人在情境中：通过情境学习统一跨领域 3D 人体运动建模

Mengyuan Liu, Xinshun Wang †, Zhongbin Fang †, Deheng Ye, Xia Li, Tao Tang, Songtao Wu, Xiangtai Li, Ming-Hsuan Yang

Abstract—本文旨在跨领域建模 3D 人体运动，其中一个模型预计能够处理多种模态、任务和数据集。现有的跨领域模型通常依赖于领域特定的组件和多阶段训练，这限制了它们的实用性和可扩展性。为克服这些挑战，我们提出了一种新的设置，通过一个过程来训练一个统一的跨领域模型，消除了对领域特定组件和多阶段训练的需求。我们首先介绍了 Pose-in-Context（PiC），它利用上下文学习来创建一个以姿态为中心的跨领域模型。虽然 PiC 在多种基于姿态的任务和数据集上具有广泛的适应性，但它在处理模态多样性、提示策略和上下文依赖处理方面遇到了困难。因此，我们提出了 Human-in-Context（HiC），这是 PiC 的扩展，扩大了跨模态、任务和数据集的泛化能力。HiC 在一个统一的框架中结合了姿态和网格表示，扩大了任务覆盖范围，并纳入了更大规模的数据集。此外，HiC 引入了一种最大-最小相似性提示采样策略，以增强在不同领域中的泛化能力，并采用了具有双分支上下文注入的网络架构，以改进对上下文依赖的处理。大量实验结果表明，HiC 在泛化能力、数据规模和性能方面都比 PiC 表现更好，涵盖了广泛的领域。这些结果展示了 HiC 在构建具有更高灵活性和可扩展性的统一跨领域 3D 人体运动模型的潜力。源代码和模型可在 https://github.com/BradleyWang0416/Human-in-Context 获取。

Index Terms—In-context learning, human motion modeling, cross-domain, unified modeling

1 引言

人体运动建模是计算机视觉中的核心主题，跨域 3D 人体运动建模旨在开发一种能够处理多个领域的模型，包括不同的任务、模态和数据集。为此，姿势 [?], [?] 和网格 [?] 是两个被广泛采用的人体运动表示形式，因为与基于 RGB 的表示 [?], [?] 相比，它们提供了更高的效率、紧凑性和更丰富的信息。利用姿势和网格表示，跨域 3D 人体运动建模已在各种应用中找到应用，例如自主驾驶的运动预测 [?], [?] 、人机合作的姿势估计 [?], [?], [?] 和虚拟现实的网格恢复 [?] 。

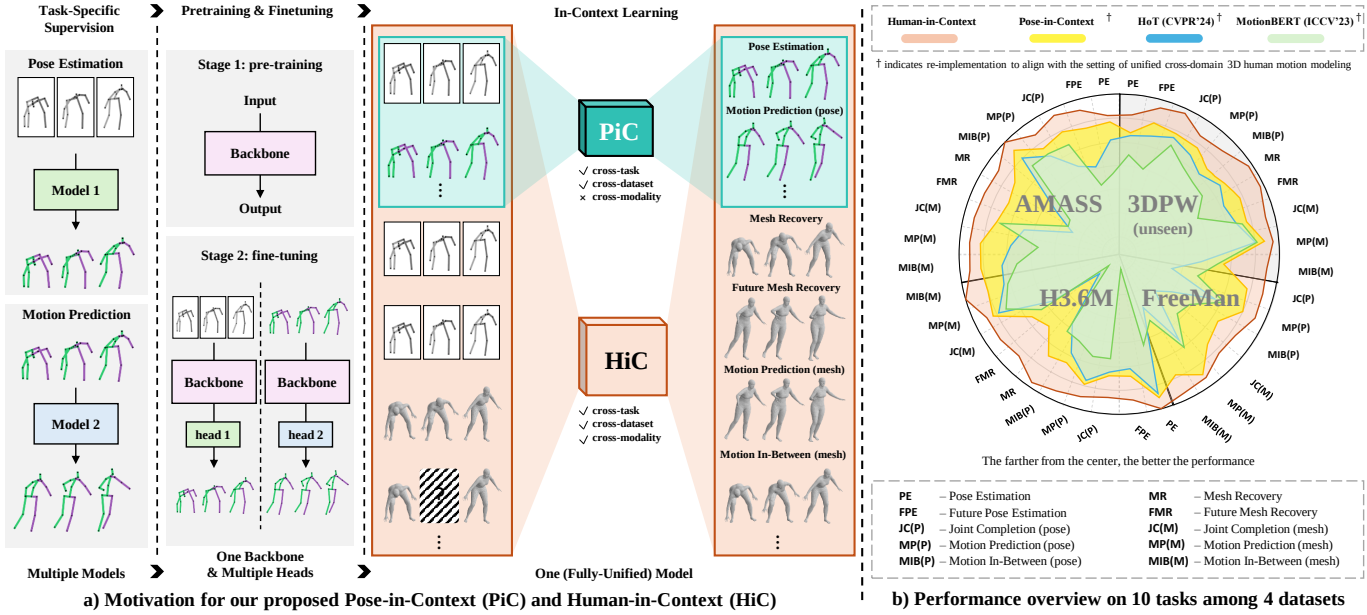
跨领域模型在计算机视觉、机器人技术和自然语言处理中已经被广泛探索，利用了多样的骨干网络 [?], [?], [?] 和训练范式 [?], [?] 。然而，将跨领域建模应用于 3D 人体动作面临更大的挑战，这主要是由于 3D 运动数据的多维和时空复杂性 [?] 。因此，现有的跨领域模型 [?], [?], [?], [?], [?] 存在两个主要限制。首先，其适用性通常局限于狭隘的范围，比如在几个数据集上执行相同任务 [?], [?] 或在单一模态数据内处理类似任务 [?], [?] 。其次，它们在很大程度上依赖于额外的特定领域模型头 [?], [?] 并需要复杂的多阶段训练 [?], [?] ，限制了它们的通用性和可扩展性。

为了克服这些限制，我们引入了一种新的设置，使得可以在单一过程中训练一个统一的跨域模型，从而消除对领域特定组件和复杂多阶段训练的需求。受启发于自然语言处理中的上下文学习 [?], [?] ，这种学习方式允许模型在不进行显式微调或重新训练的情况下执行多项任务，我们发现这种范式与我们提出的设置非常契合。虽然上下文学习已经扩展到基于图像的任务 [?], [?] 和基于点云的任务 [?] ，但据我们所知，它在 3D 人体运动建模中的应用尚未得到探索。

我们提出了 PiC，这是一种首次将上下文学习融入 3D 人体运动建模的方法，以促进以姿势为中心的跨领域模型。PiC 在各种基于姿势的任务和数据集上实现了泛化，并表现出竞争力的性能。然而，PiC 存在三个主要限制：1）PiC 专注于基于姿势的任务和数据集的范围，不能泛化到不同的模态；2）PiC 使用基于随机选择的提示策略，选择的提示与查询无关，这可能导致查询与选定的提示之间存在较大差距；以及 3）PiC 采用的是基于注意力的网络架构，并未利用除全局上下文外的上下文依赖。

为了克服 PiC 的局限性，我们开发了 Human-in-Context，这是一个 3D 人类运动建模中的跨域模型，实现了在单一统一框架内的跨模态、跨任务和跨数据集的泛化。与仅处理基于姿态任务的 PiC 不同，HiC 可以通过一次训练来处理基于姿态和网格的任务。HiC 在三个方面扩展了 PiC（见图 1 ）。首先，HiC 扩展了模型的泛化能力，以涵盖更广泛的领域，包括姿态和网格模态。通过在一个统一的公式中表示基于姿态和网格的人类运动，HiC 支持在一个上下文框架内的跨模态学习。关于任务覆盖，对于 PiC 中的每个基于姿态的任务，HiC 包含一个相应的基于网格的变体，有效地使支持的任务数量翻倍。其次，HiC 采用最大-最小相似性提示采样策略来增强泛化能力。它不是随机选择，而是从训练集中采样具有代表性的锚点，为每个查询检索最接近的匹配锚点，以确保上下文对齐。该方法动态地将硬锚点（固定代表）与软锚点（可调整的细化）配对，提高了在不同领域（既包括分布内也包括分布外）的泛化能力。第三，HiC 引入了一个新的网络 X-Fusion Net，具有双分支架构，能够实现跨提示和查询的上下文集成。每个分支由一系列 X-Fusion 块组成，以处理多个层次的上下文信息。每个块执行两个操作：多层次上下文聚合和跨层次上下文更新。在聚合步骤中，使用状态空间模型、自注意力和图卷积来处理特征，以捕获不同视图、范围和特征空间的依赖关系。在更新步骤中，网络通过估计其对目标任务的相对重要性来调整不同层次的特征表示。这些操作支持跨多样化

- *Mengyuan Liu, Xinshun Wang, and Tao Tang are with the National Key Laboratory of General Artificial Intelligence, Peking University, Shenzhen Graduate School.*
- *Zhongbin Fang and Deheng Ye are with Tencent, China.*
- *Xia Li is with the Department of Information Technology and Electrical Engineering, ETH Zurich.*
- *Xiangtai Li is with S-Lab, Nanyang Technological University, Singapore.*
- *Songtao Wu is with Sony R & D Center, China.*
- *Ming-Hsuan Yang is with the University of California at Merced, US.*
- *Xinshun Wang and Zhongbin Fang are co-corresponding authors.*



a) Motivation for our proposed Pose-in-Context (PiC) and Human-in-Context (HiC)　　b) Performance overview on 10 tasks among 4 datasets

Fig. 1: 我们提出的姿态-情境（Pose-in-Context）和人类-情境（Human-in-Context）的动机（左）及性能概述（右）。与以前需要多个模型/头和多个训练阶段的方法相比，PiC 和 HiC 通过一次性训练的全统一模型统一了各种以人为中心的任务和数据集。此外，HiC 通过在功能性和技术性上引入设计，将 PiC 扩展为一个更具包容性、可扩展性和普遍适用的跨领域模型。

运动领域的上下文感知表示学习。

大量结果表明，PiC 和 HiC 在特定领域和跨领域方法中表现出色。图 1 展示了整体框架，其中 PiC [1] 首次在我们 2024 年 CVPR 会议论文 [?] 中提出。此项工作 HiC 在多个方面扩展了 PiC：

- HiC 是一种用于 3D 人体运动建模的上下文内跨领域模型，它同时具备跨模态、跨任务和跨数据集的泛化能力。
- HiC 包含更大范围的领域，包括两种模态、十个任务和四个数据集，同时将数据规模扩展到 ∼ 21× 。
- HiC 提出了一种提示策略，叫做最大-最小相似度提示抽样，以应对跨高度多样化领域进行广义化的挑战。
- HiC 提出了一种网络 X-Fusion Net，以通过多级上下文聚合和跨级上下文更新有效地捕获复杂的上下文依赖关系。
- 大量实验结果表明，HiC 在域内和域外的泛化方面始终表现出卓越的性能，平均比 PiC 高 9.9 %，比其他特定领域/跨领域模型高 21.8 %。

2　相关工作

3D 人体动作建模。3D 人体动作建模涵盖了广泛的领域，涉及不同的任务和模式。一些广泛认可的任务包括动作预测、姿势估计、网格恢复和动作插值等等。关于模式，姿势和网格是表示 3D 人体动作的两种最流行的方式。与 RGB 图像/视频等 2D 表示相比，3D 表示提供了更高效、紧凑和信息丰富的人体动作建模方式。关于 3D 人体动作数据集，Human3.6M 是 3D 人体动作建模中使用得最广泛的数据集之一，提供了从大量不同活动的主体中捕获的高质量 3D 关节位置。3DPW 旨在用于户外环境，展示了现实世界的动作和复杂的相机视角。AMASS 提供了来自多个来源（如 CMU-Mocap）的动作捕捉数据集，并使用 SMPL 参数化。FreeMan 是一个更新的数据集，专注于基于姿势和基于网格的人体动作，旨在提高动作数据在 3D 人体动作建模中各种任务的现实性和多样性。NTU RGB+D 和 NTU RGB+D 120 主要用于分类任务，而不是输出动作序列的任务。3D 人体动作建模中的不同领域可能存在显著的差距，使得现有的工作在处理多个领域时依赖于特定领域的设计。相比之下，我们引入了 PiC 和 HiC，实现了跨领域的 3D 人体动作建模，而无需任何特定领域的设计。

人为中心的领域专用模型。早期的工作通过设计领域专用模型来处理各个领域 [?], [?] 。众多方法设计领域专用模型，包括使用 Human3.6M [?], [?], [?] 或 AMASS 数据集 [?] 的 3D 姿势数据进行运动预测，使用 3DPW 数据集 [?], [?] 的网格恢复等。近年来，已经提出了各种骨干网络用于领域专用模型，包括 MLP [?] 、卷积 [?], [?], [?], [?] 、递归 [?], [?], [?] 、图形 [?], [?], [?], [?], [?] 神经网络和 transformers [?], [?], [?], [?], [?] ，这些网络在不同数据集如 Human3.6M 和 3DPW 上单独训练。此外，Mamba [?], [?] 已被用于 3D 人类运动建模 [?], [?] 。与领域专用模型需要部署独立模型并在各个领域单独训练相反，我们利用上下文学习来建立一种统一的方法，更高效和有效地解决多个领域的问题。以人为中心的跨域模型。与特定领域的模型不同，最近的方法通过设计跨域模型来统一不同领域 [?], [?], [?], [?], [?] 。由于人体运动的多维和时空信号，现有的跨域模型局限于少数领域，同时仍需要额外的领域特定模型组件或多阶段训练。例如，在 AMASS 上训练并在 3DPW 上测试的运动预测模型 [?], [?] ，或者在人体 3.6M 上训练并在 3DPW 上测试的网格恢复模型 [?] 。跨域建模还涉及在多个任务之间进行泛化，例如，姿势和形状估计 [?], [?] 、姿势估计和动作识别 [?], [?] ，以及运动预测和动作识别 [?] 。最近的跨域模型可以在单一模态数据上跨越更多任务进行泛化 [?], [?] 。随着新骨干网络 [?] 和训练范式 [?], [?] 的出现，跨域模型展现了令人鼓舞的结果。UniHCP [?] 通过使用共享骨干网配合任务感知头来切换任务，并在任务特定监督下进行训练，从而执行若干人类感知任务。在 [?] 中，MotionBERT 预训练一个骨干网以学习统一表示，并分别微调不同的任务头以适应不同任务。另一方面，UPS [?] 可以通过使用任务特定的嵌入来激活模型中的不同块并以类似语言的方式回归生成标记，来执行三个不同的任务。LMM [?] 通过在统一的预训练表示之上微调数据集特定层，并将任务特定条件作为不同任务所需的额外输入，从而在多个任务之间进行泛化。然而，无效的骨干网络和不可避免的领域特定

1. 在我们的先前工作中，“环境中的姿态” 被称为 “环境中的骨架”。为了在本期期刊的跨领域上下文中术语的一致性，我们在整篇文章中使用 “姿态” 而不是 “骨架”。

TABLE 1: HiC 和 PiC 的比较。

Name	Data Scale	Domain Scope		
		modality	task	dataset
Pose-in-Context	0.18M	pose	5	3
Human-in-Context	3.83M	pose & mesh	10	4

模型头或微调的参与是现有跨领域模型的两个关键限制。为了解决这些限制，我们提出了一种统一的跨领域模型，可以通过一次训练解决所有任务。据我们所知，我们是首次为各种三维人类任务设计模型。

领域内学习。领域内学习 [?], [?], [?] 提供了一种基于上下文执行多个任务的新方法，不需要特定任务的显式再训练或微调，最近在视觉和自然语言处理 [?], [?], [?], [?] 中得到了应用。上下文以提示 [?], [?], [?] 的形式呈现，提示由输入和目标输出对构成，作为任务期望实现的示例 [?], [?] 。领域内学习显著依赖于两个关键设计元素：提示策略 [?] 和网络架构 [?] 。有效的提示策略确保提示被设计和使用，以提供足够的上下文供模型学习。同时，设计良好的网络架构确保模型有效处理提示，从上下文中提取隐藏的任务，然后在查询上执行所需的任务。虽然领域内学习与统一跨域 3D 人体动作建模设置相符，但其在人类动作上的应用尚未被探索。在这项工作中，我们将领域内学习引入到 3D 人体运动建模中，实现了具有强大可扩展性和泛化能力的统一跨域模型。

3 问题表述

在这一部分中，我们介绍了 Human-in-Context 所需的基本概念，包括跨域设置和上下文学习。

3.1 人类情境设置

统一不同领域是在构建跨领域模型时一个关键的初步步骤。在这项工作中，领域指的是特定任务、所涉及的模态及其应用的数据集的组合。在 < 人-情境 > 的范围内所涉及的领域在表格 2 中有详细显示。

跨模态设置。为了促进人类背景中的跨模态泛化，我们首先在统一的表述中重新解释基于姿势和基于网格的表示，使它们能够在跨模态环境中共同应用。姿势和网格是人类表示的两种不同模态，现有的 3D 人体运动建模工作通常没有在统一的框架中对它们进行联合研究。相比于不同任务或数据集之间的差距，不同模态之间的领域差距通常表现出更显著的挑战。

1) 基于姿势的人体运动被表示为一系列姿势，其中一个姿势由在 2D 或 3D 空间中用位置坐标表示的多个关节组成。设一系列姿势为 $\boldsymbol{X}_{1:F}^{\text{pose}} = [\boldsymbol{x}_1^{\text{pose}}, \boldsymbol{x}_2^{\text{pose}}, \cdots, \boldsymbol{x}_F^{\text{pose}}] \in \mathbb{R}^{F\times N\times C}$ ，其中 $\boldsymbol{x}_f^{\text{pose}}$ 为第 f 帧的姿势，包含 N 个关节。对于 2D 姿势的情况 $\boldsymbol{X}^{\text{2D\_pose}}$ ，关节由 $C=2$ 坐标 (x, y) 表示，对于 3D 姿势的情况 $\boldsymbol{X}^{\text{3D\_pose}}$ ，关节由 $C=3$ 坐标 (x, y, z) 表示。为了在与网格格式统一之前获得一个统一的姿势格式，我们通过在 z 轴上附加一个全零切片将 2D 姿势扩展为 3D。2) 基于网格的人体运动表示为由顶点和面组成的 3D 网格序列。这种基于网格的方法不仅捕捉到整体身体关节旋转配置，还捕捉到身体表面的细微几何细节。网格表示的一个常用参数化是 Skinned Multi-Person Linear (SMPL) [?] 模型，其输入为关节旋转参数 $\boldsymbol{\theta} \in \mathbb{R}^{3J}$ 和形状参数 $\boldsymbol{\beta} \in \mathbb{R}^{S}$ ，然后输出一个三角网格 $\mathcal{V} \in \mathbb{R}^{6980\times 3}$ 。SMPL 关节旋转向量 $\boldsymbol{\theta}$ 由 J 个关节组成，每个关节由一个三维轴角旋转向量 $(\theta_x, \theta_y, \theta_z)$ 表示，其中 $(\theta_x, \theta_y, \theta_z)$ 是按旋转弧度缩放的旋转轴。SMPL 形状向量 $\boldsymbol{\beta}$ 编码了个体身体形状的变化，例如身高和体重。令一系列 SMPL 关节旋转向量为 $[\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \cdots, \boldsymbol{\theta}_F] \in \mathbb{R}^{F\times 3J}$ ，其中 $\boldsymbol{\theta}_f$ 是 f 帧的 SMPL 关节旋转向量。为了使 $\boldsymbol{\theta}_f$ 的维度与 $\boldsymbol{x}_f^{\text{pose}}$ 的维度对齐，我们重新组织 $\boldsymbol{\theta}_f$ 的元素以获得 $\boldsymbol{X}_{1:F}^{\text{mesh}} = [\boldsymbol{x}_1^{\text{mesh}}, \boldsymbol{x}_2^{\text{mesh}}, \cdots, \boldsymbol{x}_F^{\text{mesh}}] \in \mathbb{R}^{F\times J\times 3}$ 。为了统一姿态和网格表示之间的跨模态数据，我们首先找到最大关节数 $max(N, J)$ ，其中 N 和 J 是 $\boldsymbol{X}_{1:F}^{\text{pose}}$ 和 $\boldsymbol{X}_{1:F}^{\text{mesh}}$ 中的关节数。然后，我们为关节数少于 $max(N, J)$ 的数据添加虚拟关节，其元素均为零。此外，对于基于姿态表示的 $\boldsymbol{X}_{1:F}^{\text{pose}}$ ，我们分配虚拟形状参数 $\boldsymbol{\beta} = \boldsymbol{0}$ ，这仅用于对齐姿态和网格表示的数据格式，不会影响训练或评估。无损推广地，假设 $J = max(N, J)$ ，统一跨模态人体运动表示定义为 $\boldsymbol{X}_{1:F} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_F] \in \mathbb{R}^{F\times J\times C}$ ，其中元素可以是位置坐标或轴角旋转向量。

跨任务设置。通过在不同的模态和/或不同的时间窗口上指定输入和目标，可以从任意运动序列 $\boldsymbol{X}_{1:2F} \in \{\boldsymbol{X}_{1:2F}^{\text{2D\_pose}}, \boldsymbol{X}_{1:2F}^{\text{3D\_pose}}, \boldsymbol{X}_{1:2F}^{\text{mesh}}\}$ 中派生出 3D 人类运动任务。由于 $\boldsymbol{X}_{1:2F}^{\text{2D\_pose}}$ 、$\boldsymbol{X}_{1:2F}^{\text{3D\_pose}}$ 和 $\boldsymbol{X}_{1:2F}^{\text{mesh}}$ 从不同视角描述了同一人类运动片段，我们可以为不同任务构建一个统一的表述。具体而言，我们探索了 10 项任务，涉及不同的模态和时间窗口，其输入和目标如表 2 所示。

假设前 F 帧代表“当前（或历史）”，而后 F 帧代表“未来”。姿势估计和网格恢复任务专注于在另一种模式下估计当前运动。诸如运动预测等任务旨在预测与历史运动相同模式下的未来运动。另一方面，未来姿势估计和网格恢复任务更具挑战性，因为它们需要在一种新的模式下预测未来运动。运动补充和联合完成任务旨在恢复运动中缺失的部分。

为了制定这些任务，我们对输入片段应用一个二进制掩码 $\boldsymbol{M}^{\text{time}} \in \mathbb{R}^{F}$ 或 $\boldsymbol{M}^{\text{joint}} \in \mathbb{R}^{J}$ ，用于确定哪些特定帧或关节缺失。掩码是为每个运动片段随机生成的，然后填充以使其具有与运动片段相同的形状。为了避免改变任务的性质，首帧和末帧以及根关节从不被掩盖。无论任务的定义如何，它们的输入和输出都是相同的形状，并且可以适配到一个统一的框架中。跨数据集设置。训练统一的跨领域模型需要从大型数据集中获取足够的数据。然而，这些数据集中并非都同时提供姿态和网格数据。因此，我们遵循现有方法的标准实践，额外处理这些数据集以获得跨姿态和网格的数据。像 3DPW [?] 和 FreeMan [?] 这样的数据集本身就提供了姿态和网格数据，因此不需要额外处理。对于提供 2D 和 3D 姿态数据而没有网格数据的数据集，如 Human3.6M (H3.6M) [?] ，地面真实 SMPL 数据是通过将 MoSh [?] 应用到稀疏的 3D 运动捕捉标记数据来生成的，如现有工作中所做的一样 [?], [?] 。注意，这些数据集中的 2D 和 3D 姿态，即使对应于同一运动剪辑，通常也不共享相同的 (x, y) 坐标值。2D 姿态对应于 2D 图像空间，因为它们表示的是投射到图像像素坐标上的人体关节，而 3D 姿态则定义在 3D 世界坐标空间中。对于提供了 SMPL 网格数据但不提供 2D/3D 姿态数据的数据集，例如 AMASS [?] ，我们使用一个由 [?] 预训练的特定数据集关节回归器，将 SMPL 顶点回归到 3D 姿态关节，与常见做法一致 [?], [?], [?], [?] 。然后，我们通过正交投影将 3D 姿态投射到 xy 平面以推导出 2D 姿态，遵循标准方法 [?] 。

3.2 上下文学习

我们使用形式 $[< input >, < target >]^{D}$ 来表示表 2 中对应于任何特定领域 D 的上下文。通常，情境学习框架使用一个模型 $\mathcal{M}(\cdot)$ 将提示（P ）和查询（Q ）作为输入，并生成所需的输出：

$$\mathcal{M}\left(\begin{array}{l}[< input >_P, < target >_P]^D, \\ < input >_Q\end{array}\right) \rightarrow < output >_Q, \tag{1}$$

其中提示和查询从同一领域 D 采样。提示输入和提示目标共同为模型提供了理解上下文所暗示的任务并对查询输入执行相同任务所需的必要上下文。情境学习依赖于两个关键设计元素：提示策略和网络架构。提示策略定义了如何从领

TABLE 2: 人性化情境设置。在这项工作的范围内，一个领域由三个方面来解释，包括任务的输入和输出、所涉及的模态以及适用的数据集。

Domain		Task		Modality			Dataset			
		Input	Output	pose (2D)	pose (3D)	mesh	H3.6M	AMASS	FreeMan	3DPW
1. Pose Estimation	PE	$\boldsymbol{X}^{\text{2D\_pose}}_{1:F}$	$\boldsymbol{X}^{\text{3D\_pose}}_{1:F}$	✓	✓		✓	✓		✓
2. Future Pose Estimation	FPE	$\boldsymbol{X}^{\text{2D\_pose}}_{1:F}$	$\boldsymbol{X}^{\text{3D\_pose}}_{1:F}$	✓	✓		✓	✓		✓
3. Mesh Recovery	MR	$\boldsymbol{X}^{\text{2D\_pose}}_{1:F}$	$\{\boldsymbol{X}^{\text{mesh}}_{1:F}, \boldsymbol{\beta}\}$	✓		✓	✓	✓		✓
4. Future Mesh Recovery	FMR	$\boldsymbol{X}^{\text{2D\_pose}}_{1:F}$	$\{\boldsymbol{X}^{\text{mesh}}_{F+1:2F}, \boldsymbol{\beta}\}$	✓		✓	✓	✓		✓
5. Motion Prediction (pose)	MP (P)	$\boldsymbol{X}^{\text{3D\_pose}}_{1:F}$	$\boldsymbol{X}^{\text{3D\_pose}}_{F+1:2F}$		✓		✓	✓	✓	✓
6. Motion In-Between (pose)	MIB (P)	$\boldsymbol{X}^{\text{3D\_pose}}_{1:F} \odot \boldsymbol{M}^{\text{time}}$	$\boldsymbol{X}^{\text{3D\_pose}}_{1:F}$		✓		✓	✓	✓	✓
7. Joint Completion (pose)	JC (P)	$\boldsymbol{X}^{\text{3D\_pose}}_{1:F} \odot \boldsymbol{M}^{\text{joint}}$	$\boldsymbol{X}^{\text{3D\_pose}}_{1:F}$		✓		✓	✓	✓	✓
8. Motion Prediction (mesh)	MP (M)	$\boldsymbol{X}^{\text{mesh}}_{1:F}$	$\{\boldsymbol{X}^{\text{mesh}}_{F+1:2F}, \boldsymbol{\beta}\}$			✓	✓	✓	✓	✓
9. Motion In-Between (mesh)	MIB (M)	$\boldsymbol{X}^{\text{mesh}}_{1:F} \odot \boldsymbol{M}^{\text{time}}$	$\{\boldsymbol{X}^{\text{mesh}}_{1:F}, \boldsymbol{\beta}\}$			✓	✓	✓	✓	✓
10. Joint Completion (mesh)	JC (M)	$\boldsymbol{X}^{\text{mesh}}_{1:F} \odot \boldsymbol{M}^{\text{joint}}$	$\{\boldsymbol{X}^{\text{mesh}}_{1:F}, \boldsymbol{\beta}\}$			✓	✓	✓	✓	✓

Some conventional task names, such as motion prediction, may lack specificity in the modality; in such cases, the modality is indicated in parentheses.

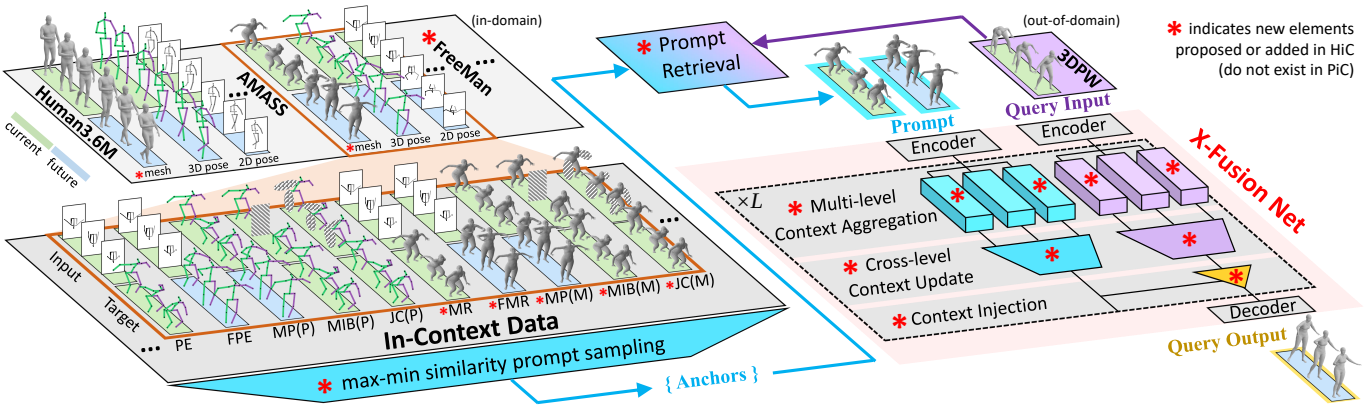


Fig. 2: 所提出的人类情境（HiC）流程，并重点介绍其与 PiC 的区别。在数据端，我们从包含以多种形式表示的运动片段的各种数据集中提取情境数据集，这些数据集包含 10 个不同任务的输入-目标对。通过所提出的最大-最小相似性提示采样，从情境数据集中抽取一组锚点。在每次推理过程中，基于查询输入（在此示例中为来自 3DPW 数据集的域外运动序列）从锚点中检索最合适的提示。在网络端，查询输入和提示通过我们提出的 X-Fusion Net 进行处理，在其中我们应用了多级情境聚合、跨级别情境更新、情境注入和其他模块。

域 D 获取提示，而网络架构定义了模型 $\mathcal{M}(\cdot)$ 如何处理提示和查询。一个有效的提示策略确保提示的设计和使用方式能够为模型提供足够的上下文供其学习，而精心设计的网络架构确保模型以一种能够有效从提示所提供的上下文中提取隐含任务并准确在查询上执行所需任务的方式处理提示和查询。在这项工作中，我们提出了两种方法，Pose-in-Context 和 Human-in-Context，以有效实施情境学习，实现统一的跨域 3D 人体运动建模，这将在接下来的两节中分别介绍。

为了符号简洁，在以下部分中将省略表示帧索引的下标，例如 $1:F$ ，因为帧索引可以根据不同域推断出来。

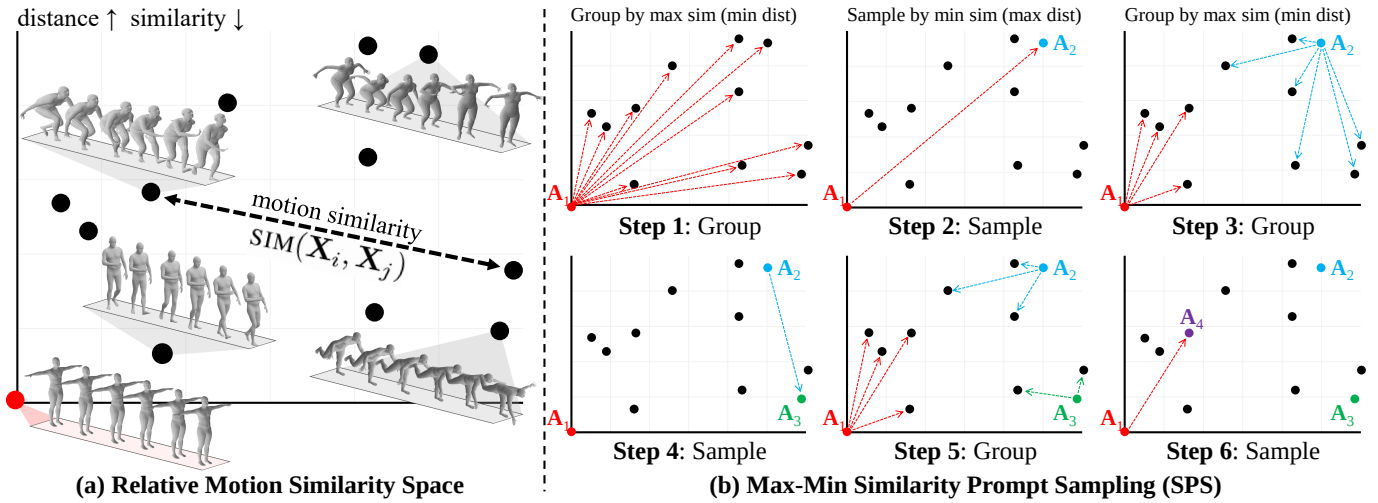
4 人类环境

我们提出了 Human-in-Context，这是一种跨领域模型，能够同时具备跨模态、跨任务和跨数据集的泛化能力，如图 2 所示。Human-in-Context 在三个方面扩展和改进了 Pose-in-Context，包括解锁跨模态泛化能力、扩大数据集和任务的规模，并改善在所有领域的性能。通过第 3.1 节的解释，将跨模态设置进行公式化，从而启用了跨模态的泛化能力。此外，Human-in-Context 引入了一种提示策略和网络架构，分别在第 4.2 节和 **??** 节中介绍，这两者共同促进了在更大规模数据集和任务下，无论在姿态还是网格方面都能提高表现。

4.1 上下文中的姿势

Pose-in-Context 旨在跨越各种基于姿态的任务和数据集进行泛化，实施了一种基于随机选择的提示策略和基于注意力的网络架构。由于 Pose-in-Context 已经在我们的会议论文中进行了详细说明 [**?**]，我们在本小节中对其关键设计进行了简要回顾。

在 PiC 中，提示策略基于任务引导提示 (TGP) 和任务统一提示 (TUP) 的结合。与第 3 节中介绍的符号一致，人类运动序列被表示为 $\boldsymbol{X} \in \mathbb{R}^{F\times J\times C}$ ，由 F 帧、J 关节和 C 通道组成。TGP 本质上是随机选择的硬提示，可以写为 $[\boldsymbol{X}^{\text{in}}_i, \boldsymbol{X}^{\text{gt}}_i]^D$



(a) Relative Motion Similarity Space **(b) Max-Min Similarity Prompt Sampling (SPS)**

Fig. 3: Human-in-Context 中提示策略的说明。首先，在训练范围内，基于它们与标准体态配置及其他动作序列的相对相似性，构造一个相对运动相似性空间。接下来，最大-最小相似性提示抽样进行迭代操作以采样一组锚点，这基本上涉及每次迭代的两个步骤：1）按最大相似性对未采样动作序列进行分组，2）按最小相似性采样新的锚点。

，与方程 (1) 一致，其中 D 表示从中随机选择 TGP 的域，i 表示域 D 中的第 i 个样本。TGP 旨在为模型执行任务提供必要的上下文信息。为了进一步增强上下文能力，我们设计了一种额外类型的提示，TUP，它能够自适应地学习将多个任务整合到一个统一的框架中。TUP 实质上是所有 TGPs 共享的可学习软提示。我们引入两种方法来实现 TUP，分别表示为 $\bar{\boldsymbol{U}}$ 和 $\tilde{\boldsymbol{U}}$ 。第一种方法使用包含先验域信息的伪姿态，然后使用编码器将其投射到特征空间中。伪姿态是所有训练姿态的平均。另一种方法直接在特征空间中应用 TUP，其中 TUP 被分解为两个可学习提示的乘积。

给定一个查询输入 $\boldsymbol{Q}_j^D$ ，代表来自域 D 的第 j 个样本，随机从相同域 D 中选择一个 TGP $[\boldsymbol{X}_i^{\text{in}}, \boldsymbol{X}_i^{\text{gt}}]^D$ ，表示第 i 个样本。在获得基于引入的提示策略的查询和提示后，我们使用共享编码器将它们映射到高维特征空间，获得提示特征 $\boldsymbol{X}_P$ 和查询特征 $\boldsymbol{X}_Q$ ：

$$\boldsymbol{X}_P = \mathcal{E}([\boldsymbol{X}_i^{\text{in}}, \boldsymbol{X}_i^{\text{gt}}]^D); \boldsymbol{X}_Q = [\mathcal{E}(\boldsymbol{Q}_j^D), \boldsymbol{U}], \tag{2}$$

，其中 $\mathcal{E}(\cdot)$ 表示编码器，$[\cdot,\cdot]$ 表示沿时间轴的连接，TUP $\boldsymbol{U} \in \{\bar{\boldsymbol{U}}, \tilde{\boldsymbol{U}}\}$ 对域 D 是不可知的，可以通过任一方法实现。在 [?] 的启发下，我们在公式（1 ）中使用一个双流时空变换器实现模型 $\mathcal{M}(\cdot)$ ，其中每个流中的模块由两个分支组成，每个分支交替以不同顺序应用空间和时间注意。通过公式（2 ）获得的提示特征 $\boldsymbol{X}_P$ 和查询特征 $\boldsymbol{X}_Q$ 分别是查询流和提示流的初始输入。由于两个流共享相同的结构，我们以提示流的 l 层为例：

$$\boldsymbol{X}_P^{l+1} = \alpha \mathcal{T}_1(\mathcal{S}_1\left(\boldsymbol{X}_P^l\right)) + \beta \mathcal{S}_2\left(\mathcal{T}_2\left(\boldsymbol{X}_P^l\right)\right), \tag{3}$$

，其中 α 和 β 是可学习的平衡参数，$\mathcal{S}$ 和 $\mathcal{T}$ 分别表示空间和时间注意。在通过 L 层获取 $\boldsymbol{X}_P^{L+1}$ 和 $\boldsymbol{X}_Q^{L+1}$ 后，它们通过聚合函数 $\mathcal{A}(\cdot)$ 混合为 $\boldsymbol{X}^{L+1} = \mathcal{A}(\boldsymbol{X}_P^{L+1}, \boldsymbol{X}_Q^{L+1})$ ，其中 $\mathcal{A}(\cdot)$ 是具有自适应权重的求和函数。然后，模型将两个流合并为一个，其中混合特征 $\boldsymbol{X}^{L+1}$ 通过另一组 L' 层，该层的定义方式与公式（3 ）相同。

4.2 提示策略

与 PiC 中不反映任何领域分布模式的随机选择提示策略相比，我们在 HiC 中提出了一种最大-最小相似性提示采样（SPS）方法，该方法引入锚点以从训练数据中编码先验知识。如图 3 所示，我们获得了一组紧凑的硬锚，每个锚点都编码了来自相似分布的上下文信息簇。每个硬锚与一个独特的可学习的软锚配对，以动态适应和细化上下文。当模型收到一个查询时，该查询会与硬锚进行比较，并检索到最相关的硬锚及其对应的软锚作为提示，提供反映跨领域分布的上下文信息。该提示策略通过将训练数据的分布结构融入提示构建过程来提高上下文敏感学习。

相对运动相似性空间。使用第 3 节中的符号，令 $\boldsymbol{X} \in \mathbb{R}^{F\times J\times C}$ 表示一个由 F 帧、J 关节和 C 通道组成的运动序列，其中元素可以表示姿态中的位置坐标或网格中的轴角旋转。为简单起见，省略帧索引。我们首先定义任意两个运动序列 $\boldsymbol{X}_i^{D_1}$ 和 $\boldsymbol{X}_k^{D_2}$ 的相似性，其中 $\boldsymbol{X}_i^{D_1}$ 是域 D_1 中的第 i 个样本，而 $\boldsymbol{X}_k^{D_2}$ 是域 D_2 中的第 k 个样本。相似性 $\text{SIM}(\cdot,\cdot)$ 定义为两个运动序列之间的平均距离：

$$\text{SIM}(\boldsymbol{X}_i^{D_1}, \boldsymbol{X}_k^{D_2}) = -\frac{1}{FJ}\sum_{f=1}^{F}\sum_{j=1}^{J}\sqrt{\sum_{c=1}^{C}\left[\boldsymbol{X}_i^{D_1} - \boldsymbol{X}_k^{D_2}\right]^2_{(f,j,c)}}, \tag{4}$$

其中 (f, j, c) 表示帧 f ，关节 j ，和通道 c ，方程中的负号确保较小的值表示较低的相似性（不太相似），而较大的值表示较高的相似性（更相似）。这个相似性度量反映了两个运动序列之间平均关节级别的距离。

接下来，我们在整个动作序列训练集上构建一个相对运动相似性空间，该空间被定义为一个特征空间，其中每个动作序列根据其与一个标准身体配置的相对相似性以及与数据集中其他动作序列的关系被定位。标准身体配置（T 体）是通过将 SMPL [?] 模型的关节旋转参数设为零获得的，结果是一个人体自然地像字母 T 一样伸展。标准身体配置（T 体）在这个空间中作为参考点。图 3 （a）展示了一个由 11 个动作序列构建的相对运动相似性空间的示例，其中为了说明的目的，每个序列被映射为二维空间中的一个点。相对运动相似性空间有助于理解动作序列的分布模式，并从各种领域中识别出具有代表性的人体运动。

最大-最小相似性提示采样 (SPS)。如上所述，我们提出了最大-最小相似性提示采样，以在构建的相对运动相似性空间中迭代选择困难的锚点。图 3 (b) 展示了一个逐步从 11 个运动序列的集合中采样 4 个锚点的示意性玩具示例。对于任意域 D 中的任一样本 i 给定整个训练集的运动序列 $\mathcal{X} = \{\boldsymbol{X}_i^D\}$ ，我们提出最大-最小相似性提示采样，以便采样出一组较小的

Algorithm 1 最大-最小相似性提示采样 (SPS)

Input: A set of motion sequences $\mathcal{X}$, number of anchors K , similarity function $\text{SIM}(\cdot,\cdot)$ as defined in Eq. (4)
Output: A set of anchors $\mathcal{A} = \{\boldsymbol{A}_k\}_{k=1}^{K}$
Initialize:
Define $\boldsymbol{A}_1$ as a sequence of duplicated canonical bodies.
$\mathcal{A} \leftarrow \{\boldsymbol{A}_1\}$ // Set of anchors
$\mathcal{U} \leftarrow \mathcal{X} \setminus \mathcal{A}$ // Set of unsampled sequences
for $k = 2$ to K **do**
  // Step 1: Group Unsampled Sequences by Max Similarity
  for all $\boldsymbol{X}_i \in \mathcal{U}$ **do**
    Compute max similarity between $\boldsymbol{X}_i$ and anchors in $\mathcal{A}$:

$$\text{MAXSIM}(\boldsymbol{X}_i, \mathcal{A}) = \max_{\boldsymbol{A} \in \mathcal{A}} \text{SIM}(\boldsymbol{X}_i, \boldsymbol{A})$$

  end for
  // Step 2: Sample New Anchor by Min Similarity
  Find $\boldsymbol{X}^* \in \mathcal{U}$ with minimum $\text{MAXSIM}(\boldsymbol{X}_i, \mathcal{A})$:

$$\boldsymbol{X}^* = \arg\min_{\boldsymbol{X}_i \in \mathcal{U}} \text{MAXSIM}(\boldsymbol{X}_i, \mathcal{A})$$

  Define $\boldsymbol{A}_k$ as $\boldsymbol{X}^*$:

$$\boldsymbol{A}_k \leftarrow \boldsymbol{X}^*$$

  // Step 3: Update Sets
  $\mathcal{A} \leftarrow \mathcal{A} \cup \{\boldsymbol{A}_k\}$ // Add $\boldsymbol{A}_k$ to $\mathcal{A}$
  $\mathcal{U} \leftarrow \mathcal{U} \setminus \{\boldsymbol{A}_k\}$ // Remove $\boldsymbol{A}_k$ from $\mathcal{U}$
  if $\mathcal{U} = \emptyset$ **then**
    break
  end if
end for
return $\mathcal{A}$

K 运动序列作为困难锚点 $\mathcal{A} = \{\boldsymbol{A}_k\}_{k=1}^{K}$ 。

在最大-最小相似性提示采样过程中，我们跟踪两个集合：$\mathcal{A}$ 和 $\mathcal{U} = \mathcal{X} \setminus \mathcal{A}$ ，分别表示已采样和未采样的运动序列集合。为了初始化采样过程，我们使用标准的身体配置（T 形体）作为第一个锚点 $\boldsymbol{A}_1$ ，提供一个通用的参考基准。我们用第一个锚点更新锚点集合 $\mathcal{A} = \{\boldsymbol{A}_1\}$ ，并将其从 $\mathcal{U}$ 中移除。最大-最小相似性提示采样本质上涉及两个步骤的迭代：

1) 按照最大相似性将未采样的运动序列分组。使用在公式 (4) 中定义的相似性度量，我们计算每个 $\boldsymbol{X}_i \in \mathcal{U}$ 和每个 $\boldsymbol{A}_k \in \mathcal{A}$ 之间的相似性。对于每个 $\boldsymbol{X}_i \in \mathcal{U}$ ，我们在 $\mathcal{A}$ 中找到与 $\boldsymbol{X}_i$ 具有最大相似值的锚点：

$$\text{MAXSIM}(\boldsymbol{X}_i, \mathcal{A}) = \max_{\boldsymbol{A} \in \mathcal{A}} \text{SIM}(\boldsymbol{X}_i, \boldsymbol{A}). \tag{5}$$

这一步可以解释为基于具有最大相似值的锚点对未采样序列进行分组，其中锚点充当 “组代表”。它确保同一组内的序列共享相似的模式。

2) 通过最小相似度采样新锚点。基于分组的运动序列及其对应的 $\text{MAXSIM}(\boldsymbol{X}_i, \mathcal{A})$ 值，我们找到在所有 $\text{MAXSIM}(\boldsymbol{X}_i, \mathcal{A})$ 中取最小值的 $\boldsymbol{X}^* \in \mathcal{U}$ ：

$$\boldsymbol{X}^* = \arg\min_{\boldsymbol{X}_i \in \mathcal{U}} \text{MAXSIM}(\boldsymbol{X}_i, \mathcal{A}). \tag{6}$$

待采样的新锚点定义为 $\boldsymbol{X}^*$ ，并相应地更新集合 $\mathcal{A}$,$\mathcal{U}$ ：

$$\begin{aligned} &\boldsymbol{A}_k \leftarrow \boldsymbol{X}^*; \\ &\mathcal{A} \leftarrow \mathcal{A} \cup \{\boldsymbol{A}_k\}; \\ &\mathcal{U} \leftarrow \mathcal{U} \setminus \{\boldsymbol{A}_k\}. \end{aligned} \tag{7}$$

这一步可以解释为寻找运动序列组中表现出最为多样化模式的组，其中组内运动序列彼此之间的距离比其他组内的运动序列更远。然后，将距离 “组代表” 即其所在组的锚点最远的运动序列确定为新的锚点。这个过程确保了采样的运动序列分散多样，覆盖了相对运动相似性的不同区域，并且每个新锚点都与现有锚点集合中最为不同，从而增加了采样数据的多样性。

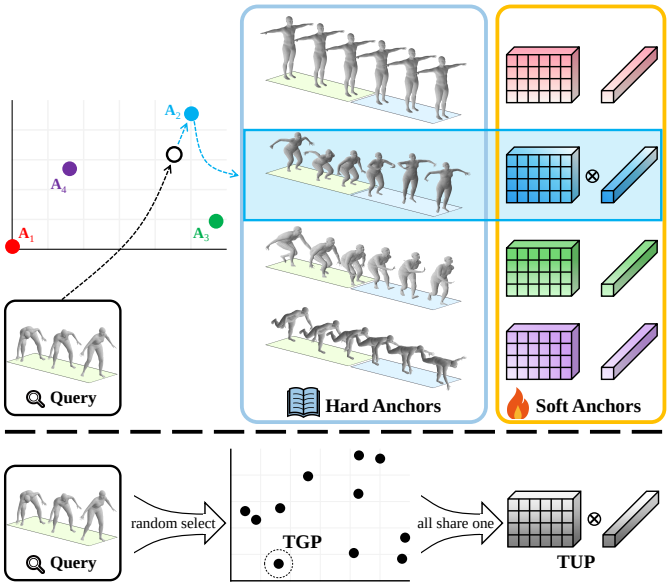


Fig. 4: 在 HiC（上）和 PiC（下）中，提示检索的示意图。在 HiC 中，基于与查询的相似性，从硬锚点中检索出特定的锚点，同时也检索出相应的软锚点。在 PiC 中，一个 TGP 从所有训练数据中随机选出，且所有 TGP 共享单个 TUP。

当满足以下条件之一时，最大-最小相似性提示采样过程停止：

$$\begin{aligned} &\text{Condition 1: } |\mathcal{A}| = K; \\ &\text{Condition 2: } \mathcal{U} = \emptyset, \end{aligned} \tag{8}$$

，其中 K 是预定的锚点数量，这个超参数在表示能力和计算效率之间进行平衡。尽管较大的 K 可以通过涉及更多的锚点来增加表示能力，但它也增加了计算负荷。在极端情况下，$\mathcal{A} = \mathcal{X}$ ，即 $\mathcal{U} = \emptyset$ ，其中所有训练数据都设置为锚点，会导致过拟合和较差的泛化能力。因此，在实践中，超参数 K 设置为远小于训练规模。具体值通过消融研究确定，如表 5 所示。算法 1 总结了最大-最小相似性提示采样过程。

硬锚点和软锚点。基于通过极大-极小相似性提示采样得到的硬锚点集合 $\{\boldsymbol{A}_k\}_{k=1}^{K}$ ，我们为每个 $\boldsymbol{A}_k$ 分配一个软锚点 $\boldsymbol{U}_k$ 。软锚点在高维隐藏表示空间中定义为 $\boldsymbol{U}_k = \boldsymbol{W}_1^k \boldsymbol{W}_2^k$ ，其中 $\boldsymbol{W}_1^k \in \mathbb{R}^{F \times J \times 1}$ 和 $\boldsymbol{W}_2^k \in \mathbb{R}^{1 \times 1 \times H}$ 是可训练的权重，H 是隐藏表示空间的维度。由于硬锚点是从一个统一的相对运动相似性空间中采样出来的，该空间跨越了多种模态、任务和数据集，因此它们可以提取领域相关的信息，而无需任何领域特定的设计。软锚点提供动态上下文的补充优化，使模型更具普适性（见表 6 ）。硬锚点和软锚点的结合使用，共同解决了跨越广泛领域和大规模数据的一般化挑战，使模型能够灵活编码领域相关信息（通过硬锚点）和可推广的领域模式（通过软锚点），以提高适应性。

提示检索。如图 4 所示，当向模型提出一个查询时，我们通过基于相似度的方法从上面定义的锚中检索到最相关的提示。首先，我们将查询输入序列 $\boldsymbol{Q}^{\text{in}}$ 投影到与硬锚相同的空间，即相对运动相似空间。其次，我们使用方程 (4) 计算 $\boldsymbol{Q}^{\text{in}}$ 与每个硬锚 $\boldsymbol{A}_k$ 之间的相似度，并获取与查询输入相似度最高的

硬锚。这些步骤具体描述如下：

$$\begin{aligned} \boldsymbol{A}^* &= \arg\max_{\boldsymbol{A}_k \in \mathcal{A}} \text{SIM}(\boldsymbol{Q}^{\text{in}}, \boldsymbol{A}_k); \\ \boldsymbol{P}^{\text{in}} &\leftarrow \boldsymbol{A}^*, \end{aligned} \tag{9}$$

其中 $\boldsymbol{P}^{\text{in}}$ 表示提示输入。由于锚最初来自训练集，我们还获得了与 $\boldsymbol{P}^{\text{in}}$ 相对应的真实目标 $\boldsymbol{P}^{\text{gt}}$ 。然后，我们检索与硬锚 $\boldsymbol{A}^*$ 对应的软锚 $\boldsymbol{U}^*$ 。提示输入 $\boldsymbol{P}^{\text{in}}$ 和提示目标 $\boldsymbol{P}^{\text{gt}}$ 的组合表示一个硬提示，而 $\boldsymbol{U}^*$ 代表一个软提示。模型接受 $\boldsymbol{Q}^{\text{in}}$ 、$\boldsymbol{P}^{\text{in}}$ 、$\boldsymbol{P}^{\text{gt}}$ 和 $\boldsymbol{U}^*$ 作为进一步处理的输入。所提出的提示策略有助于 Human-in-Context 在涉及各种领域的不同上下文中有效适应，同时保持稳健的泛化能力。

除了提示策略之外，情境学习的另一个关键组件是网络架构，该架构定义了如何处理提示和查询以在特定的上下文中生成最终输出。在 Human-in-Context 中，我们提出 X-Fusion Net 来实现网络架构。

给定网络需要同时从各种模态、任务和数据集中学习的异常大规模数据，要求一种比现有架构更有效和更为稳健的架构。为此，我们提出了 X-Fusion Net，其被公式化为：

$$\mathcal{M}(\boldsymbol{Q}^{\text{in}}, \boldsymbol{P}^{\text{in}}, \boldsymbol{P}^{\text{gt}}, \boldsymbol{U}^*) \rightarrow \boldsymbol{Q}^{\text{out}}, \tag{10}$$

，其中，$\boldsymbol{Q}^{\text{in}}, \boldsymbol{Q}^{\text{out}}, \boldsymbol{P}^{\text{in}}, \boldsymbol{P}^{\text{gt}} \in \mathbb{R}^{F \times J \times C}$ 分别代表查询输入/输出和提示输入/目标，$\boldsymbol{U}^* \in \mathbb{R}^{F \times J \times H}$ 是软锚点，H 表示隐藏表示的维度。尽管 $\mathcal{M}(\cdot)$ 可以通过支持多输入的任何现有网络进行实例化，但由于在单次训练中在多模态、任务和数据集上进行上下文学习的极大困难，性能将不尽如人意。在 Human-in-Context 中，我们通过提出 X-Fusion Net 来实例化 $\mathcal{M}(\cdot)$ ，如图 2 所示。X-Fusion Net 具有双分支结构，即查询和提示分支，分支间有上下文注入。查询和提示分支都采用了 X-Fusion 块作为其基本块，其架构如图 5 所示。

4.2.1 上下文编码

X-Fusion Net 采用双分支结构设计，由查询（Q ）和提示（P ）组成，

$$\begin{aligned} \boldsymbol{H}_Q &= \mathcal{E}_Q\big([\boldsymbol{Q}^{\text{in}} \parallel \boldsymbol{P}^{\text{gt}}]\big) + \boldsymbol{U}^*; \\ \boldsymbol{H}_P &= \mathcal{E}_P\big([\boldsymbol{P}^{\text{in}} \parallel \boldsymbol{P}^{\text{gt}}]\big), \end{aligned} \tag{11}$$

其中 $\boldsymbol{H}_Q$ 和 $\boldsymbol{H}_P \in \mathbb{R}^{F \times J \times H}$ 表示查询和提示的上下文特征，$[\cdot \| \cdot]$ 表示沿通道轴的连接。此外，$\mathcal{E}_Q$ 和 $\mathcal{E}_P$ 是查询和提示编码器，用 MLPs 和位置嵌入来实现 [**?**] 。查询和提示的上下文特征通过查询和提示分支传递。每个分支包含一系列 X-Fusion 块作为其基本模块。

4.2.2 *X-Fusion* 块

上下文表示结合了可以在不同层次捕获的多种依赖关系。具体来说，上下文依赖关系可以在空间和时间视图中建模，从全局和局部范围中提取，并在嵌入空间、图空间和状态空间中学习。为此，所提出的 X-Fusion 块包含两个组件：1）多层次的上下文聚合，2）跨层次的上下文更新。这两个组件共同提升了在不同层次内和跨层次有效捕获和融合上下文依赖关系的能力。

多级上下文聚合是通过一组聚合函数实现的，这些函数记作 $\{\text{AGGREGATE}^l(\cdot)\}_{l=1}^L$ ，由级别 $l = 1, 2, \cdots, L$ 索引。跨级上下文更新是通过一个更新函数实现的，该函数记作 $\text{UPDATE}(\cdot)$ 。结合多级聚合函数和更新函数，X-Fusion 块被表述为：

$$\begin{aligned} \boldsymbol{Z} &= \text{XFUSION}(\boldsymbol{H}) \\ &= \text{UPDATE}\big(\{\text{AGGREGATE}^l(\boldsymbol{H})\}_{l=1}^L\big), \end{aligned} \tag{12}$$

，其中 $\boldsymbol{H} \in \mathbb{R}^{F \times J \times H}$ 代表输入，$\boldsymbol{Z} \in \mathbb{R}^{F \times J \times H'}$ 代表当前 X-Fusion 块的输出。

接下来，我们讨论这些函数的具体实现，并解释它们在 X-Fusion 块中的各自作用。

多级上下文聚合。给定前一层的上下文表示 $\boldsymbol{H} \in \mathbb{R}^{F \times J \times H}$ ，不同级别的聚合函数 $\{\text{AGGREGATE}^l(\cdot)\}_{l=1}^L$ 的设计目标是通过专注于不同的依赖关系来提取上下文特征，它们各自的输出是：

$$\boldsymbol{Y}^l = \text{AGGREGATE}^l(\boldsymbol{H}). \tag{13}$$

首先，我们在空间和时间视角中解开上下文依赖关系。关于时空依赖关系，$\boldsymbol{H} \in \mathbb{R}^{F \times J \times H}$ 可以在空间或时间视角中建模。在空间视角中，$\boldsymbol{H}$ 被解释为每帧 $\boldsymbol{H}_{\text{S}} \in \mathbb{R}^{J \times H}$ 的空间特征，而在时间视角中，$\boldsymbol{H}$ 被解释为每个关节点的时间特征 $\boldsymbol{H}_{\text{T}} \in \mathbb{R}^{F \times H}$ 。由于空间特征和时间特征的处理方式相似，我们将以时间特征为例，并省略下标 S/T 来演示如何应用多级上下文聚合。

接下来，在每一个视图中，我们使用不同的实现方法在多个层次应用聚合函数。具体来说，我们采用自注意力（SelfAttn）、图卷积（GraphConv）和状态空间模型（SSM）来实现如下聚合函数：

Implementation	Dependency
Level 1: $\boldsymbol{Y}^1 = \text{SelfAttn}(\boldsymbol{H})$	embed-space; global
Level 2: $\boldsymbol{Y}^2 = \text{GraphConv}(\boldsymbol{H})$	graph-space; local
Level 3: $\boldsymbol{Y}^3 = \text{SSM}(\boldsymbol{H})$	state-space; local

(14)

。自注意力、图卷积和状态空间模型的协作优势帮助网络捕捉在不同特征空间和不同范围内反映的依赖关系，它们的联合和个体贡献可以在图 **??** 中看到。在推理过程中，多层聚合能够为每一个提示-查询对动态学习最相关的依赖关系，这在图 7 中的可视化验证了这一点。跨层次上下文更新。在通过多层次上下文聚合获得代表不同层次依赖关系的一组特征 $\boldsymbol{Y}^l \in \mathbb{R}^{F \times H'}$ 之后，我们应用跨层次上下文更新来生成 X-Fusion 块的最终输出 $\boldsymbol{Z}$ ：

$$\boldsymbol{Z} = \text{UPDATE}\big(\{\boldsymbol{Y}^l\}_{l=1}^L\big). \tag{15}$$

。跨层次上下文更新旨在根据动态评估其各自贡献的准确性权重来动态更新聚合特征。具体来说，跨层次上下文更新在每一 $t = 1, 2, \cdots, F$ 中的 $\{\boldsymbol{y}_t^l\}_{l=1}^L$ 上逐帧应用，其中 $\boldsymbol{y}_t^l \in \mathbb{R}^{H'}$ 是 $\boldsymbol{Y}^l$ 的 t 行，这意味着每个层次的贡献是逐帧评估的。

首先，不同层级在特定帧 t 的影响通过将上下文特征 $\{\boldsymbol{y}_t^l\}_{l=1}^L$ 压缩成一串数字 $\{a_t^l\}_{l=1}^L$ 来确定，其中每个数字对应于不同的层级，并代表其对最终输出的影响。压缩有助于去除冗余并找出最具辨识度的信息。为开始上下文压缩步骤，向量集合 $\{\boldsymbol{y}_t^l\}_{l=1}^L$ 在通道维度上被连接成一个更长的 LH' 维向量。上下文压缩步骤定义为：

$$\begin{bmatrix} a_t^1 \\ a_t^2 \\ \vdots \\ a_t^L \end{bmatrix} = \boldsymbol{W}^{\text{compress}} \begin{bmatrix} \boldsymbol{y}_t^1 \\ \boldsymbol{y}_t^2 \\ \vdots \\ \boldsymbol{y}_t^L \end{bmatrix} + \begin{bmatrix} b^1 \\ b^2 \\ \vdots \\ b^L \end{bmatrix}, \tag{16}$$

，其中 $\boldsymbol{W}^{\text{compress}} \in \mathbb{R}^{L \times LH'}$ 是一个可训练的压缩矩阵，并在不同的帧 t 之间共享，向量 $[b^1, b^2, \cdots, b^L]^\top \in \mathbb{R}^L$ 是与每个聚合层级相关的可训练偏置项，也在不同帧 t 之间共享。获得的向量 $[a_t^1, a_t^2, \cdots, a_t^L]^\top \in \mathbb{R}^L$ 包含了表明每个层级特定于帧的影响的分数。然后，我们应用 softmax 函数来规范化影响分数：

$$[\alpha_t^1, \alpha_t^2, \cdots, \alpha_t^L] = \text{Softmax}([a_t^1, a_t^2, \cdots, a_t^L]). \tag{17}$$

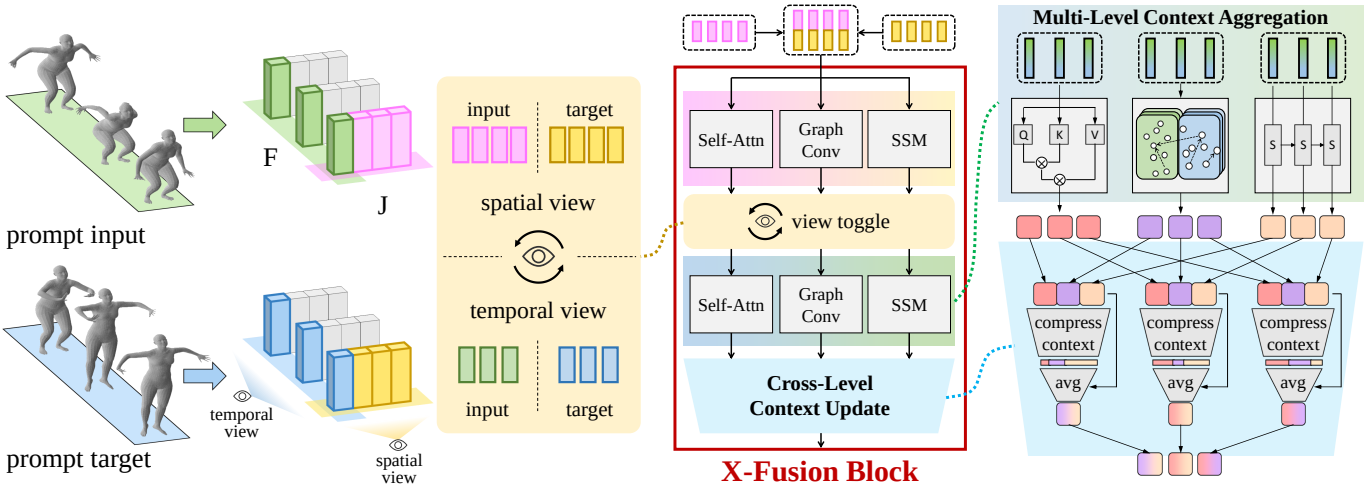


Fig. 5: X-Fusion 块的插图。X-Fusion 块应用多层次上下文聚合和跨层次上下文更新，以捕获和融合不同层次内及层次之间的上下文依赖，同时在空间和时间视图之间切换。在多层次上下文聚合中，我们利用自注意力、图卷积和状态空间模型分别在嵌入空间中学习全局依赖、在图空间中学习局部依赖以及在状态空间中学习局部依赖。在跨层次上下文更新中，我们压缩上下文以消除冗余，并更新特征，重点关注贡献最大的层次。

通过根据帧特定的影响分数 α_t^l 融合 $\{\boldsymbol{y}_t^l\}_{l=1}^L$，获得交叉更新特征更新的最终输出 $\boldsymbol{Z} \in \mathbb{R}^{F\times J\times H'}$（以时间视角）。$\boldsymbol{Z}$ 的逐帧表示是：

$$\boldsymbol{z}_t = \sum_{l=1}^{L} \alpha_t^l \boldsymbol{y}_t^l, \tag{18}$$

，其中 $\boldsymbol{z}_t \in \mathbb{R}^{H'}$ 是当前 X-Fusion 块最终输出 $\boldsymbol{Z}$ 中第 t 行。基于时间和空间视图之间的类比，$\boldsymbol{Z} \in \mathbb{R}^{F\times J\times H'}$（在空间视图中）的联合表示 $\boldsymbol{z}_j \in \mathbb{R}^{H'}$ 可由 $\boldsymbol{z}_j = \sum_{l=1}^{L} \alpha_j^l \boldsymbol{y}_j^l$ 推断，其中 $\{\boldsymbol{y}_j^l\}_{j=1,\cdot,J}^{l=1,\cdots,L}$ 是通过应用于输入上下文表示 $\boldsymbol{H}$ 的空间视图 $\boldsymbol{H}_S$ 的多级聚合获得的，α_j^l 是通过应用于 $\{\boldsymbol{y}_j^l\}$ 的跨层更新获得的联合特异性影响得分，其压缩矩阵 $\boldsymbol{W}^{\text{compress}}$ 在空间视图和时间视图之间共享。与简单平均多个层的输出相比，动态上下文感知更新提供了更多适应性，导致更好的性能，如表 8 所示。

在层 k 中，查询（Q）和提示（P）分支各包含一个 X-Fusion 模块来提取各自的上下文特征，可以写为：

$$\boldsymbol{Z}_Q^{(k)} = \text{XFusion}_Q^{(k)}(\boldsymbol{H}_{\boldsymbol{Q}}^{(\boldsymbol{k})}); \boldsymbol{Z}_P^{(k)} = \text{XFusion}_P^{(k)}(\boldsymbol{H}_{\boldsymbol{P}}^{(\boldsymbol{k})}), \tag{19}$$

，其中 $\boldsymbol{H}_{\boldsymbol{Q}}^{(\boldsymbol{k})}, \boldsymbol{H}_{\boldsymbol{P}}^{(\boldsymbol{k})} \in \mathbb{R}^{F\times J\times H}$ 分别是查询和提示分支在 X-Fusion 模块中的输入，而 $\boldsymbol{Z}_Q^{(k)}, \boldsymbol{Z}_P^{(k)} \in \mathbb{R}^{F\times J\times H'}$ 是输出。随着查询和提示分支各自层次的加深，它们获得了更深层次的上下文理解，但查询分支应该意识到来自提示的上下文信息，因为查询分支是生成最终查询输出的主要分支。为此，我们在每一层应用一个上下文注入模块 $\text{ContextInject}(\cdot)$，将提示上下文注入查询分支，促使它学习提示和查询在不同深度间的依赖性。应用上下文注入模块后，当前层中查询和提示分支各自的输出为：

$$\boldsymbol{H}_Q^{(k+1)} = \text{ContextInject}(\boldsymbol{Z}_P^{(k)}, \boldsymbol{Z}_Q^{(k)}); \boldsymbol{H}_P^{(k+1)} = \boldsymbol{Z}_P^{(k)}, \tag{20}$$

其中 $\boldsymbol{H}_Q^{(k+1)}, \boldsymbol{H}_P^{(k+1)}$ 表示下一层的输入（如果存在）。在 HiC 中，上下文注入模块简单地通过求和函数实现，如 $\boldsymbol{H}_Q^{(k+1)} = \boldsymbol{Z}_P^{(k)} + \boldsymbol{Z}_Q^{(k)}$ 所示。

为了在优化之前避免对某些聚合级别的偏向偏好，压缩矩阵 $\boldsymbol{W}^{\text{compress}}$ 和偏差项 $[b^1, b^2, \cdots, b^L]$ 根据以下条件进行初始化：这些条件确保在任何优化之前，任何帧 t 和任何级别 l 的影响得分具有相同的值 $\alpha_t^l = \frac{1}{L}$，这表明在优化之前对不同聚合级别贡献的无偏假设。

在训练过程中，损失函数涉及计算姿态关节位置和速度，并且还涉及计算网格顶点位置、关节旋转参数和形状参数，遵循 [?]。更多的网络和优化细节请参考补充材料。

5 实验

实现细节。我们实施了所提出的最大-最小相似度提示采样以获得 $|\mathcal{A}| = 800$ 锚点。对于所提出的模型，我们使用 $K = 8$ 层和隐藏特征维度 $H = 128$。运动序列包含 $F = 16$ 帧和 $J = 24$ 关节。对于需要掩码的任务，我们应用 40 % 的掩码比率。我们在 PyTorch 中实现 Human-in-Context，使用 AdamW 优化器，学习率线性衰减，从 0.0002 开始，每个 epoch 后减少 1 %。我们在一台配备 4 块 NVIDIA A6000 GPU 的 Linux 机器上训练 Human-in-Context 120 个 epoch。

实验设置。为了验证 HiC 作为统一跨域模型的有效性，我们将其与一系列基准方法进行比较，包括 MotionBERT [?]、PoseRetNet [?]、TCPFormer [?]、HoT [?] 以及 PiC [?]。所有模型均被重新实现，以符合统一跨域三维人体运动建模的设置，即在不使用任何特定领域模型头的情况下，对所有任务和数据集进行一次性训练。

数据集、任务和指标。为了进行性能评估，我们使用四个大规模的 3D 人体运动数据集，包括 AMASS [?]、Human3.6M [?]、FreeMan [?] 和 3DPW [?]。为了验证在域内和域外的泛化能力，我们使用 AMASS、Human3.6M 和 FreeMan 作为域内数据集，而使用 3DPW 作为域外数据集。更具体来说，我们将 AMASS、Human3.6M 和 FreeMan 各自分成训练集和测试集，按照其各自的分割标准。我们使用 3DPW 作为测试集。所有模型都是在 AMASS、Human3.6M 和 FreeMan 上训练，然后在所有四个数据集上进行测试。每个模型最多在表格 2 中描述的 10 个任务上进行训练和测试。在这 10 个任务中，具有姿态输出的任务使用每个关节位置误差均值（MPJPE）进行评估，具有网格输出的任务使用每个顶点误差均值（MPVE）进行评估，这两个评估均符合标准协议。

5.1 定量结果

域内数据集。表 3 展示了其域内泛化能力通过三个数据集进行广泛评估：AMASS [?]、Human3.6M [?] 和 FreeMan [?]

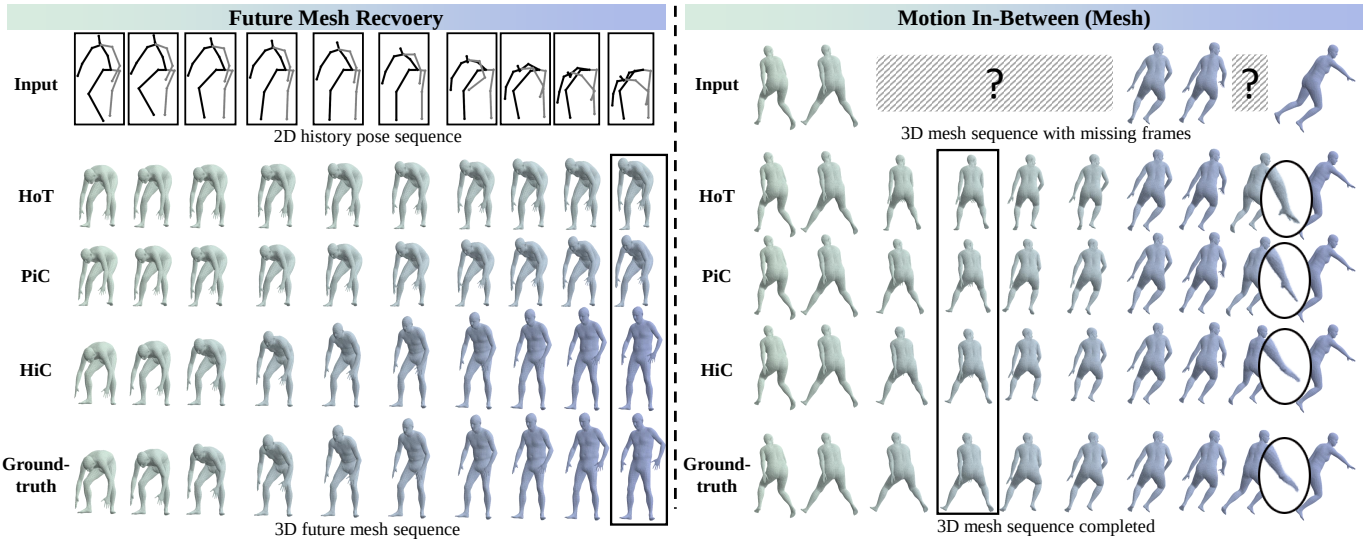


Fig. 6: 未来网格恢复（左）和中间运动（网格）（右）的定性结果。显著的改进用方框/圆圈标出，并放大了细节以提高清晰度。更多定性结果请参阅补充材料。

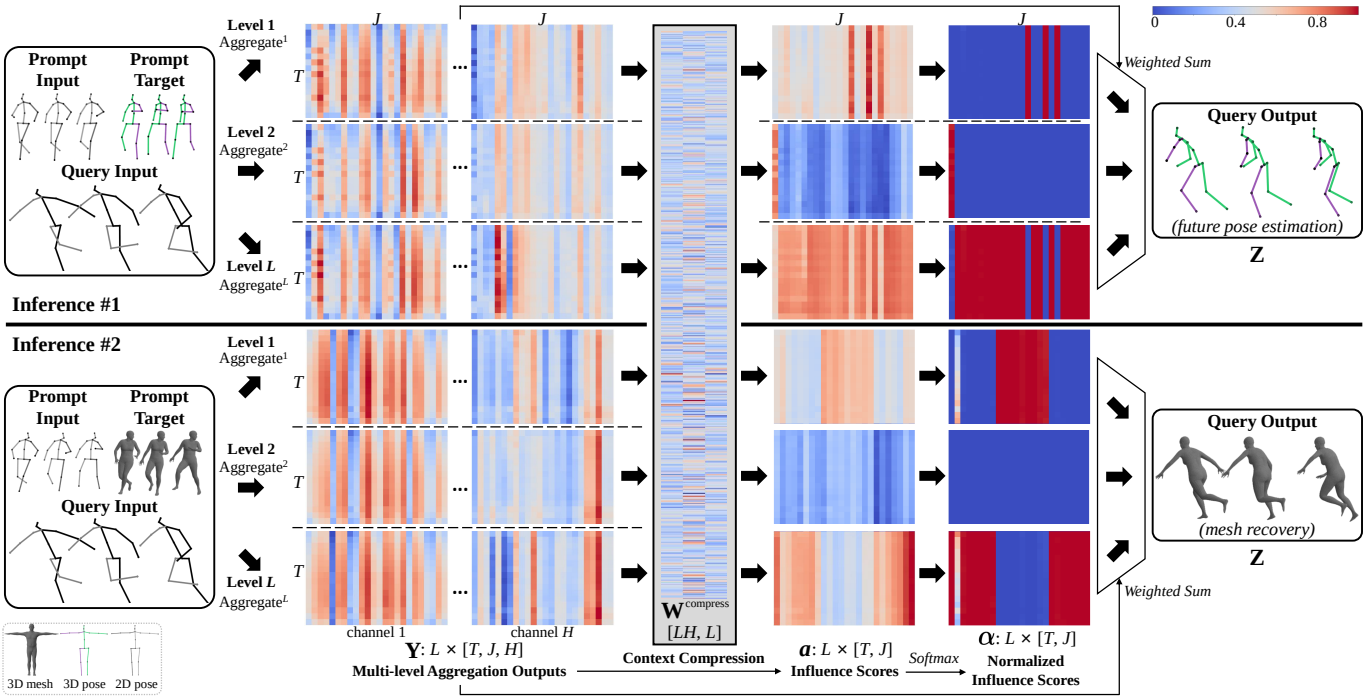


Fig. 7: 两个推理过程产生的特征和权重的可视化。在每个推理中，不同的提示与相同的查询配对，我们可视化了多级聚合的结果输出、压缩权重和跨级更新中的影响分值。这表明，即使对于相同的查询，X-Fusion Net 中的多级聚合和跨级更新也可以适应不同的提示，通过多个级别的聚合学习依赖关系，识别出帧级和关节级的最具影响力的上下文特征，并在不同的上下文下动态生成所需的输出。这也验证了在多个级别聚合上下文特征并通过动态权衡不同级别的贡献来更新它们的必要性。

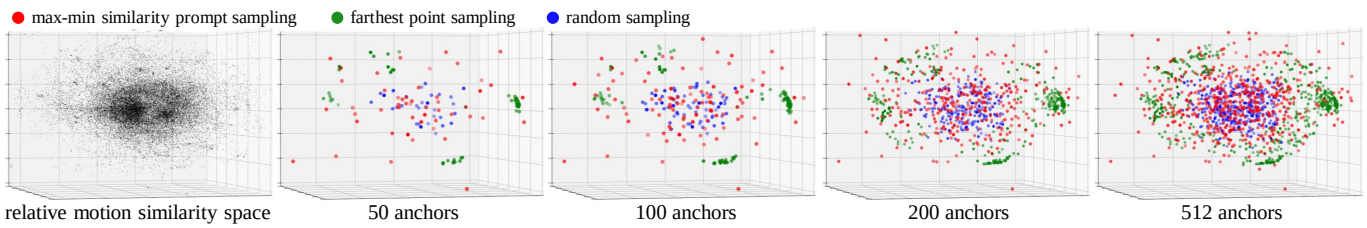


Fig. 8: 我们提出的最大-最小相似性提示采样、最远点采样和随机采样之间的视觉比较。应用在相同的相对运动相似性空间中，基于 FPS 的锚点（绿色）集中在几个外围区域，而基于随机的锚点（蓝色）往往聚集在最密集的区域（具有压倒性的更高概率）。相反，基于 SPS 的锚点（红色）更好地捕捉了所有数据的整体分布模式，覆盖了外围和中心，以及密集和稀疏区域，这对于在域内数据和域外数据之间的稳健泛化至关重要。

TABLE 3: 在三个域内数据集: AMASS [?] 、Human3.6M [?] 和 FreeMan [?] 上的基于姿态和基于网格的任务结果。† 表示重新实现以与统一跨域 3D 人体运动建模的设置对齐，也就是说，在所有任务和数据集上使用完全统一的模型进行单一的训练过程。较低的数字表示更好的性能。

(a) AMASS [?]

Models	Venue	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)					Mesh (MPVE ↓)				
MotionBERT † [?]	ICCV'23	54.27	68.76	56.40	36.98	34.46	72.07	83.94	65.58	72.18	53.30
PoseRetNet † [?]	ECCV'24	40.58	46.74	64.42	39.47	33.58	478.51	481.95	79.86	67.14	59.85
TCPFormer † [?]	AAAI'25	71.88	81.13	70.92	56.95	55.67	72.20	84.89	75.40	58.51	53.86
HoT † [?]	CVPR'24	39.06	59.31	67.78	38.33	30.36	65.46	86.01	76.24	61.47	48.03
PiC † [?]	CVPR'24	32.65	42.68	50.94	32.19	25.39	43.34	56.35	63.59	48.57	38.64
HiC	Ours'25	24.96	34.06	41.93	24.98	17.23	38.03	48.98	55.90	44.42	34.44

(b) Human3.6M [?]

Models	Venue	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)					Mesh (MPVE ↓)				
MotionBERT † [?]	ICCV'23	98.36	162.48	83.72	103.95	47.03	108.82	201.50	87.25	134.59	59.14
PoseRetNet † [?]	ECCV'24	96.33	135.22	93.93	103.41	54.56	328.24	343.60	104.39	136.80	67.01
TCPFormer † [?]	AAAI'25	95.53	150.46	87.87	105.41	76.56	118.89	192.58	101.04	141.26	90.72
HoT † [?]	CVPR'24	74.69	119.40	78.48	92.05	41.42	112.45	188.93	92.42	130.80	58.16
PiC † [?]	CVPR'24	74.35	116.86	76.10	91.14	37.83	95.38	155.42	82.02	127.54	50.71
HiC	Ours'25	62.94	115.79	58.37	87.81	24.93	78.67	151.53	69.52	125.67	42.89

(c) 自由人 [?]

Models	Venue	Joint Completion	Motion Prediction	Motion In-Between	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)			Mesh (MPVE ↓)		
MotionBERT † [?]	ICCV'23	113.29	99.22	61.33	112.97	130.24	68.74
PoseRetNet † [?]	ECCV'24	141.43	120.19	88.40	137.64	142.09	89.12
TCPFormer † [?]	AAAI'25	134.54	128.35	107.46	186.41	201.44	168.54
HoT † [?]	CVPR'24	124.20	103.31	70.23	127.39	133.17	72.19
PiC † [?]	CVPR'24	91.57	90.25	48.69	85.30	117.73	41.78
HiC	Ours'25	82.01	82.21	37.35	83.33	112.89	38.86

TABLE 4: 在 3DPW [?] 数据集上的结果。为了评估域外泛化能力，所有模型在 3 个域内数据集上进行训练（Human3.6M、AMASS 和 FreeMan），然后在 3DPW 数据集上进行测试。数值越低表示性能越好。

Models	Venue	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)					Mesh (MPVE ↓)				
MotionBERT † [?]	ICCV'23	118.83	140.74	77.83	71.64	44.48	145.42	171.86	94.95	97.28	56.10
PoseRetNet † [?]	ECCV'24	120.88	132.30	81.91	71.20	48.94	314.26	311.91	97.30	95.12	68.59
TCPFormer † [?]	AAAI'25	126.25	139.09	90.38	84.52	71.20	136.54	155.79	100.98	97.72	71.68
HoT † [?]	CVPR'24	107.95	126.62	73.31	66.57	40.73	134.14	159.96	96.14	97.84	62.51
PiC † [?]	CVPR'24	107.22	120.73	71.60	62.96	36.15	133.05	151.83	92.53	93.49	53.69
HiC	Ours'25	96.36	111.93	56.71	55.97	23.00	119.45	141.95	82.60	84.44	35.28

。在所有三个数据集上仅进行一次训练后，Human-in-Context 在所有数据集的所有任务中，无论是基于姿态还是网格的方法，始终优于所有其他方法。

1) 基于姿态的任务包括姿态估计、未来姿态估计、关节点补全（姿态）、运动预测（姿态）和运动中间（姿态）。这些任务的评估指标是关节位置平均误差（单位为毫米），用于测量输出关节位置与真实值的平均距离，该距离是在对齐根关节后得到的。在 Human3.6M 数据集上，HiC 在所有任务中都达到了最先进的水平。在姿态估计方面，它以 62.94 毫米的得分超越了 MotionBERT、PoseRetNet、TCPFormer 和 PiC。类似地，在未来姿态估计中，HiC 得到 115.79 毫米的得分。在关节点补全、运动预测和基于姿态的运动中间任务中，HiC 也一致优于其他模型。在运动中间（姿态）任务中，HiC 的 MPJPE 为 24.93 毫米，远低于 MotionBERT 的 47.03 毫米。同样，在 AMASS 数据集的所有任务上，HiC 表现良好。在姿态估计方面，HiC 的 MPJPE（24.96 毫米）低于其他方法，如 MotionBERT（54.27 毫米）和 PoseRetNet（40.58 毫米）。此外，它在未来姿态估计（34.06 毫米）、关节点补全（41.93 毫米）和运动预测（24.98 毫米）方面表现出色。在运动中间（姿态）任务中，HiC 表现最佳，达到了 17.23 毫米。对于 FreeMan 数据集，HiC 在所有适用任务中也优于其他方法。

2) 基于网格的任务包括网格恢复、未来网格恢复、关节补全（网格）、运动预测（网格），以及介于两者之间的运动（网格）。这些任务的评估指标是每顶点平均误差，单位为毫米，用于测量在对齐根关节后，估算的顶点与真实顶点之间的平均距离。在 Human3.6M 数据集上，HiC 在网格恢复中达到 78.67 毫米的误差，这明显优于其他方法。在未来网格恢复中，HiC 达到 151.53 毫米的误差，优于所有其他模型。在关节补全和运动预测（网格）方面，HiC 显示出比其他方法更低的误差。在 AMASS 数据集上，HiC 在网格恢复中达到 38.03 毫米的误差，在未来网格恢复中达到 48.98 毫米的误差，优于所有其他模型。HiC 在运动预测（网格）和关节补全（网格）上也表现良好。在 FreeMan 数据集上，HiC 在所有适用任务中也取得了最佳结果。

TABLE 5: 关于使用最大-最小相似性提示采样选择的锚点数量的消融研究。模型在情境设置下使用 AMASS、Human3.6M 和 FreeMan 进行训练，然后在 AMASS 和 3DPW 上进行测试。

Datasets	Anchors	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)					Mesh (MPVE ↓)				
AMASS	256	27.08	34.77	43.98	29.34	20.84	36.00	45.63	53.43	43.80	33.25
	512	27.20	34.71	42.71	27.83	19.63	37.42	46.81	54.67	43.92	33.22
	800	24.96	34.06	41.93	24.98	17.23	38.03	48.98	55.90	44.42	34.44
	1024	26.01	35.43	44.40	26.38	18.87	39.95	51.00	57.41	45.10	35.43
3DPW	256	104.99	119.82	62.54	63.85	27.40	126.37	149.06	87.57	108.29	37.35
	512	99.87	113.21	62.35	58.41	28.42	121.54	141.15	83.60	92.39	35.76
	800	96.36	111.93	56.71	55.97	23.00	119.45	141.95	82.60	84.44	35.28
	1024	97.11	113.85	57.59	58.12	24.18	121.91	143.46	83.68	86.29	35.91

TABLE 6: 关于软锚点的消融实验。这些模型在情境设置下接受 AMASS、Human3.6M 和 FreeMan 的数据训练，然后在 AMASS 和 3DPW 上进行测试。

Datasets	#	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)					Mesh (MPVE ↓)				
AMASS	w/o soft anchors	29.06	36.34	44.75	29.96	21.19	39.01	49.75	56.37	45.70	34.78
	w soft anchors	24.96	34.06	41.93	24.98	17.23	38.03	48.98	55.90	44.42	34.44
3DPW	w/o soft anchors	101.45	115.25	64.06	60.33	31.63	123.21	143.48	85.16	94.02	37.34
	w/ soft anchors	96.36	111.93	56.71	55.97	23.00	119.45	141.95	82.60	84.44	35.28

TABLE 7: 关于采样方法的消融研究。模型在 AMASS、Human3.6M 和 FreeMan 数据集上进行上下文训练，然后在 AMASS 和 3DPW 数据集上进行测试。

Datasets	#	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)					Mesh (MPVE ↓)				
AMASS	random	32.20	43.51	52.67	36.62	33.30	44.76	54.53	61.92	47.64	36.54
	FPS	30.86	40.33	49.83	33.97	24.67	42.71	52.32	60.41	46.78	35.79
	cluster	29.15	40.06	45.79	28.53	21.41	43.03	55.85	60.50	49.60	39.35
	SPS	24.96	34.06	41.93	24.98	17.23	38.03	48.98	55.90	44.42	34.44
3DPW	random	105.09	118.80	70.99	67.20	38.64	132.57	158.67	91.51	92.39	48.24
	FPS	104.35	118.45	68.47	65.49	37.58	127.45	156.90	88.41	92.25	44.24
	cluster	99.63	115.61	63.01	59.66	26.09	123.72	148.19	87.09	88.12	41.66
	SPS	96.36	111.93	56.71	55.97	23.00	119.45	141.95	82.60	84.44	35.28

域外数据集。正如表 4 所示，域外泛化能力在 3DPW [?] 数据集上进行了评估。在对三个域内数据集进行一次训练过程后，Human-in-Context 可以有效地泛化到域外数据，超越了所有其他方法在所有基于姿态和基于网格的任务中的表现。在 3DPW 数据集上，HiC 在基于姿态的任务中优于所有其他方法。在姿态估计中，HiC 达到 96.36 mm。在未来姿态估计中，HiC 达到 111.93 mm，低于其他模型。在关节补全（姿态）中，HiC 达到 56.71 mm。在运动预测（姿态）中，HiC 达到 55.97 mm，比其他模型更好。在运动过渡（姿态）中，HiC 同样表现优于其他模型。在网格恢复中，HiC 达到 119.45 mm。在未来网格恢复和关节补全（网格）中，HiC 同样优于其他方法。在运动预测（网格）和运动过渡（网格）中，HiC 分别达到 84.44 mm 和 35.28 mm。

5.2 定性结果

5.2.1 网格恢复和中间运动

图 6 展示了两个不同任务的定性结果，左侧是未来网格恢复，右侧是中间运动（网格）。这些任务使用三种不同的方法进行评估：HoT、PiC 和 HiC，并与真实数据进行比较。

1) 对于未来网格恢复任务，输入是历史的二维姿态，输出是未来的三维网格。输入在第一行中被表示为线条图。在第二行中，显示了 HoT 的结果，其中网格恢复存在一些不准确，尤其是在躯干和四肢区域，与真实值比较导致恢复不完美。第三行显示了 PiC 的结果。最后，第四行显示了 HiC 的结果，展示了最准确的网格恢复，人的网格与真实值非常相似，特别是在关节的位置和整体人体形态方面。

2) 对于运动插值任务，目标是在 3D 网格序列中完成缺失帧。第二、第三和第四行分别展示了 HoT、PiC 和 HiC 在插值姿势时的结果。HoT 在四肢的运动和关节的位置上表现出明显的差异。同样地，PiC 在运动过渡上表现出一些不准确，身体的某些部分并未被平滑插值。相比之下，第四行展示的 HiC 结果显示出更为准确的姿势过渡，肢体运动更加自然并且与真实运动保持一致。

5.2.2 特征和权重的可视化

图 7 显示了在 X-Fusion Net 的多层级聚合过程中学习到的特征和权重。对于每个提示-查询对，图中展示了与不同聚合层级对应的输出、上下文压缩过程以及每个层级的影响评分。影响评分被归一化，并且输出是通过动态调整来自多个聚合层级的特征生成的。可视化展示了模型如何学习不同聚合层级之间的依赖关系，以及在跨层级更新中如何选择和加权最具贡献性的上下文特征，包括逐帧和逐联合特征。这些结果强调了模型在不同上下文下生成各种任务的准确输出的能力，

TABLE 8: 跨层上下文更新的消融实验。静态表示对所有层的输出取平均，而不是在 X-Fusion Net 中动态地衡量它们的贡献。这些模型在 AMASS、Human3.6M 和 FreeMan 的上下文设置下进行训练，然后在 AMASS 和 3DPW 上进行测试。

Datasets	#	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
		Pose (MPJPE ↓)					Mesh (MPVE ↓)				
AMASS	static	35.40	46.87	63.95	37.04	28.95	38.92	49.74	56.23	44.55	34.42
	dynamic	24.96	34.06	41.93	24.98	17.23	38.03	48.98	55.90	44.42	34.44
3DPW	static	100.68	114.10	65.29	59.98	30.55	122.16	145.82	84.62	93.07	36.27
	dynamic	96.36	111.93	56.71	55.97	23.00	119.45	141.95	82.60	84.44	35.28

TABLE 9: 关于模型层数的消融实验。模型在 AMASS、Human3.6M 和 FreeMan 数据集上进行上下文训练，然后在 AMASS 和 3DPW 上进行测试。

Datasets	Layers	Params	Pose Estimation	Future Pose Estimation	Joint Completion	Motion Prediction	Motion In-Between	Mesh Recovery	Future Mesh Recovery	Joint Completion	Motion Prediction	Motion In-Between
			Pose (MPJPE ↓)					Mesh (MPVE ↓)				
AMASS	4	36.14M	28.48	35.71	44.51	29.37	21.09	36.81	46.05	54.39	43.54	32.81
	8	46.67M	24.96	34.06	41.93	24.98	17.23	38.03	48.98	55.90	44.42	34.44
	10	51.74M	26.15	33.40	42.22	26.66	18.34	36.91	45.46	55.04	44.47	33.58
3DPW	4	36.14M	105.61	120.62	61.51	63.42	25.52	129.01	151.38	87.93	92.93	36.15
	8	46.67M	96.36	111.93	56.71	55.97	23.00	119.45	141.95	82.60	84.44	35.28
	10	51.74M	98.63	114.24	60.16	61.30	25.73	117.73	141.52	84.61	86.35	38.44

例如未来姿态估计和网格恢复。这些特征的加权和最终导致了所需的输出，展示了 X-Fusion Net 在处理多样的输入提示和任务时的灵活性。此外，这些结果展示了在多个层级聚合上下文特征并通过动态权衡不同层级的贡献来更新它们的必要性。

5.3 消融研究

5.3.1 提示策略

采样方法。表 7 和图 8 显示了关于采样方法的定量和定性消融研究。除了获得更好的定量结果外，SPS 通过视觉比较更有效地捕获了整体数据分布。FPS 由于在每次迭代中仅采样最远点而被限制在外围区域。随机采样忽略了在稀疏区域中具有显著较低概率的数据，这些数据对于域外泛化是必要的。此外，我们还将 SPS 与基于聚类的采样方法进行比较，在该方法中，我们使用 k-means 聚类运动序列，并使用聚类中心作为锚点。如表所示，SPS 优于基于聚类的采样方法。更好地反映整体数据分布有助于提升 HiC 在域内和域外的泛化能力。

锚的数量。表 5 显示了通过最大-最小相似性提示采样获得的锚数量的消融研究。一般来说，增加锚的数量会带来更好的性能，但随着数量进一步增加，在几个任务中收益开始减少。最佳锚数量（800）确保了泛化效果和计算效率，同时不会有过拟合的风险。

软锚。表 6 通过比较有软锚和无软锚的模型展示了一项关于软锚的消融研究。结合软锚一致地在域内和域外的推广性能方面表现更好。

5.3.2 网络架构

多级上下文聚合。图 **??** 通过比较不同级别的数量和不同聚合功能的实现进行了一项关于多级上下文聚合的消融研究。我们定量分析了状态空间模型 (M)、自注意力机制 (T) 和图卷积 (G) 如何对模型性能做出贡献。每个柱状图代表一个特定的模块组合：M+G、M+T、G+T、M（仅状态空间模型）、G（仅图卷积）和 T（仅自注意力）。从默认架构 (M+G+T) 中移除模块持续导致性能下降，突显出不同聚合方式的互补性。除去图 7，图 **??** 进一步验证了聚合多级上下文特征的必要性。

跨级别上下文更新。表 8 显示了关于跨级别上下文更新的消融研究。具体来说，我们在 X-Fusion Net 中比较了静态和动态上下文更新在不同级别的有效性。静态上下文更新方法对所有级别的输出进行平均，而动态上下文更新方法动态地权衡每个级别的贡献。结果显示，在所有任务和数据集上，动态方法始终优于静态方法。

特征维度。图 **??** 展示了关于特征维度的消融研究。模型在 128 个特征维度时表现最佳，因为此时特征表达力足够，而不会使表示过于复杂。更大的维度会导致过多参数化、效率低下和泛化能力下降。同时，减少维度也会导致准确性下降，因为这不足以捕捉复杂的上下文依赖关系。

层数。表 9 展示了关于模型层数的消融研究。增加层数可以提升多个任务的结果，但也需要更大的参数规模。在权衡效果和效率之后，我们使用 8 层来实现拟议的 X-Fusion Net 在 Human-in-Context 中。

6 未来工作

更广泛的领域范围。跨域的统一 3D 人类运动建模的探索可以扩展到更广泛的领域范围。关于模式，多模态数据包括 RGB 视频和点云序列可以被使用，从而提供对人类外观和深度细节的更全面视角，这可以通过提供更充足和丰富的上下文信息来使上下文学习受益。关于任务，未来的工作可以研究结合更复杂的任务，例如动作识别、人物再识别和分割，这将进一步增加模型的多功能性。此外，更多的数据集可以改善对数据规模和运动多样性的泛化能力。来自不同上下文背景的数据集可以进一步增强模型的鲁棒性，使其能够应用于更广泛的现实应用场景，如机器人技术和增强现实。

效率和可扩展性。除了扩大范围，未来的工作还应着重于提高模型的效率和可扩展性。通过使用更高效的注意力机制或者网络剪枝技术等方法来减少计算复杂性和内存使用，可以帮助在资源受限的环境中部署跨域模型。此外，推进迁移学习技术可以使模型更好地从较少的标注数据中进行泛化，加速训练过程，使模型更易于实际应用。

高级提示工程。随着模型变得更加多模态（处理 3D 和 2D 数据），将需要能够有效融合多种数据源的提示。未来的工作可能会生成整合 3D、视觉甚至文本线索的提示，以启用更强大且具情境感知的模型。随着模型变得更加复杂，理解提示如何影响模型的决策过程将变得至关重要。未来的工作可以探