似弹性体形变后的光学响应；而 Tac3D 传感器 [45] 利用触觉皮肤和两个摄像头测量三维形变信息。此外，即使是同一类别中的物体（例如，金属物体），也会因表面属性和成像条件的差异而呈现出多种视觉外观，如图1(a)所示。这些差异造成了显著的模态与领域差距，使得建立统一的视觉-触觉特征空间十分困难。现有的 VTL 方法 [14, 38, 12] 建立在这样一种假设之上：测试域分布被训练域分布所涵盖；这限制了它们在真实世界未见场景中的泛化能力。在此，我们将由不同触觉传感器在不同场景下采集的视觉-触觉数据分布称为一个特定域。例如，单域到单域方法 [14, 38] （如图1(b)所示）泛化能力较差，因为它们是在与训练域具有相同分布的测试域上进行评估的。以往的多域到单域方法[12] 先在多个 VIS-TAC 域上训练模型，再在目标域上进行微调，以获得更有效的视觉-触觉表征并减轻模态差异，如图1(c)所示。然而，这些方法需要数百万个训练样本，使数据收集和模型训练的成本高得难以承受。受上述挑战启发，我们提出了一项面向多模态 VTL 的单域泛化（**SDG-VTL**）任务，旨在实现稳健的跨域 VTL。

单域泛化（SDG） [4, 26, 52, 36]旨在从单一源域中学习，并泛化到多个未见领域，在视觉泛化方面取得了令人瞩目的成果。尽管一些 SDG 方法 [26, 52] 采用对抗学习来模拟跨域表征，但它们往往面临不稳定收敛和不可靠生成的问题。其他 SDG 方法 [4, 17] 采用度量学习来获取域共享特征，但未能有效利用多模态线索，导致跨模态对齐效果欠佳，并限制了在未见 VIS-TAC 域上的泛化能力。近年来，视觉语言模型 [31, 30] 通过从大规模图文对中学习可迁移表征，在提升视觉泛化能力方面展现出了显著成效。然而，VIS 传感器捕获的是全局感受野

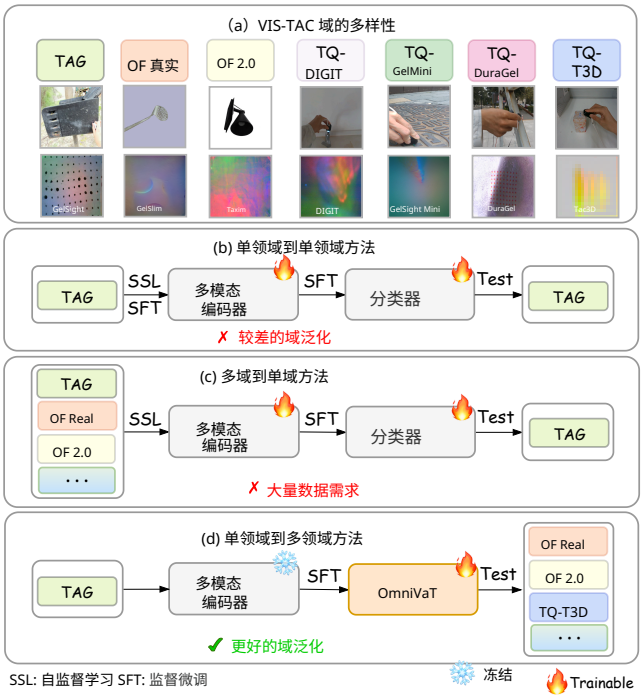


图1 (a) 各种 VIS-TAC 领域的可视化（例如，金属物体），(b) 单领域到单领域方法 [14, 38], (c) 多域到单域方法 [12], (d) 我们的单领域到多领域方法。

，而 TAC 传感器本质上具有局部性，并受限于接触区域，如图所示。1（a）。这种巨大的模态差距阻碍了现有域泛化方法在直接应用于跨域 VTL 时的有效性。

现有的 VTL 方法通常采用跨模态融合 [33, 24, 46] 或共同特征映射 [38, 9, 12]来缓解模态差异。然而，这些方法仅在单一嵌入空间内处理差异，却未能学习具有判别性的决策边界来缩小模态差距。一些多模态方法 [53, 7, 49] 尝试通过傅里叶变换将特征映射到频域，以减轻模态差异。然而，对原始特征空间探索不足，阻碍了不同模态间共同语义信息的发现。因此，上述单一嵌入空间或基于频率的方法可能无法构建具有判别性的多模态特征空间，因而无法缩小模态差距。与嵌入或频域处理不同，分数阶傅里叶变换 [51, 41, 6, 22] 将特征映射到统一嵌入-频率空间，为特征投影提供了更有效的视角。这有助于缓解同一类别特征之间的模态差异和域差异。

为应对上述挑战：模态差距 和域差距，我们提出了 OmniVaT，这是一种新颖的单一视觉-触觉域泛化框架，包含一个多模态分数阶傅里叶适配器（MFFA）模块和离散树生成（DTG）模块。我们方法的关键创新在于新颖地提出了❶利用统一的嵌入 - 频率表示以及 ❷ 扩展多样且可靠的特征节点，以学习具有判别性且合理的多模态特征空间，从而大幅提升模型缓解模态差异和域差异的能力。我们的主要贡献总结如下：

- 我们观察到，VTL 中模态差距与领域差距并存，因此提出一种面向实际应用的新型 SDG-VTL 任务。
- 我们率先采用新颖的 OmniVaT 框架解决单域泛化视觉触觉学习任务。该框架包含 MFFA 模块，可在统一嵌入-频率空间中对齐 VIS-TAC 特征；还包含 DTG 模块，可将分数表示离散化为层次树，从而分别缓解模态差距与领域差距。
- 我们提出的 OmniVaT 平均准确率达到 66.9%，在八种 SDG-VTL 设置下均至少领先现有最先进方法 13.0%。

2 相关工作

视觉-触觉学习（VTL）。为缓解 VIS 与触觉图像之间的模态差异，现有 VTL 方法可分为跨模态融合和共同特征映射两类。基于跨模态融合的方法 [33, 24, 46] 尝试融合 VIS 和 TAC 模态特征，以建立跨模态特征关联，从而减小模态差异。例如，Li等人[24]提出了一种基于注意力的视觉-触觉融合网络，通过融合 VIS 和 TAC 特征来提升识别性能。共同特征映射则因其通过度量学习获得模态共享特征的简便性而受到广泛关注 [38, 9, 12]。例如，Feng 等人[12] 尝试通过自监督学习，利用大量视觉-触觉数据学习多传感器不变表示，以缓解模态差距。然而，这些方法仅在单一嵌入空间中缓解模态差异，并假设训练-测试分布相同，因此无法泛化到未见场景。

域泛化。域泛化（DG）致力于解决域偏移问题，即目标域的数据分布不同于训练所用源域的数据分布。单域泛化（SDG） [4, 52, 26, 36, 17] 旨在仅有一个源域可用时实现域泛化。例如，Xu等[36]通过风格化和去风格化操作缓解域偏移，但存在收敛不稳定和生成不可靠的问题。Gupta等[17] 通过对比学习学习域共享特征，但未能充分利用多模态线索，因此对 VIS-TAC 领域的泛化能力有限，

有几种域泛化方法[8, 44, 35]尝试引入视觉语言模型[31]来提取域共享特征，但无法泛化到预训练数据中未包含的未见触觉域。基于频率的域泛化方法 [34, 28, 15, 20]利用傅里叶变换获得域不变表示，但依赖傅里叶空间与嵌入空间之间复杂的融合。相比之下，分数阶傅里叶变换 [51, 50, 41, 6, 22]提供了可调的嵌入-频率投影，以提取域不变特征。尽管 Zhao 等[50]采用分数阶融合进行空谱适应，但其方法仍需在目标域上微调，且缺乏有效对齐，无法缓解分数阶傅里叶空间中的模态差异。

嵌入扩展。 为增强模型提取具有判别性的域共享或类别共享特征的能力，一系列研究 [21, 48, 49] 尝试通过嵌入扩展生成多样化的特征表示，以克服域偏移。Ko 等人[21] 在两个特征点之间进行线性插值扩展，并在嵌入空间中生成合成点，用于度量学习。Zhang等人[49]采用卷积层生成多样的振幅分量差异，用于可见光-红外行人重识别。上述方法在嵌入空间或频率空间中进行扩展，却忽略了统一的嵌入-频率空间能够提供更有效的类别不变特征表示。此外，上述依赖串行或并行分支的方法缺乏对嵌入表示进行离散化的有效机制，因此限制了它们模拟各种域偏移的能力。

3 方法

3.1 SDG-VTL任务定义

现有VTL方法 [38, 9, 12] 通常假设测试域分布与训练分布相似。因此，由于VIS-TAC域的多样性，这些方法可能会对已见域特征过拟合，无法有效识别来自未见VIS-TAC领域的物体。受广受认可的单域泛化 [26, 52, 36], 启发，我们提出多模态VTL单域泛化（SDG-VTL）任务，在控制未见目标域最坏情况风险的同时，最小化单源VIS-TAC域 $\mathcal{S}$ 风险

$\mathcal{T}$，也就是说

$$\min_{\theta} \sup_{\mathcal{T}:G(\mathcal{S},\mathcal{T})\leq\rho} \mathbb{E}_{(x^v,x^t,y)\sim\mathcal{T}}\left[\ell(\theta;(x^v,x^t,y))\right], \quad (1)$$

其中， G 是用于度量域距离的相似度度量， ρ 表示 $\mathcal{S}$ 与$\mathcal{T}$之间最大的域差异； x^v 和 x^t 分别是视觉和触觉特征； y 是真实标签； θ 是网络的参数； ℓ(;)是任务评估（例如，交叉熵损失）。

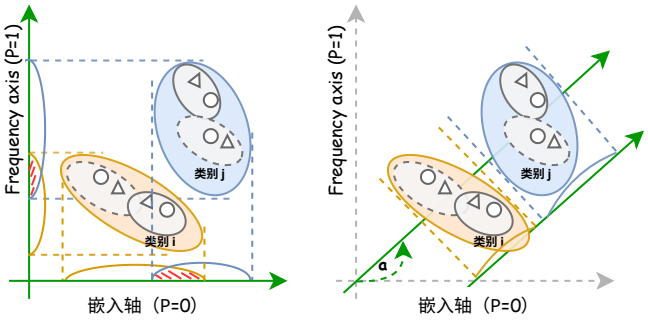


图2 嵌入（$p = 0$）或频率（$p = 1$）处理会导致投影重叠和类间混淆。然而，分数阶傅里叶变换（FrFT， $p = a$）能够提供统一的嵌入-频率表示，从而更有效地区分不同类别的特征。圆形和三角形分别表示视觉和触觉特征。

3.2 动机

如图1（a）所示，模态与领域差距是阻碍现有域泛化和VTL 方法有效学习判别性特征的主要问题。分数阶傅里叶变换（FrFT） [51, 41,6, 22] 允许在任意角度下灵活地在嵌入空间与频率空间之间进行变换，从而为捕获表示的语义一致性提供了更强大的机制，如图2所示。这可以被视为获得类别无关特征的一种有效方法。嵌入 p 的 阶 FrFT $\mathbf{E} \in \mathbb{R}^{1\times D}$定义如下：

$$\mathrm{FrFT}_p(\mathbf{E}(u_0)) = \int_{-\infty}^{\infty} K_p(u_0,u_p)\mathbf{E}(u_0)\,du_0, \quad (2)$$

其中， K_p（u_0， u_p）是分数阶傅里叶变换的核函数； u_0 指原始嵌入空间， u_p 表示变换后的分数阶空间。**有关 FrFT 的更多细节，请参阅补充材料。**

随后，我们基于 FrFT 设计模块，以解决模态与领域差距。对于**模态差距：** 现有的 VTL 方法 [38, 46], 主要依赖特征融合或度量对齐，往往需要精心设计模块，且语义利用不足。语言（LANG）信息提供类别的高层抽象表示，有助于网络获得模态共享特征。因此，我们旨在设计一个多模态分数阶傅里叶适配器（MFFA）模块，利用 FrFT 在语言信息的引导下将可见光-热成像特征映射到统一分数阶空间，以减轻模态差异。语言信息仅在训练期间使用，因为其中包含类别信息，因此无法用于推理。

对于**域差距：**现有的单域泛化方法通常利用对抗学习或特征增强来增加域多样性。然而，由于风格化和去风格化过程的不稳定性，这会

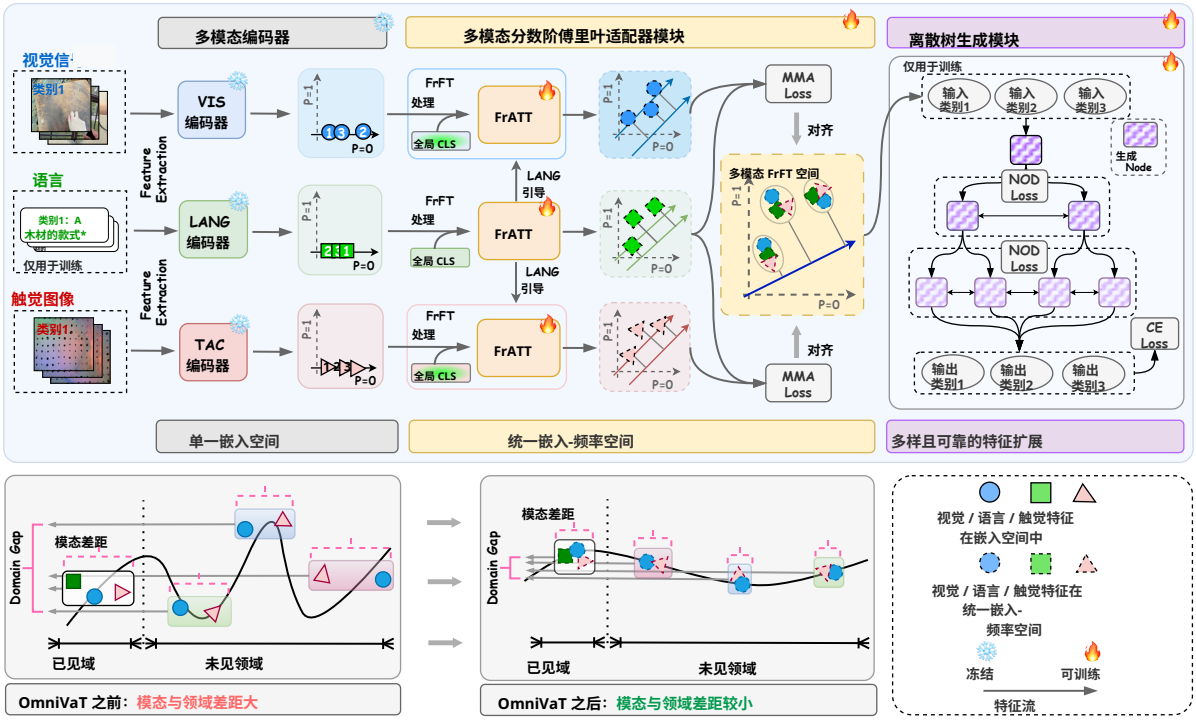


图3 所提出的 OmniVaT 概览，包括冻结的多模态编码器、采用分数阶傅里叶注意力（FrATT）机制的多模态分数阶傅里叶适配器（MFFA）模块，以及离散树生成（DTG）模块。OmniVaT 通过多模态对齐（MMA）损失、节点多样性（NOD）损失和交叉熵（CE）损失进行联合优化。冻结的编码器源自 CLIP [31]。MFFA 和 DTG 模块在不同模态之间共享同一组参数。由于 LANG嵌入包含类别信息，因此我们仅在训练期间使用它。

导致无法生成可靠的特征。尽管一些研究 [21, 48, 49] 扩展嵌入以缓解域偏移，但它们采用串行或并行扩展设计，难以有效生成多样化的特征表示。基于树的学习方法 [40, 29, 39]确保扩展特征的连续性，同时保持特征之间的多样性，因此自然成为学习多种特征表示的有效方案，从而提升模型的泛化性能。基于上述观察，我们提出离散树生成（DTG）模块，用于探索并学习多样化的分数表示集，从而增强模型对未见域的单域泛化能力。

3.3 所提出的 OmniVaT 概览

我们提出的 OmniVaT 概览如图所示。3。OmniVaT 分别由冻结的 VIS、触觉和语言编码器、一个 MFFA 模块以及一个 DTG 模块组成。我们采用 PromptStyler [8] 来生成缺失的语言信息（例如，“一种 [类别]的风格*”）。

3.4 多模态分数阶傅里叶适配器（MFFA）模块

假设我们拥有单源VIS-TAC域$\mathcal{D}_s = \{x_i^v, x_i^t, y_i\}_{i=1}^N$及其辅助语言信息$x_i^l$，其中 N 表示样本数量。经过冻结的 VIS 编码器、触觉编码器和语言编码器后，我们分别得到 VIS 嵌入$\mathbf{E}^v \in \mathbb{R}^{1\times D}$、触觉嵌入$\mathbf{E}^t \in \mathbb{R}^{1\times D}$和 LANG嵌入$\mathbf{E}^l \in \mathbb{R}^{1\times D}$。其中， D 表示嵌入的维度。为了缩小模态差距，我们提出 MFFA 模块，该模块借助跨模态分数阶傅里叶注意力机制，在语言信息的引导下将 VIS 和触觉嵌入映射到分数阶嵌入频率空间，并通过多模态对齐损失进一步缓解模态差异。该模块无需精心设计跨模态融合，同时能够更全面地挖掘模态间的信息。首先，我们引入分数阶傅里叶变换处理，将语言嵌入映射到统一分数阶空间，即

$$\begin{aligned}\mathbf{F}^l &= \text{FrFT Processing}(\mathbf{E}^l) \\ &= \mathcal{U}(\mathcal{R}(\text{FrFT}_p(\theta_{e,l}\mathbf{E}^l))) + j\,\mathcal{U}(\mathcal{I}(\text{FrFT}_p(\theta_{e,l}\mathbf{E}^l))),\end{aligned} \tag{3}$$

其中 $\theta_{e,l} \in \mathbb{R}^{E\times 1}$ 是可训练参数，用于扩展嵌入维度；$\text{FrFT}_p(\cdot)$ 表示 p阶分数阶傅里叶变换操

作； $\mathcal{R}(\cdot)$ 和 $\mathcal{I}(\cdot)$ 分别表示实部和虚部算子； $\mathcal{U}(\cdot)$ 表示 ReLU 操作； j 是虚数单位。

接着，我们提出一种分数阶傅里叶注意力（FrATT）机制，以获得编码后的分数阶 LANG 表征，其可表示为：

$$\begin{aligned}\bar{\mathbf{F}}^l &= \mathrm{FrATT}(\mathbf{F}^l, \mathbf{F}_g^l \oplus \mathbf{F}^l) \\ &= \mathrm{FFN}(\mathrm{Mean}(\frac{\mathrm{SM}(\theta_Q \mathbf{F}^l \theta_K (\mathbf{F}_g^l \oplus \mathbf{F}^l)^{\mathrm{T}})}{\sqrt{D}} \theta_V (\mathbf{F}_g^l \oplus \mathbf{F}^l))),\end{aligned} \tag{4}$$

其中，全局类别（CLS）令牌 $\mathbf{F}_g^l \in \mathbb{R}^{1\times D}$ 表示小批次中输入 $\mathbf{F}^l$ 同类别样本的平均值表征，可获取类别的全局信息以克服类内差异； $\oplus$ 表示广播加法操作； $\theta_Q \in \mathbb{R}^{D\times D}$、 $\theta_K \in \mathbb{R}^{D\times D}$ 和 $\theta_V \in \mathbb{R}^{D\times D}$ 是线性层，分别将 $\mathbf{F}^l$ 和 $(\mathbf{F}_g^l \oplus \mathbf{F}^l)$ 投影到查询空间、键空间和值空间；SM(·) 和 Mean(·) 分别表示 Softmax 和均值操作。FFN(·) 是全连接前馈网络。随后，在 LANG 特征 $\bar{\mathbf{F}}^l$ 的引导下，我们进一步将视觉嵌入 $\mathbf{E}^v$ 投影到统一的嵌入频率特征空间，以获得模态不变表征，即

$$\mathbf{F}^v = \mathrm{FrFT\ Processing}(\theta_{e,v}(\bar{\mathbf{F}}^l + \mathbf{E}^v)), \tag{5}$$

其中， $\theta_{e,\ v} \in \mathbb{R}^{E\times 1}$ 是可训练参数，用于扩展嵌入的数量。我们进一步采用 FrATT 机制来建立不同模态之间的语义关联，如下所示

$$\bar{\mathbf{F}}^v = \mathrm{FrATT}(\bar{\mathbf{F}}^l, \mathbf{F}_g^v \oplus \mathbf{F}^v). \tag{6}$$

与式（5）和（6）类似，我们也可以得到经过语言引导和 FrATT 机制处理后的分数阶触觉特征 $\bar{\mathbf{F}}^t$ 。为了更好地优化嵌入-频率空间中的分数阶，我们通过多模态对齐（MMA）损失对齐视觉、触觉和语言特征的表示，以缩小模态差距

$$\mathcal{L}_{\mathrm{MMA}} = \lambda(D_{\mathrm{KL}}(\bar{\mathbf{F}}^l, \bar{\mathbf{F}}^v) + D_{\mathrm{KL}}(\bar{\mathbf{F}}^l, \bar{\mathbf{F}}^t)), \tag{7}$$

其中， $D_{\mathrm{KL}}(\cdot)$表示 Kullback-Leibler（KL）散度函数， λ 用于调整 KL 散度的贡献。经过 MFFA 模块后，我们能够在统一嵌入-频率空间中高效地利用不同模态嵌入之间的关联，从而大幅减轻模态差异。

3.5 离散树生成（DTG）模块

为减小域偏移，我们提出一种 DTG 模块，借助树结构在统一的嵌入-频率空间中分层扩展多样且可靠的特征。具体而言，给定一个统一的嵌入-频率 VIS-TAC-语言特征 $\mathbf{T}$ (即 , $\bar{\mathbf{F}}^v + \bar{\mathbf{F}}^t + \bar{\mathbf{F}}^l$)，我们构建一棵二叉树，其中每个节点对应一个扩展后的特征。子节点通过作用于父节点的线性变换函数 $\mathcal{F}(\cdot)$ 生成，从而实现多层扩展。形式上，第 r 层的扩展过程可写为

$$\mathbf{T}_{m,n}^{(r+1)} = \mathcal{F}(\mathbf{T}_m^{(r)}, \mathbf{W}_{m,n}^{(r)}), \tag{8}$$

其中， $\mathbf{T}_m^{(r)}$表示第 m层的第 m个父特征节点， R是树的深度； $\mathbf{W}_{m,\ n}^{(r)} \in \mathbb{R}^{D\times D}$ 是第m个父节点的第 n个子节点的可学习参数。通过递归应用 $\mathcal{F}(\cdot)$，我们可以得到每一层扩展后的分数阶特征集 $\mathbf{T}^{tree} = \{\mathbf{T}_m^{(r)}\}$。为了进一步增强特征多样性并避免扩展节点之间的冗余，我们引入节点多样性（NOD）损失作为正则化项。通过最小化 NOD 损失（即，降低不同节点表示之间的余弦相似度），我们鼓励节点彼此独立，从而更分散地探索特征空间。具体而言，第 r 层的第 i个节点 $\mathbf{T}_i^{(r)}$ 与第 j个节点 $\mathbf{T}_j^{(r)}$ 之间的余弦相似度 $\mathbf{A}_{i,\ j}^{(r)}$ 可计算为

$$\mathbf{A}_{i,j}^{(r)} = \frac{\mathbf{T}_i^{(r)\top} \mathbf{T}_j^{(r)}}{\ell_2(\mathbf{T}_i^{(r)})\ell_2(\mathbf{T}_j^{(r)})}, \tag{9}$$

其中， $\ell_2(\cdot)$表示L2归一化操作。我们将 NOD 损失纳入其中，以便在不同层级对节点表示进行分层离散化，具体如下

$$\mathcal{L}_{\mathrm{NOD}} = \sum\nolimits_{r=1}^{R} \|\mathbf{A}^{(r)} - \mathbf{I}^{(r)}\|_F. \tag{10}$$

其中， $\mathbf{I}^{(r)}$ 是单位矩阵，其形状为 $2^{r-1} \times 2^{r-1}$； $\|\cdot\|_F$ 表示弗罗贝尼乌斯范数，它能提供稳定的梯度并促进更可靠的优化。NOD 损失可促进节点特征的结构化渐进解耦，并增强域表示中的视觉和触觉特征，即 $\hat{}\mathbf{F}^v$ 和$\hat{}\mathbf{F}^t$，即

$$\hat{\mathbf{F}}^v = \mathrm{Mean}(\mathbf{T}^{tree}) + \bar{\mathbf{F}}^v, \hat{\mathbf{F}}^t = \mathrm{Mean}(\mathbf{T}^{tree}) + \bar{\mathbf{F}}^t. \tag{11}$$

这样，所提出的 DTG 模块能够以结构化、渐进的方式探索多样化的特征空间，在避免以往基于嵌入的方法进行过度简化的并行扩展的同时，生成稳健的模态不变表征。联合损失定义为

$$\mathcal{L} = \mathcal{L}_{\mathrm{MMA}} + \mathcal{L}_{\mathrm{NOD}} + \mathcal{L}_{\mathrm{CE}}, \tag{12}$$

其中， $\mathcal{L}_{\mathrm{CE}}$ 表示交叉熵损失。

4 实验

数据集、任务设置、消融研究和比较结果的更多详情请见补充材料。

表 1 在八个 VIS-TAC 域上，SDG-VTL 与最先进方法的比较。“$\mathcal{S}$”/“$\mathcal{T}$”分别表示源域/目标域，而“→ X”表示对其他未见目标域（不包括已见源域）的评估。 ◊ 和 ⋈ 分别表示原始 TAC 学习方法和 VTL方法。最佳结果以 **粗体**标出。 Δ LDC 表示所提出的 OmniVaT 相较于 LDC 的提升。指标分别为准确率（ ACC，% ）和宏平均 F1 （ F1*，% ）。

方法 / 发表 venue	$\mathcal{S}$: TAG $\mathcal{T}$: →X 准确率 / F1*	$\mathcal{S}$: OF Real $\mathcal{T}$: →X 准确率 / F1*	$\mathcal{S}$: OF 2.0 A $\mathcal{T}$: →X 准确率 / F1*	$\mathcal{S}$: OF 2.0 B $\mathcal{T}$: →X 准确率 / F1*	$\mathcal{S}$: TQ-DIGIT $\mathcal{T}$: →X 准确率 / F1*	$\mathcal{S}$: TQ-GelMini $\mathcal{T}$: →X 准确率 / F1*	$\mathcal{S}$: TQ-DuraGel $\mathcal{T}$: →X 准确率 / F1*	$\mathcal{S}$: TQ-T3D $\mathcal{T}$: →X 准确率 / F1*	平均值 准确率 / F1*
SITR$^{\diamond}$ / ICLR-25 [17]	33.1 / 16.7	25.3 / 14.7	34.7 / 19.8	20.1 / 9.7	30.6 / 16.8	33.6 / 33.6	34.1 / 22.3	38.4 / 25.5	31.2 / 19.9
VT CMC$^{\bowtie}$ / NeurIPS-22 [38]	31.5 / 10.7	25.8 / 9.8	33.9 / 6.3	32.9 / 5.7	36.0 / 21.4	28.5 / 13.1	31.7 / 19.2	15.5 / 7.3	29.5 / 11.7
RobustVisH$^{\bowtie}$ / MM-25 [46]	32.7 / 17.9	30.9 / 10.8	32.9 / 11.6	32.5 / 11.5	27.1 / 10.4	27.5 / 10.7	28.1 / 11.9	23.5 / 9.7	29.4 / 11.8
使用 *CLIP* 预训练权重的 *ResNet-50 [18]* *[31]*									
CLIP / ICML-21 [31]	40.7 / 33.4	41.4 / 34.8	40.6 / 34.5	40.6 / 34.5	39.1 / 30.2	39.8 / 30.2	38.7 / 30.2	29.6 / 22.5	38.8 / 31.3
PromptStyler / ICCV-23 [8]	48.7 / 37.3	36.3 / 25.3	45.5 / 38.7	34.9 / 29.8	44.4 / 35.5	46.1 / 38.4	45.0 / 39.3	47.0 / 31.8	43.5 / 34.5
SPG / ECCV-24 [1]	47.9 / 43.3	42.8 / 38.7	45.2 / 42.9	38.5 / 29.7	58.7 / 52.5	56.4 / **54.7**	55.2 / 51.0	47.1 / 35.6	49.0 / 43.6
LDC / CVPR-25 [25]	43.9 / 33.3	43.2 / 32.7	44.8 / 33.8	52.3 / 43.7	38.3 / 22.0	37.1 / 22.6	39.4 / 25.6	43.9 / 27.9	42.9 / 30.2
OmniVaT / 我们的方法	**52.0/ 49.7**	**51.6 / 50.0**	**52.4/ 52.3**	**53.0 / 55.2**	**59.1 /54.6**	**57.6** / 52.7	**58.4 / 52.9**	**52.4/ 39.8**	**54.6 / 50.9**
Δ LDC	8.1 / 16.4	8.4 / 17.3	7.6 / 18.5	0.7 / 11.5	20.8 / 32.6	20.5 / 30.1	19.0 / 27.3	8.5 / 11.9	11.7 / 20.7
*ViT-B / 16 [11]*采用来自 *CLIP* 的预训练权重 *[31]*									
CLIP / ICML21 [31]	47.8 / 40.0	48.2 / 40.1	47.8 / 41.1	47.8 / 41.1	48.3 / 36.7	46.5 / 37.4	46.5 / 35.0	42.6 / 30.8	46.9 / 37.8
PromptStyler / ICCV-23 [8]	51.7 / 40.6	42.3 / 30.1	48.9 / 41.2	38.0 / 26.7	54.3 / 47.2	47.0 / 39.8	51.8 / 48.1	49.4 / 30.9	47.9 / 38.1
SPG / ECCV-24 [1]	49.4 / 41.9	49.8 / 42.0	49.1 / 42.8	49.4 / 42.0	53.1 / 46.9	53.8 / 48.6	51.6 / 46.3	50.4 / 43.8	50.8 / 44.3
MMRL / CVPR-25 [16]	46.0 / 33.5	45.9 / 45.4	42.9 / 30.1	42.3 / 37.2	36.0 / 17.9	27.3 / 15.2	29.6 / 14.4	48.8 / 34.8	39.9 / 28.6
LDC / CVPR-25 [25]	50.1 / 36.3	49.5 / 37.6	52.2 / 43.9	52.2 / 43.8	53.8 / 38.9	44.7 / 35.3	49.2 / 38.3	52.9 / 34.9	50.6 / 38.6
OmniVaT / 我们的方法	**56.2/ 53.7**	**58.4 / 56.5**	**58.6/ 58.2**	**61.2 / 60.3**	**56.4 /50.1**	**57.3 / 53.5**	**56.3 / 51.8**	**54.1/ 47.9**	**57.3 / 54.0**
Δ LDC	6.1 / 17.4	8.9 / 18.9	6.4 / 14.3	9.0 / 16.5	2.6 / 11.2	12.6 / 18.2	7.1 / 13.5	1.2 / 13.0	6.7 / 15.4
ViT-L / 14 [11] 使用 *CLIP* 的预训练权重 *[31]*									
CLIP / ICML-21 [31]	48.7 / 42.6	48.9 / 42.8	49.8 / 44.8	49.8 / 44.8	48.9 / 39.3	46.7 / 38.3	46.1 / 37.9	39.7 / 30.7	47.3 / 40.1
PromptStyler / ICCV-23 [8]	54.4 / 47.6	42.6 / 34.9	49.8 / 47.1	44.7 / 35.6	55.5 / 46.7	54.3 / 46.4	54.6 / 49.4	49.8 / 33.3	50.7 / 42.6
SPG / ECCV-24 [1]	57.5 / 54.5	53.6 / 49.7	47.8 / 39.8	39.0 / 31.9	61.0 / 53.4	55.7 / 49.3	65.0 / 61.8	51.9 / 46.3	53.9 / 48.3
MMRL / CVPR-25 [16]	59.5 / 54.6	35.2 / 35.1	34.3 / 22.9	44.6 / 39.4	40.6 / 41.4	33.3 / 22.8	30.1 / 19.6	51.2 / 43.2	41.1 / 34.9
LDC / CVPR-25 [25]	55.2 / 45.5	49.2 / 43.2	52.7 / 43.5	55.3 / 46.7	53.9 / 38.9	53.6 / 42.0	55.3 / 41.6	53.5 / 35.2	53.6 / 42.1
OmniVaT / 我们的方法	**62.6/ 64.1**	**64.9 / 65.8**	**67.4/ 68.0**	**67.2 / 68.5**	**68.5 /65.9**	**68.8 / 67.2**	**70.1 / 66.8**	**65.4/ 58.0**	**66.9 / 65.5**
Δ LDC	7.4 / 18.6	15.7 / 22.6	14.7 / 24.5	11.9 / 21.8	14.6 / 27.0	15.2 / 25.2	14.8 / 25.2	11.9 / 22.8	13.3 / 23.4

4.1 实验设置

1) 数据集与评估指标：我们从四个数据集收集视觉-触觉图像（即Touch-and-Go（TAG） [38],ObjectFolder（OF）真实数据集 [13], ObjectFolder（OF）2.0 [14],和 TacQuad（TQ） [12]），涵盖 8 个不同领域（即TAG、OF Real、OF 2.0 A、OF 2.0 B、TQ-DIGIT、TQ-GelMini、TQ-DuraGel 和 TQ-T3D），这些领域共享五种材料类别（即木材、钢、塑料、陶瓷和玻璃）。我们在单一源域上训练，并在其余未见目标域上进行评估，报告三个随机种子下的平均 Top-1 准确率（准确率）和宏平均 F1（F1*）。

2) 实现细节： 在训练步骤中，视觉和触觉图像被调整为 $3 \times 224 \times 224$，并通过随机水平翻转和特征归一化进行增强。我们选择公开可用的预训练模型 CLIP [31] 作为冻结的视觉、触觉和语言编码器。视觉和触觉编码器采用 ResNet-50 [18], ViT- B/16 [11], 或 ViT-L/14 [11], 实现；实验中语言编码器采用 Transformer。使用 ResNet-50 / ViT-B/16 / ViT-L/14 实现视觉和触觉编码器时，语言、视觉和触觉嵌入的维度分别为 D = 1024 / 512 / 768。在 FrFT 和 DTG 模块中，分数阶 p、扩展参数 E、平衡权重 λ以及树 R 的深度分别设为 0.5、4、10 和 3。对于 PromptStyler [8], , 我们为每个类别生成 80 个语言样本。对于每个小批次，随机选取 16 组成对的视觉-触觉样本及其辅助语言信息

表 2 在 TAG 上训练、并在未见过的 OF Real、OF 2.0 A 和 TQ-T3D 领域上测试的不同多模态学习方法的余弦相似度边际。

方法	OF Real	OF 2.0 A	TQ-T3D	[OF Real; TQ-T3D]	[OF 2.0 A; TQ-T3D]
CLIP [31]	0.036	0.075	0.033	0.007	0.006
PromptStyler [8]	0.034	0.061	0.035	0.013	0.017
SPG [1]	0.029	0.050	0.026	0.016	0.020
MMRL [16]	0.016	0.030	0.022	0.008	0.015
LDC [25]	0.041	0.040	0.046	0.028	0.017
OmniVaT	0.168	0.258	0.218	0.191	0.143

选定。我们采用余弦退火预热策略[27]来更新学习率，并将初始学习率设为 0.05。OmniVaT 使用动量为 0.9 的随机梯度下降进行优化，训练 20 个周期，并在 NVIDIA RTX 3090 GPU 上通过 PyTorch 实现。

4.2 与最先进方法的比较

1) SDG-VTL 的有效性：表1 显示，与先前采用共同特征映射的 VTL 方法（即，VT CMC [38]）相比，所提出的 OmniVaT 在 TAG → X 任务上的识别准确率提高了 20.5%。OmniVaT 的宏平均 F1 也达到 50.0%，比 RobustVisH [46]高出约 39.0%；后者通过跨模态融合在 OF 真实 → X 任务上整合多模态特征。尽管 SITR [17] 旨在通过在大规模模拟 TAC 数据上进行预训练来学习传感器不变表征，

表 3 所有方法均使用 TAG 领域，在单张 NVIDIA RTX 3090 GPU 上进行训练和测试。报告的指标包括可训练参数（M）、TAG 数据集上的训练时间（秒/轮）、GPU 内存（MB）、推理速度（FPS）、准确率（ACC，%）和宏平均 F1（F1*，%）。TAG → X 表示方法在 TAG 领域上训练，并在其他未见领域上测试。‡ 表示将特征维度降至 64。CLIP [31] 是零样本方法。

方法 / 发表 venue	可训练参数	训练 Time	GPU 内存	推理速度	TAG → X 准确率 / F1*	OF 2.0 A → X 准确率 / F1*
ViT-B / 16 [11] 使用 CLIP 的预训练权重 [31]						
CLIP / ICML-21 [31]	-	-	1,158	188.2	47.8 / 40.0	47.8 / 41.1
PromptStyler / ICCV-23 [8]	0.3	34.1	2,555	156.8	51.7 / 40.6	48.9 / 41.2
SPG / ECCV-24 [1]	4.7	23,180.2	4,936	91.1	49.4 / 41.9	49.1 / 42.8
MMRL / CVPR-25 [16]	5.0	58.3	3,603	163.3	46.0 / 33.5	42.9 / 30.1
LDC / CVPR-25 [25]	3.4	23.2	2,011	172.5	50.1 / 36.3	52.2 / 43.9
OmniVaT / 我们的方法	5.1	141.9	2,845	75.2	56.2 / 53.7	58.6 / 58.2
OmniVaT ‡ / 我们的方法	0.2	98.4	2,779	111.1	55.0 / 54.1	56.3 / 55.1

但在从模拟场景迁移到真实场景的设置下训练时（例如，OF 2.0 A → X），我们提出的 OmniVaT 在准确率上至少超过 SITR 17.7%。表1 还显示，在 TQ-T3D [16] X 任务中，使用预训练的 ViT-B/16 时，我们的 OmniVaT 比先前的多模态学习方法 MMRL 和 LDC [25]的宏平均 F1 高出约 13.0%。在 TQ-DuraGel → X 任务中，使用预训练的 ViT-B/16 时，OmniVaT 的准确率比依赖生成模型获取类别不变特征的 SPG → 高出 4.7%。在 ViT-L/14 设置下，我们的 OmniVaT 在 8 个领域上的平均准确率和宏平均 F1 均达到最佳，分别为 66.9% 和 65.5%。上述实验结果表明，OmniVaT 能够有效缓解模态与领域差距，实现 VTL 的高效单域泛化。

为进一步评估不同方法在 TAG → X 任务上的效果，我们分别从未见的 OF Real、OF 2.0 A、TQ-T3D、 [OF Real; TQ- T3D], 和 [OF 2.0 A; TQ-T3D]领域中随机抽取 657,000、657,000、308,025、499,500 和 499,500 个类内/类间样本对，并报告余弦相似度间隔 [5], , 如表 2所示。这里， [;] 表示两个未见领域的拼接。我们的方法 OmniVaT 获得了最大的余弦相似度间隔，在未见的 [OF Real; TQ-T3D]和 [OF 2.0 A; TQ-T3D] 领域上分别达到 0.191 和 0.143。这些结果表明，我们的方法 OmniVaT 能有效缩小未见领域之间的模态与领域差距，从而增强跨域泛化。

2）计算成本：表 3总结了各方法在 TAG 数据集上的可训练参数（M）、训练时间（秒/轮）、GPU 内存（MB）、推理速度（以批量大小为 16 时的每秒帧数 FPS 衡量）、准确率（ACC，%）和宏平均 F1（F1*，%），以及单域泛化 TAG → X 任务上的表现。正如表 3所示，OmniVaT 相较于 SPG 将训练时间缩短了 23,038.3 秒 [1], ，同时将准确率从 49.4% 提升至 56.2%。值得注意的是，TAC 传感器通常以 30 FPS 捕获触觉图像，而 OmniVaT 的推理速度达到 75.2 FPS,

表 4 多模态 VIS-TAC 图像表征学习对 TAG → X 任务的影响。指标为准确率（准确率，%）和宏平均 F1（F1*，%）。

方法	训练模态	测试模态	TAG → X
	*ViT-B / 16 [11]*使用 *CLIP [31]* 的预训练权重		
	语言 / 视觉 / 触觉	视觉 / 触觉	准确率 / F1*
MMRL [16]	- / ✓ / -	✓ / -	14.4 / 5.5
	- / ✓ / ✓	✓/ ✓	14.3 / 5.5
	✓/ ✓/ -	✓ / -	51.0 / 42.6
	✓ / ✓/ ✓	✓/ ✓	46.0 / 33.5
LDC [25]	- / ✓ / -	✓ / -	46.8 / 34.4
	- / ✓ / ✓	✓/ ✓	42.5 / 26.7
	✓/ ✓/ -	✓ / -	52.9 / 43.5
	✓ / ✓/ ✓	✓/ ✓	50.1 / 36.3
OmniVaT（我们的方法）	- / ✓ / -	✓ / -	53.0 / 48.1
	- / ✓ / ✓	✓/ ✓	54.5 / 48.0
	✓/ ✓/ -	✓ / -	54.9 / 51.8
	✓ / ✓/ ✓	✓/ ✓	56.2 / 53.7

这表明其适合实时部署。为评估 OmniVaT 在资源受限条件下的性能，我们将表示维度从 512 降低至 64。轻量化的 OmniVaT‡ 不仅将可训练参数数量分别减少了 0.1M、4.5M、4.8M 和 3.2M，相较于 Pro mptStyler [8], SPG [1], MMRL [16], 和 LDC[25], , 还在 TAG → X 任务上取得了令人瞩目的 54.1% 宏平均 F1。这些实验表明，我们的 OmniVaT 在计算效率与性能之间实现了出色的平衡。

4.3 消融研究

1) 多模态 ***VTL*** 的有效性：如表 4 所示，与单一视觉模态的域泛化相比，我们的 Om- niVaT 有效减少了模态差异，并在 TAG → X 的语言信息训练泛化任务上将泛化性能提升了 1.9% 的宏平均 F1。这些结果表明，所提出的 Omni- VaT 有效缓解了不同模态之间的差异，并提升了域泛化能力。尽管先前的视觉语言方法 [16, 25] 借助单一视觉模态实现了良好的泛化，但它们难以处理

表 5 所提出的 OmniVaT 的关键组件分别对 TAG → X 和 OF 2.0 A → X 任务的影响。指标为准确率（准确率，%）和宏平均 F1（F1*，%）。

#	方法	TAG → X 准确率 / F1*	OF 2.0 A → X 准确率 / F1*
1	基线 + $\mathcal{L}_{\text{CE}}$	51.7 / 40.6	48.9 / 41.2
2	+MFFA（不含 $\mathcal{L}_{\text{MMA}}$）	54.5 / 51.2	51.4 / 54.6
3	+MFFA	54.8 / 52.5	56.9 / 56.2
4	+DTG（不含 $\mathcal{L}_{\text{NOD}}$）	53.2 / 48.0	53.8 / 52.1
5	+DTG	53.6 / 49.8	54.9 / 54.6
6	+MFFA+DTG	56.2 / 53.7	58.6 / 58.2

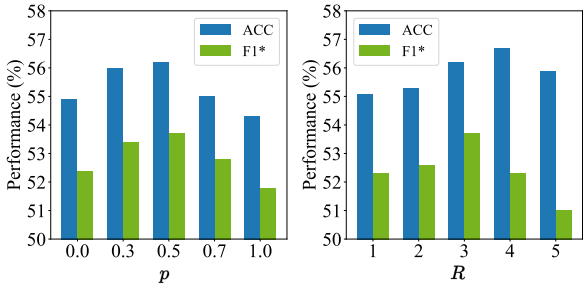


图4 分数阶 p 和树的深度 R 对 TAG → X 任务的影响。当 p = 0.0 或 1.0 时，其为固定值；否则，其为可学习的。

视觉-触觉模态差异，导致泛化性能下降。

2）关键组件的有效性：如表 5所示，我们评估了所提出的 OmniVaT 中关键组件的有效性。该方法由 MFFA 模块和 DTG 模块组成。我们分别在 TAG→ X 和 OF 2.0 A → X 任务上训练所提出的模块。方法 1 表示使用 CLIP 预训练 ViT-B/ [8] 权重的基线16 [11]方法 [31]。加入 MFFA 但不使用 MMA 损失（+MFFAw/o $\mathcal{L}_{\text{MMA}}$）后，方法 2 在 TAG → X 任务上的宏平均 F1 比基线提高了 10.6%，这表明 MFFA 通过 FrATT 在统一嵌入-频率空间中实现了有效的多模态嵌入交互。引入 MMA 损失后，方法 3 在 OF 2.0 A → X 任务上取得了 56.2% 的宏平均 F1，表明 $\mathcal{L}_{\text{MMA}}$ 进一步提升了 VIS-TAC 对齐并缓解了模态差异。同样，与方法 1 相比，加入 DTG 但不使用 NOD 损失（+DTG w/o$\mathcal{L}_{\text{NOD}}$）使 TAG →X 上的准确率提高了 1.5%。方法 5（+DTG）在 TAG → X 任务上的宏平均 F1 比方法 4 进一步提高了 1.8%，表明 DTG 能够有效地丰富分数表示，从而实现单域泛化。最后，结合 MFFA 和 DTG（方法 6）后，在 OF 2.0 A +lX 任务上，其准确率和宏平均 F1 分别比方法 3 高出 .7% 和 +2.0%，证实 OmniVaT 能够有效降低模态差异并增强域泛化。

3）树生成的有效性：为评估离散树生成（DTG）的影响，我们将

表6 不同特征生成方法分别对源 TAG 和 OF 2.0 A 领域的影响。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法	TAG → X 准确率 / F1*	OF 2.0 A →X 准确率 / F1*
基线	51.7 / 40.6	48.9 / 41.2
OmniVaT（线性插值）	54.6 / 51.5	56.5 / 56.2
OmniVaT（串行生成）	55.3 / 52.3	56.9 / 56.6
OmniVaT（并行生成）	55.0 / 51.8	57.3 / 57.4
OmniVaT（DTG）	56.2 / 53.7	58.6 / 58.2

OmniVaT 中的 DTG 模块替换为不同的特征生成方法，以生成七个特征节点，如表 6 所示。与线性插值 [21]，序列生成和并行生成 [48], 相比，我们的 OmniVaT 在 TAG → X 任务上的宏平均 F1 至少提升了 1.4%。这些结果表明，离散树生成能够有效获得多样且可靠的多模态分数阶表示，从而缩小领域差距。

4) 超参数：我们评估了分数阶 p 和树 R 深度的影响，如图 4所示。可以观察到，与原始嵌入空间（$p = 0.0$）和频率空间（$p = 1.0$）相比，统一分数阶域空间（$p = 0.5$能够更有效地实现多模态 VTL 的单域泛化。此外，当树深度 R 设为 3 时，OmniVaT 的整体性能最佳。

5 结论

在本工作中，我们观察到 VTL 天生同时存在模态与领域差距，并提出了面向实际应用的新型 SDG-VTL 任务。为解决 SDG-VTL 任务，我们提出了首个用于单一 VIS-TAC 领域泛化的 OmniVaT框架。OmniVaT 集成了 MFFA 模块，在统一嵌入-频率空间中对齐 VIS-TAC 表征，并通过 DTG 模块构建可靠的层次分数特征，从而有效缓解模态与领域差距。实验结果表明，在多种多模态 VTL 场景的单域泛化中，OmniVaT 的表现优于最先进方法，展现出其有效性与鲁棒性。OmniVaT 为将单域泛化从视觉模态扩展到更广泛的多模态学习提供了通用蓝图。未来的工作将扩展 OmniVaT 以处理振动触觉数据，从而克服现有基于图像的视觉触觉学习方法的普遍缺陷，实现更全面的物体理解。

参考文献

[1] Shuanghao Bai、Yuedi Zhang、Wanqi Zhou、Zhirong Luan 和 Badong Chen。用于域泛化的软提示生成。载于欧洲计算机视觉会议论文集，第 434–450 页，2024年。6，7，13，14

[2] Berk Calli、Arjun Singh、Aaron Walsman、Siddhartha Srinivasa、Pieter Abbeel 和 Aaron M Dollar。YCB物体与模型集：迈向操作研究的通用基准。载于《国际先进机器人会议论文集》，第 510–517 页，2015年。11[3] Cagatay Candan、M Alper Kutay 和 Haldun M Ozaktas。离散分数阶傅里叶变换。《IEEE信号处理汇刊》，48(5):1329–1337，2000年。11[4] Jin Chen、Zhi Gao、Xinxiao Wu 和 Jiebo Luo。用于单域泛化的元因果学习。载于《IEEE计算机视觉与模式识别会议论文集》，第 7683–7692 页，2023年。1, 2[5] Linwei Chen、Ying Fu、Lin Gu、Chenggang Yan、Tatsuya Harada 和 Gao Huang。用于密集图像预测的频率感知特征融合。《IEEE模式分析与机器智能汇刊》，2024年。7[6] Maoyang Chen、Shou Feng、Chunhui Zhao、Bo Qu、Nan Su、Wei Li 和 Ran Tao。基于分数阶傅里叶的频率-空间-光谱原型网络，用于农业高光谱图像开放集分类。《IEEE地球科学与遥感汇刊》，2024年。2, 3[7] Yaxiong Chen、Qicong Wang、Yichen Zhao、Shengwu Xiong 和 Xiaoqiang Lu。用于多模态遥感分类的双线性并行傅里叶变换器。《IEEE地球科学与遥感汇刊》，2025年。2[8] Junhyeong Cho、Gilhyun Nam、Sungyeon Kim、Hunmin Yang 和 Suha Kwak。PromptStyler：用于无源域泛化的提示驱动风格生成。载于《IEEE国际计算机视觉会议论文集》，第 15702–15712 页，2023年。3, 4, 6, 7, 8, 13, 14[9] Vedant Dave、Fotios Lygerakis 和 Elmar Rueckert。通过自监督对比预训练进行多模态视觉-触觉表示学习。载于《IEEE国际机器人与自动化会议论文集》，第 8013–8020 页，2024年。2, 3 – 。[10] Elliott Donlon、Siyuan Dong、Melody Liu、Jianhua Li、Edward Adelson 和 Alberto Rodriguez。GelSlim：高分辨率、紧凑、稳健且经过校准的触觉传感手指。载于《IEEE/RSJ国际智能机器人与系统会议论文集》，第 1927–1934 页。IEEE，2018年。11[11] Alexey Dosovitskiy、Lucas Beyer、Alexander Kolesnikov、Dirk Weissenborn、Xiaohua Zhai、Thomas Unterthiner、Mostafa Dehghani、Matthias Minderer、Georg Heigold、Sylvain Gelly 等。图像胜过千言万语：用于大规模图像识别的变换器。载于《国际学习表征会议论文集》，2020年。6, 7, 8, 13, 14[12] Ruoxuan Feng、Jiangyu Hu、Wenke Xia、Tianci Gao、Ao Shen、Yuhao Sun、Bin Fang 和 Di Hu。AnyTouch：在多种视觉-触觉传感器间学习统一的静态-动态表示。载于《国际学习表征会议论文集》，2025年。1, 2, 3, 6, 11, 12[13] Ruohan Gao、Yiming Dou、Hao Li、Tanmay Agarwal、Jeannette Bohg、Yunzhu Li、Li Fei-Fei 和 Jiajun Wu。ObjectFolder基准：利用神经与真实物体进行多感官学习。载于《IEEE计算机视觉与模式识别会议论文集》，第 17276–17286 页，2023年。6, 11, 12[14] Ruohan Gao、Zilin Si、Yen-Yu Chang、Samuel Clarke、Jeannette Bohg、Li Fei-Fei、Wenzhen Yuan 和 Jiajun Wu。ObjectFolder 2.0：用于仿真到现实迁移的多感官物体数据集。发表于《IEEE计算机视觉与模式识别会议论文集》，第10598–10608页，2022年。1、2、6、11、12[15] Hariseetharam Gunduboina、Muhammad Haris Khan 和 Bi- plab Banerjee。Frogdognet：用于遥感领域 CLIP 域泛化的傅里叶频率保留视觉提示输出引导。发表于《IEEE计算机视觉与模式识别会议论文集》，第2359–2372页，2025年。3[16] Yuncheng Guo 和 Xiaodong Gu。MMRL：面向视觉语言模型的多模态表征学习。发表于《IEEE计算机视觉与模式识别会议论文集》，第25015–25025页，2025年。6、7、13、14[17] Harsh Gupta、Yuchen Mo、Shengmiao Jin 和 Wenzhen Yuan。传感器不变触觉表征。发表于《国际学习表征会议》，2025年。1、2、6[18] Kaiming He、Xiangyu Zhang、Shaoqing Ren 和 Jian Sun。用于图像识别的深度残差学习。发表于《IEEE计算机视觉与模式识别会议论文集》，第770–778页，2016年。6[19] GelSight公司。GelSight Mini，2022年。11[20] Zhong Ji、Zhilong Wang、Xiyao Liu、Yunlong Yu、Yanwei Pang 和 Jungong Han。频率-空间互补：用于跨域少样本学习的统一通道特定风格攻击。《IEEE图像处理汇刊》，2025年。3[21] Byungsoo Ko 和 Geonmo Gu。嵌入扩展：嵌入空间中的数据增强，用于深度度量学习。发表于《IEEE计算机视觉与模式识别会议论文集》，第7255–7264页，2020年。3、4、8[22] Emirhan Koç、Tuna Alikaşifoğlu、Arda Can Aras 和 Aykut Koç。可训练分数阶傅里叶变换。《IEEE信号处理快报》，31：751–755，2024年。2、3[23] Mike Lambeta、Po-Wei Chou、Stephen Tian、Brian Yang、Ben- jamin Maloon、Victoria Rose Most、Dave Stroud、Raymond Santos、Ahmad Byagowi、Gregg Kammerer 等。DIGIT：一种新型低成本、紧凑型高分辨率触觉传感器设计及其在手内操作中的应用。《IEEE机器人与自动化快报》，5(3)：3838–3845，2020年。11[24] Baojiang Li、Jibo Bai、Shengjie Qiu、Haiyan Wang 和 Yuting Guo。Vito-Transformer：用于物体识别的视觉-触觉融合网络。《IEEE仪器与测量汇刊》，2023年。2[25] Shuo Li、Fang Liu、Zehua Hao、Xinyi Wang、Lingling Li、Xu Liu、Puhua Chen 和 Wenping Ma。利用 CLIP 进行少样本学习的逻辑值去混淆。发表于《IEEE计算机视觉与模式识别会议论文集》，第25411–25421页，2025年。6、7、13、14[26] Songhua Liu、Xin Jin、Xingyi Yang、Jingwen Ye 和 Xinchao Wang。StyDeSty：用于单域泛化的极小极大风格化与去风格化。《国际机器学习会议论文集》，2024年。1、2、3[27] Ilya Loshchilov 和 Frank Hutter。SGDR：带热重启的随机梯度下降。发表于《国际学习表征会议》，2017年。6[28] Wang Lu、Jindong Wang、Haoliang Li、Yiqiang Chen 和 Xing Xie。域泛化的域不变特征探索。《机器学习研究汇刊》，2022年。3

[29] Xupeng Miao、Gabriele Oliaro、Zhihao Zhang、Xinhao Cheng、Zeyu Wang、Zhengxin Zhang、Rae Ying Yee Wong、Alan Zhu、Lijie Yang、Xiaoxiang Shi 等。Specinfer：通过基于树的推测推理与验证加速大语言模型服务。发表于《ACM编程语言与操作系统架构支持国际会议论文集》，第932–949页，2024年。4[30] Maxime Oquab、Timothée Darcet、Théo Moutakanni、Huy Vo、Marc Szafraniec、Vasil Khalidov、Pierre Fernandez、Daniel Haziza、Francisco Massa、Alaaeldin El-Nouby 等。DINOv2：无需监督地学习鲁棒视觉特征。arXiv预印本 arXiv:2304.07193，2023年。1[31] Alec Radford、Jong Wook Kim、Chris Hallacy、Aditya Ramesh、Gabriel Goh、Sandhini Agarwal、Girish Sastry、Amanda Askell、Pamela Mishkin、Jack Clark 等。从自然语言监督中学习可迁移视觉模型。发表于《国际机器学习会议论文集》，第8748–8763页，2021年。1、3、4、6、7、8、13、14[32] Zilin Si 和 Wenzhen Yuan。Taxim：一种用于 GelSight 触觉传感器的基于示例的仿真模型。《IEEE机器人与自动化快报》，7(2)：2361–2368，2022年。1、11[33] Fuyang Wei、Jianhui Zhao、Chudong Shan 和 Zhiyong Yuan。视觉-触觉物体识别的对齐与多尺度融合。发表于《IEEE国际神经网络联合会议论文集》，第1–8页，2022年。2[34] Qinwei Xu、Ruipeng Zhang、Ya Zhang、Yanfeng Wang 和 Qi Tian。一种用于域泛化的基于傅里叶变换的框架。发表于《IEEE计算机视觉与模式识别会议论文集》，第14383–14392页，2021年。3[35] Xiusheng Xu、Lei Qi、Jingyang Zhou 和 Xin Geng。Bat-styler：推进无源域泛化的多类别风格生成。《IEEE电路与系统视频技术汇刊》，2025年。3[36] Zhipeng Xu、De Cheng、Xinyang Jiang、Nannan Wang、Dong- sheng Li 和 Xinbo Gao。用于单域泛化的对抗式域提示调优与生成。发表于《IEEE计算机视觉与模式识别会议论文集》，第18584–18595页，2025年。1、2、3[37] Fengyu Yang、Chao Feng、Ziyang Chen、Hyoungseob Park、Daniel Wang、Yiming Dou、Ziyao Zeng、Xien Chen、Rit Gan- gopadhyay、Andrew Owens 等。将触觉与万物关联：学习统一的多模态触觉表征。发表于《IEEE计算机视觉与模式识别会议论文集》，第26340–26353页，2024年。1[38] Fengyu Yang、Chenyang Ma、Jiacheng Zhang、Jing Zhu、Wen- zhen Yuan 和 Andrew Owens。Touch and Go：从人类采集的视觉与触觉数据中学习。发表于《神经信息处理系统进展》，2022年。1、2、3、6、11、12[39] Lei Yang。离散分布网络。发表于《国际学习表征会议》，2025年。4[40] Shunyu Yao、Dian Yu、Jeffrey Zhao、Izhak Shafran、Tom Grif- fiths、Yuan Cao 和 Karthik Narasimhan。思维树：利用大语言模型进行审慎问题求解。《神经信息处理系统进展论文集》，36：11809–11822，2023年。4[41] Hu Yu、Jie Huang、Lingzhi Li、Feng Zhao 等。深度分数阶傅里叶变换。《神经信息处理系统进展论文集》，36：72761–72773，2023年。2、3[42] Wenzhen Yuan、Siyuan Dong 和 Edward H Adelson。Gel-sight：用于估算几何形状和力的高分辨率机器人触觉传感器。《Sensors》，17(12):2762，2017年。1，11[43] Chaofan Zhang、Peng Hao、Xiaoge Cao、Xiaoshuai Hao、Shaowei Cui 和 Shuo Wang。Vtla：用于插入操作、结合偏好学习的视觉-触觉-语言-动作模型。arXiv 预印本 arXiv:2505.09577，2025年。1[44] Haoran Zhang、Shuanghao Bai、Wanqi Zhou、Jingwen Fu 和 Badong Chen。Promptta：用于无源域泛化的提示驱动文本适配器。载于 IEEE 国际声学、语音与信号处理会议论文集，第1–5页，2025年。3[45] Lunwei Zhang、Yue Wang 和 Yao Jiang。Tac3D：一种新型基于视觉的触觉传感器，用于测量力分布和估算摩擦系数分布。arXiv 预印本 arXiv:2202.06211，2022年。1，11[46] Rouqi Zhang、Chengdi Lu、Hancheng Lu、Yang Cao 和 Tiesong Zhao。RobustVisH：传输干扰下鲁棒的视觉-触觉跨模态识别。载于 ACM 多媒体国际会议论文集，2025年。2，3，6[47] Shixin Zhang、Yiyong Yang、Fuchun Sun、Lei Bao、Jianhua Shan、Yuan Gao 和 Bin Fang。一种用于微米级触觉感知的紧凑型视觉-触觉机器人皮肤。《IEEE Sensors Journal》，24(9):15273–15282，2024年。11[48] Yukang Zhang 和 Hanzi Wang。用于可见光-红外行人重识别的多样化嵌入扩展网络和低光跨模态基准。载于 IEEE 计算机视觉与模式识别会议论文集，第2153–2162页，2023年。3，4，8[49] Yukang Zhang、Hanzi Wang、Yang Lu、Yan Yan 和 Xuelong Li。可见光-红外行人重识别的频域细微特征挖掘。《IEEE Transactions on Information Forensics and Security》，20:5411–5424，2025年。2，3，4[50] Xudong Zhao、Mengmeng Zhang、Ran Tao、Wei Li、Wenzhi Liao 和 Wilfried Philips。使用分数阶融合和空间-光谱域适配对多源遥感数据进行跨域分类。《IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing》，15:5721–5733，2022年。3[51] Xudong Zhao、Mengmeng Zhang、Ran Tao、Wei Li、Wenzhi Liao、Lianfang Tian 和 Wilfried Philips。用于多模态遥感数据分类的分数阶傅里叶图像 Transformer。《IEEE Transactions on Neural Networks and Learning Systems》，35(2):2314–2326，2022年。2，3[52] Guangtao Zheng、Mengdi Huai 和 Aidong Zhang。Advst：重新审视单域泛化的数据增强。载于 AAAI 人工智能大会论文集，第21832–21840页，2024年。1，2，3[53] Youjie Zhou、Guofeng Mei、Yiming Wang、Yi Wan 和 Fabio Poiesi。采用傅里叶注意力的多模态融合 SLAM。《IEEE机器人与自动化快报》，10(2):1050–1057，2025年。2

更多分数阶傅里叶变换细节

嵌入 $\mathbf{E} \in \mathbb{R}^{1\times D}$ 的第 p 阶分数阶傅里叶变换（FrFT）定义如下:

$$\mathrm{FrFT}_p(\mathbf{E}(u_0)) = \int_{-\infty}^{\infty} K_p(u_0, u_p)\mathbf{E}(u_0)\, du_0, \qquad (13)$$

其中 $K_p(u_0,\ u_p)$ 是分数阶傅里叶变换的核函数，见表7，u_0 和 u_p 分别表示原始嵌入空间和分数阶空间。虽然式（13）定义了连续分数阶傅里叶变换，但实际的嵌入是离散信号。因此，我们采用离散分数阶傅里叶变换（DFrFT）[3],，即

$$\mathbf{E}(u_p)[m] = \sum_{n=1}^{D} \mathbf{F}_p[m,n]\mathbf{E}(u_0)[n], m = 1, \cdots, D \qquad (14)$$

其中 $\mathbf{F}_p[m,n] = \sum_{k=1}^{D} v_k(m)(\lambda_k)^p\, v_k(n)$ 是离散分数阶傅里叶变换矩阵 [3], v_k 是第 k个离散厄米-高斯向量，λ_k 表示相应的特征值。嵌入$\mathbf{E}$ 的最终 p阶离散分数阶傅里叶变换可表示为$\mathbf{E}(u_p) = \mathbf{F}^p\mathbf{E}$。

表7 不同 α $\alpha = p\pi/2$ A_α =值下分数阶傅里叶变换（FrFT）的核函数 $K_p(u_0,\ u_p)$ ()。(cot)和( )是狄拉克δ函数。

核表达式 $K_p(u_0,\ u_p)$	条件
A_α 指数 $j\,(u_0^2$余切 $\alpha/2 - u_p u_0$余割 $\alpha + u_p^2$余切$\alpha/2)$	$\alpha \neq n\pi$
$\delta(u_0 - u_p)$	$\alpha = 2n\pi$
$\delta(u_0 + u_p)$	$\alpha = (2n+1)\pi$

算法 1 OmniVaT 的训练过程。

1：输入：带有辅助语言信息的单源VIS-TAC域 $\mathcal{S} = \{x^v, x^t, y\}\, x^l$；OmniVaT 的初始参数 θ；训练轮数 H；小批次大小 B。2：输出：OmniVaT 的最终参数 θ'。3：对于 $h = 1$到 H 执行 4：对于 $\mathcal{S}$ 中的每个小批次 $\{x^v, x^t, x^l\}_b^B$执行 5：通过冻结的多模态编码器获取视觉、触觉和语言嵌入。6：通过 MFFA 模块将多模态视觉、触觉和语言嵌入映射到统一的嵌入-频率空间，以通过公式（3-6）缓解模态差距。7：应用 DTG 模块，获取多样且可靠的特征 $\mathbf{T}^{tree}$，以提升模型的域泛化能力。8：通过公式（12）计算联合损失。9：通过随机梯度下降更新 θ。10：结束循环 11：结束循环

算法 2 OmniVaT 的推理过程。

1: **输入：**未见目标视觉-触觉领域 $\mathcal{T} = \{x^v, x^t\}$; OmniVaT 的最终参数 θ'。2: **输出：**未见 VIS–TAC 样本的预测类别。3: **对于**每个 $\{x^v, x^t\}$ 在 $\mathcal{T}$**中，** 4: 通过冻结的多模态编码器获取视觉和触觉嵌入。5: 应用 MFFA 模块，将多模态视觉和触觉嵌入映射到统一的嵌入频率空间，并提高跨模态语义一致性。6: 根据多模态分数特征计算类别概率。7: **结束循环**

B 更多 OmniVaT 详情

所提出的 Omni- VaT 的训练和推理过程分别总结于算法1和2中。

C 更多数据集和任务详情

在 Touch-and-Go（TAG）数据集上，[38], 触觉数据由 GelSight 传感器采集，[42]OF Real 数据集 [13]使用 GelSlim 传感器，[10], 而 OF 2.0 [14] 提供由 Taxim 生成的模拟触觉信号，[32]。我们将 OF 2.0 数据集中的前 100 个物体记为 OF 2.0 A，这些物体来自 3D Model Haven、YCB [2], 和 Google Scanned Objects，其余物体则称为 OF 2.0 B。TacQuad（TQ）数据集 [12] 包含四种触觉传感器: DIGIT、[23], GelSight Mini、[19], DuraGel[47], 和 Tac3 D，[45], 由此形成 TQ-DIGIT、TQ-GelMini、TQ-DuraGel 和 TQ-T3D 领域。数据集统计信息汇总于表8中。由于 OF 2.0 A 和 OF 2.0 B 使用相同的、基于 Taxim 的触觉模态，为避免领域泄漏，我们不对二者进行跨域评估。由于 TQ 领域不包含陶瓷物体，因此在构建未见领域时，我们排除了陶瓷样本。此外，在 TQ 领域上训练时，我们将其划分为室内和室外系列，使模型在不重叠的数据上进行训练和评估。SDG-VTL 设置详见表9。

D 更多消融研究

1）全局类别标记的有效性：如表10所示，全局类别标记提升了 OmniVaT 的表现。具体而言，在 TAG $\rightarrow$ X 和 OF 2.0 A $\rightarrow$ X 任务中，采用全局类别标记的 OmniVaT 相较于未采用该标记的版本，宏平均 F1 分别提高了 0.2% 和 0.5%。这表明，全局类别标记能够减少类内变化并提升 SDG-VTL 的表现。

2）超参数：图 5 展示了准确率变化的情况——

表 8 数据集（领域）汇总。每个数据集（领域）均提供 VIS-TAC 样本。

领域	触觉传感器	训练集	测试集
TAG [38]	GelSight	5,520	1,770
OF 真实 [13]	GelSlim	3,401	1,355
OF 2.0 A [14]	Taxim	1,000	2,200
OF 2.0 B [14]	Taxim	9,500	2,500
*TQ*室内外系列			
TQ-DIGIT [12]	DIGIT	-	810
TQ-GelMini [12]	GelSight Mini	-	810
TQ-DuraGel [12]	DuraGel	-	810
TQ Indoor 系列			
TQ-DIGIT（室内） [12]	DIGIT	-	525
TQ-GelMini（室内） [12]	GelSight Mini	-	525
TQ-DuraGel（室内） [12]	DuraGel	-	525
TQ-T3D（室内） [12]	Tac3D	1,196	682
*TQ*户外系列			
TQ-DIGIT（室外） [12]	DIGIT	779	285
TQ-GelMini（室外） [12]	GelSight Mini	779	285
TQ-DuraGel（室外） [12]	DuraGel	779	285

表9 多模态视觉-触觉学习（SDG-VTL）任务的单域泛化设置。

已见域	未见领域
TAG [38]	OF 真实 [13],OF 2.0 A [14], OF 2.0 B [14],TQ-DIGIT[12], TQ-GelMini [12],TQ-DuraGel [12],TQ-T3D [12]
OF 真实 [13]	TAG [38],OF 2.0 A [14], OF 2.0 B [14],TQ-DIGIT [12], TQ-GelMini [12],TQ-DuraGel [12],TQ-T3D [12]
OF 2.0 A [14]	TAG [38],OF 真实 [13],TQ-DIGIT [12], TQ-GelMini [12],TQ-DuraGel [12],TQ-T3D [12]
OF 2.0 B [14]	TAG [38],OF 真实 [13],TQ-DIGIT [12], TQ-GelMini [12],TQ-DuraGel [12],TQ-T3D [12]
TQ-DIGIT [12]（室外）	TAG[38],OF 真实 [13], OF 2.0 A [14], OF 2.0 B [14],TQ-GelMini（室内） [12] , TQ-DuraGel（室内） [12] ,TQ-T3D（室内） [12]
TQ-GelMini [12]（室外）	TAG[38],OF 真实 [13], OF 2.0 A [14],OF 2.0 B [14], TQ-DIGIT（室内） [12], TQ-DuraGel（室内） [12], TQ-T3D（室内） [12]
TQ-DuraGel [12]（室外）	TAG[38],OF 真实 [13], OF 2.0 A [14],OF 2.0 B [14], TQ-DIGIT（室内） [12], TQ-GelMini（室内） [12],TQ-T3D（室内） [12]
TQ-T3D [12]（室内）	TAG[38],OF 真实 [13], OF 2.0 A [14], OF 2.0 B [14],TQ-DIGIT（室外） [12], TQ-GelMini（室外） [12], TQ-DuraGel（室外） [12]

表10 所提出的 OmniVaT 的全局类别标记分别对源 TAG 和 OF 2.0 A 领域的影响。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法	TAG →X	OF2.0 A →X
	准确率 / F1*	准确率 / F1*
基线	51.7 / 40.6	48.9 / 41.2
OmniVaT（不含全局类别标记）	55.8 / 53.5	58.3 / 57.7
OmniVaT（含全局类别标记）	56.2 / 53.7	58.6 / 58.2

以及 TAG → X 任务中准确率和宏平均 F1 随扩展参数 E 和平衡权重 λ 变化的情况。可以观察到，当$E = 4$和 $\lambda = 10$时，性能达到最优。

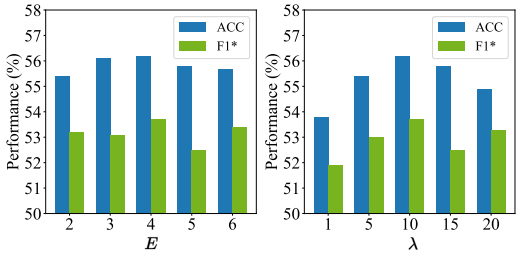


图5 扩展参数 E 和平衡权重 λ 对 TAG → X 任务的影响。

表明我们的 OmniVaT 能够为 VIS-TAC学习实现更有效的单域泛化。

E 更多比较结果

表格11 至 18 详细比较了 OmniVaT 与先前最先进方法在 8 个 VIS-TAC 领域上的表现。总体而言，OmniVaT取得了最高的平均性能。这

表11 在 TAG → X 任务上的最先进方法比较。“TAG → X”表示对其他未见目标域的评估，不包括已见源域 TAG。 ΔLDC 表示所提出的 OmniVaT 相较于 LDC 的提升。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ OF Real 准确率 / F1*	→ OF 2.0 A 准确率 / F1*	→ OF 2.0 B 准确率 / F1*	→ TQ-DIGIT 准确率 / F1*	→ TQ-GelMini 准确率 / F1*	→ TQ-DuraGel 准确率 / F1*	→ TQ-T3D 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11] 使用来自 CLIP 的预训练权重 [31]								
CLIP / ICML-21 [31]	49.0 / 48.8	51.6 / 37.5	48.1 / 46.0	37.3 / 32.6	47.5 / 31.8	44.4 / 31.8	56.9 / 51.5	47.8 / 40.0
PromptStyler / ICCV-23 [8]	47.5 / 37.4	46.0 / 24.8	43.3 / 22.4	59.3 / 58.9	53.8 / 46.8	53.6 / 49.3	58.4 / 44.4	51.7 / 40.6
SPG / ECCV-24 [1]	46.3 / 37.3	39.5 / 19.9	38.4 / 27.2	52.8 / 52.4	55.9 / 55.5	54.7 / 43.0	58.2 / 57.8	49.4 / 41.9
MMRL / CVPR-25 [16]	37.4 / 24.7	37.5 / 18.0	26.7 / 17.4	53.1 / 46.6	56.0 / 49.1	54.4 / 52.5	57.1 / 26.4	46.0 / 33.5
LDC / CVPR-25 [25]	51.2 / 36.0	43.8 / 27.6	41.2 / 34.1	49.4 / 47.6	53.8 / 44.8	54.2 / 33.6	57.1 / 30.1	50.1 / 36.3
OmniVaT / 我们的方法	56.7 / 50.8	52.3 / 38.1	48.7 / 46.7	61.3 / 60.0	58.7 / 60.4	56.1 / 59.2	59.3 / 60.5	56.2 / 53.7
Δ LDC	5.5 / 14.8	8.5 / 10.5	7.5 / 12.6	11.9 / 12.4	4.9 / 15.6	1.9 / 25.6	2.2 / 30.4	6.1 / 17.4

表12 在 OF Real → X 任务上的最先进方法比较。“OF Real → X”表示对其他未见目标域的评估，不包括已见源域 OF Real。 Δ LDC 表示所提出的 OmniVaT 相较于 LDC 的提升。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ TAG 准确率 / F1*	→ OF 2.0 A 准确率 / F1*	→ OF 2.0 B 准确率 / F1*	→ TQ-DIGIT 准确率 / F1*	→ TQ-GelMini 准确率 / F1*	→ TQ-DuraGel 准确率 / F1*	→ TQ-T3D 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11] 使用 CLIP 的预训练权重 [31]								
CLIP / ICML-21 [31]	51.4 / 49.8	51.6 / 37.5	48.1 / 46.0	37.3 / 32.6	47.5 / 31.8	44.4 / 31.8	56.9 / 51.5	48.2 / 40.1
PromptStyler / ICCV-23 [8]	38.7 / 34.0	45.1 / 34.3	39.6 / 32.0	47.0 / 26.8	42.1 / 25.4	42.0 / 24.1	41.8 / 34.4	42.3 / 30.1
SPG / ECCV-24 [1]	54.7 / 39.9	49.8 / 34.9	33.4 / 26.6	47.1 / 41.4	61.3 / 51.1	43.8 / 48.1	58.3 / 52.3	49.8 / 42.0
MMRL / CVPR-25 [16]	44.4 / 37.9	36.9 / 32.4	32.7 / 34.0	44.3 / 43.7	53.7 / 60.8	49.5 / 42.5	59.5 / 66.5	45.9 / 45.4
LDC / CVPR-25 [25]	41.5 / 39.1	52.8 / 39.3	46.7 / 40.0	42.3 / 32.9	50.9 / 33.1	57.8 / 39.7	54.7 / 39.2	49.5 / 37.6
OmniVaT / 我们的方法	67.9 / 66.1	51.8 / 34.9	47.3 / 40.2	61.6 / 64.5	62.9 / 62.6	56.3 / 58.9	60.7 / 68.1	58.4 / 56.5
Δ LDC	26.4 / 27.0	-1.0 / -4.4	0.6 / 0.2	19.3 / 31.6	12.0 / 29.5	-1.5 / 19.2	6.0 / 28.9	8.9 / 18.9

表13 在 OF 2.0 A → X 任务上与最先进方法的比较。“OF 2.0 A → X”表示对其他未见目标域（不包括已见源域OF 2.0 A）的评估。 Δ LDC 表示所提出的 OmniVaT 相较于 LDC 的改进。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ TAG 准确率 / F1*	→ 真实 准确率 / F1*	→ TQ-DIGIT 准确率 / F1*	→ TQ-GelMini 准确率 / F1*	→ TQ-DuraGel 准确率 / F1*	→ TQ-T3D 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11]使用来自 CLIP 的预训练权重 [31]							
CLIP / ICML-21 [31]	51.4 / 49.8	49.0 / 48.8	37.3 / 32.6	47.5 / 31.8	44.4 / 31.8	56.9 / 51.5	47.8 / 41.1
PromptStyler / ICCV-23 [8]	49.4 / 43.0	48.6 / 37.0	40.4 / 29.4	55.0 / 50.7	51.8 / 47.5	48.1 / 39.3	48.9 / 41.2
SPG / ECCV-24 [1]	41.3 / 36.0	40.7 / 36.4	57.8 / 46.8	60.4 / 53.3	52.1 / 42.6	42.3 / 41.7	49.1 / 42.8
MMRL / CVPR-25 [16]	39.9 / 27.6	40.9 / 23.6	43.5 / 34.1	42.8 / 39.2	46.4 / 25.4	43.8 / 30.5	42.9 / 30.1
LDC / CVPR-25 [25]	49.2 / 42.5	50.2 / 40.9	51.9 / 47.1	61.4 / 54.3	54.5 / 44.5	46.1 / 33.8	52.2 / 43.9
OmniVaT / 我们的方法	51.0 / 45.5	51.8 / 43.0	64.8 / 66.5	65.1 / 68.2	62.7 / 65.6	56.4 / 60.7	58.6 / 58.2
Δ LDC	1.8 / 3.0	1.6 / 2.1	12.9 / 19.4	3.7 / 13.9	8.2 / 21.1	10.3 / 26.9	6.4 / 14.3

表 14 OF 2.0 B 上与最先进方法的比较 → X 任务。“OF 2.0 B → X”表示对其他未见目标域的评估，不包括已见源 OF 2.0 B 领域。 Δ LDC 表示所提出的 OmniVaT 相较于 LDC 的提升。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ TAG 准确率 / F1*	→ OF Real 准确率 / F1*	→ TQ-DIGIT 准确率 / F1*	→ TQ-GelMini 准确率 / F1*	→ TQ-DuraGel 准确率 / F1*	→ TQ-T3D 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11]使用 CLIP 的预训练权重 [31]							
CLIP / ICML-21 [31]	51.4 / 49.8	49.0 / 48.8	37.3 / 32.6	47.5 / 31.8	44.4 / 31.8	56.9 / 51.5	47.8 / 41.1
PromptStyler / ICCV-23 [8]	33.8 / 27.8	35.2 / 20.9	34.7 / 23.8	32.8 / 17.3	30.8 / 13.5	60.5 / 56.7	38.0 / 26.7
SPG / ECCV-24 [1]	56.8 / 53.0	49.0 / 42.2	53.6 / 45.6	58.2 / 46.0	42.9 / 37.1	35.9 / 27.9	49.4 / 42.0
MMRL / CVPR-25 [16]	49.3 / 44.1	45.5 / 36.6	47.3 / 40.2	47.7 / 46.9	23.6 / 21.8	40.3 / 33.8	42.3 / 37.2
LDC / CVPR-25 [25]	49.2 / 47.9	49.7 / 50.6	51.3 / 41.0	52.4 / 44.2	57.4 / 43.6	53.0 / 35.6	52.2 / 43.8
OmniVaT / 我们的方法	65.0 / 62.6	57.3 / 52.4	59.5 / 60.0	64.1 / 65.1	59.2 / 54.4	62.4 / 67.2	61.2 / 60.3
Δ LDC	15.8 / 14.7	7.6 / 1.8	8.2 / 19.0	11.7 / 20.9	1.8 / 10.8	9.4 / 31.6	9.0 / 16.5

表15 在 TQ-DIGIT → X 任务上与最先进方法的比较。“TQ-DIGIT → X”表示对其他未见目标域的评估，不包括已见的源域 TQ-DIGIT。 Δ LDC 表示所提出的 OmniVaT 相较于 LDC 的提升。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ TAG 准确率 / F1*	→OF Real 准确率 / F1*	→ OF 2.0 A 准确率 / F1*	→ OF 2.0 B 准确率 / F1*	→ TQ-GelMini 准确率 / F1*	→ TQ-DuraGel 准确率 / F1*	→ TQ-T3D 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11] 采用 CLIP [31] 的预训练权重								
CLIP / ICML-21 [31]	53.0 / 40.4	46.7 / 40.6	45.8 / 26.1	41.2 / 33.7	57.3 / 34.1	37.3 / 30.7	56.9 / 51.5	48.3 / 36.7
PromptStyler / ICCV-23 [8]	52.4 / 47.7	61.3 / 50.5	52.0 / 36.8	45.4 / 38.6	58.9 / 53.3	58.6 / 46.7	51.8 / 56.8	54.3 / 47.2
SPG / ECCV-24 [1]	53.4 / 50.8	56.4 / 51.1	49.9 / 37.7	42.4 / 35.7	60.5 / 57.9	55.1 / 45.0	54.1 / 50.0	53.1 / 46.9
MMRL / CVPR-25 [16]	28.4 / 17.3	31.9 / 12.8	27.8 / 10.9	26.8 / 13.5	50.3 / 26.4	46.7 / 23.5	39.9 / 21.1	36.0 / 17.9
LDC / CVPR-25 [25]	53.4 / 51.7	56.1 / 49.6	54.9 / 25.3	44.7 / 33.3	54.9 / 38.6	55.4 / 41.9	57.0 / 32.1	53.8 / 38.9
OmniVaT / 我们的方法	55.7 / 52.2	61.8 / 51.7	57.8 / 36.6	44.3 / 35.2	57.8 / 52.8	58.2 / 63.5	58.9 / 59.0	56.4 / 50.1
Δ LDC	2.3 / 0.5	5.7 / 2.1	2.9 / 11.3	-0.4 / 1.9	2.9 / 14.2	2.8 / 21.6	1.9 / 26.9	2.6 / 11.2

表16 在 TQ-GelMini 上与最先进方法的比较 → X 任务。“TQ-GelMini → X”表示对其他未见目标域的评估，不包括已见源 TQ-GelMini 领域。 Δ LDC 表示所提出的 OmniVaT 相较于 LDC 的改进。指标为准确率（准确率，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ TAG 准确率 / F1*	→OF 真实 准确率 / F1*	→OF 2.0 A 准确率 / F1*	→OF 2.0 B 准确率 / F1*	→ TQ-DIGIT 准确率 / F1*	→ TQ-DuraGel 准确率 / F1*	→ TQ-T3D 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11] 使用来自 CLIP 的预训练权重 [31]								
CLIP / ICML-21 [31]	53.0 / 40.4	46.7 / 40.6	45.8 / 26.1	41.2 / 33.7	44.4 / 39.1	37.3 / 30.7	56.9 / 51.5	46.5 / 37.4
PromptStyler / ICCV-23 [8]	35.8 / 33.2	55.7 / 45.2	50.7 / 36.2	44.3 / 37.1	43.3 / 46.2	47.5 / 35.0	52.0 / 45.6	47.0 / 39.8
SPG / ECCV-24 [1]	53.7 / 50.0	56.4 / 48.7	51.2 / 38.0	40.5 / 34.9	60.1 / 57.9	54.5 / 51.5	60.4 / 59.5	53.8 / 48.6
MMRL / CVPR-25 [16]	32.1 / 22.7	32.1 / 12.7	27.8 / 11.0	24.0 / 12.3	11.6 / 8.4	30.7 / 19.7	33.0 / 19.4	27.3 / 15.2
LDC / CVPR-25 [25]	52.9 / 48.5	44.7 / 28.2	47.1 / 32.8	36.4 / 28.0	43.1 / 33.3	41.2 / 33.9	47.6 / 42.2	44.7 / 35.3
OmniVaT / 我们的方法	58.0 / 60.0	61.1 / 49.8	53.6 / 39.8	45.1 / 33.9	64.9 / 69.6	56.3 / 60.5	62.0 / 61.0	57.3 / 53.5
Δ LDC	5.1 / 11.5	16.4 / 21.6	6.5 / 7.0	8.7 / 5.9	21.8 / 36.3	15.1 / 26.6	14.4 / 18.8	12.6 / 18.2

表 17 与 TQ-DuraGel 上最先进方法的比较 → X 任务。“TQ-DuraGel → X”表示对其他未见目标域（不包括已见源域 TQ-DuraGel）的评估。 Δ LDC 表示所提出的 OmniVaT 相较于 LDC 的改进。指标为准确率（ACC，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ TAG 准确率 / F1*	→ OF Real 准确率 / F1*	→ OF 2.0 A 准确率 / F1*	→OF 2.0 B 准确率 / F1*	→ TQ-DIGIT 准确率 / F1*	→ TQ-GelMini 准确率 / F1*	→ TQ-T3D 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11] 使用来自 CLIP 的预训练权重 [31]								
CLIP / ICML-21 [31]	53.0 / 40.4	46.7 / 40.6	45.8 / 26.1	41.2 / 33.7	44.4 / 39.1	57.3 / 34.1	37.3 / 30.7	46.5 / 35.0
PromptStyler / ICCV-23 [8]	46.9 / 46.0	58.4 / 47.7	47.9 / 36.6	41.8 / 38.2	54.3 / 59.1	53.4 / 47.5	59.6 / 61.9	51.8 / 48.1
SPG / ECCV-24 [1]	54.0 / 53.1	50.6 / 43.2	38.8 / 34.9	40.7 / 34.6	65.2 / 63.8	56.2 / 44.1	55.5 / 50.2	51.6 / 46.3
MMRL / CVPR-25 [16]	25.0 / 10.6	31.9 / 12.1	27.8 / 11.0	25.0 / 10.0	29.0 / 12.0	36.6 / 26.2	32.1 / 19.0	29.6 / 14.4
LDC / CVPR-25 [25]	53.3 / 51.7	38.4 / 29.8	47.3 / 26.1	43.8 / 25.2	52.6 / 48.4	54.4 / 50.5	54.7 / 36.6	49.2 / 38.3
OmniVaT / 我们的方法	57.2 / 57.0	58.3 / 49.8	47.8 / 39.8	49.3 / 36.9	64.1 / 69.6	56.0 / 49.3	61.2 / 60.0	56.3 / 51.8
Δ LDC	3.9 / 5.3	19.9 / 20.0	0.5 / 13.7	5.5 / 11.7	11.5 / 21.2	1.6 / -1.2	6.5 / 23.4	7.1 / 13.5

表18 在 TQ-T3D → X 任务上与最先进方法的比较。“TQ-T3D → X”表示对除已见源 TQ-T3D 领域之外的其他未见目标域的评估。 Δ LDC 表示所提出的 OmniVaT 相对于 LDC 的提升。指标为准确率（准确率，%）和宏平均 F1（F1*，%）。

方法 / 发表 venue	→ TAG 准确率 / F1*	→ OF Real 准确率 / F1*	→OF 2.0 A 准确率 / F1*	→ OF 2.0 B 准确率 / F1*	→ TQ-DIGIT 准确率 / F1*	→ TQ-GelMini 准确率 / F1*	→ TQ-DuraGel 准确率 / F1*	平均值 准确率 / F1*
ViT-B / 16 [11]使用 CLIP 的预训练权重 [31]								
CLIP / ICML-21 [31]	53.0 / 40.4	46.7 / 40.6	45.8 / 26.1	41.2 / 33.7	24.2 / 20.4	29.5 / 27.4	57.5 / 27.3	42.6 / 30.8
PromptStyler / ICCV-23 [8]	52.9 / 48.8	50.7 / 37.1	43.3 / 21.4	44.7 / 26.1	51.0 / 26.0	50.8 / 22.0	52.6 / 35.2	49.4 / 30.9
SPG / ECCV-24 [1]	54.3 / 53.8	46.4 / 42.2	45.4 / 27.8	36.2 / 32.9	59.4 / 48.6	56.8 / 50.1	54.4 / 50.9	50.4 / 43.8
MMRL / CVPR-25 [16]	38.3 / 33.6	56.6 / 46.4	47.5 / 31.1	40.9 / 25.9	53.7 / 37.5	53.9 / 42.9	50.9 / 26.2	48.8 / 34.8
LDC / CVPR-25 [25]	56.1 / 43.2	55.4 / 36.8	46.5 / 21.3	39.2 / 22.5	54.9 / 35.6	58.2 / 42.3	60.2 / 42.4	52.9 / 34.9
OmniVaT / 我们的方法	56.5 / 59.1	58.6 / 49.3	47.1 / 30.2	42.2 / 38.1	57.9 / 50.9	59.2 / 54.7	57.1 / 53.3	54.1 / 47.9
Δ LDC	0.4 / 15.9	3.2 / 12.5	0.6 / 8.9	3.0 / 15.6	3.0 / 15.3	1.0 / 12.4	-3.1 / 10.9	1.2 / 13.0