ACTLLM: 动作一致性调优的大语言模型

Jing Bi[1], Lianggong Bruce Wen[2], Zhang Liu[2], Chenliang Xu[1]

Abstract— **本文介绍了 ACTLLM（动作一致性调整的大型语言模型），这是一种针对动态环境中的机器人操作的新方法。传统的基于视觉的系统往往难以学习在任务执行和空间推理方面都表现出色的视觉表示，从而限制了它们在动态环境中的适应性。ACTLLM 通过利用语言来构建结构化的场景描述符，为灵活的语言指令下的空间理解和任务性能提供了一个统一的接口，以此解决这些挑战。此外，我们引入了一种新的动作一致性约束，使视觉感知与相应的动作对齐，从而增强可操作视觉表示的学习。另外，我们将操纵任务的马尔可夫决策过程重新定义为一个多轮次的视觉对话框架。这种方法能够通过从任务执行历史中得出增强的情境相关性来建模长期的任务执行。在我们的评估过程中，ACTLLM 在多种场景下表现出色，证明了其在具有挑战性的基于视觉的机器人操作任务中的有效性。**

I. 介绍

适应多样且动态的环境，同时执行灵活的任务规范，仍然是机器人操作方法中的一个重大挑战。基于语言的视觉操作系统通过利用丰富的视觉信息和语言来更好地理解不同的环境条件和任务规范，提供了一个有希望的解决方案 [1], [2], [3], [4]。这些系统通常包含两个关键组成部分：一个将语言指令中的概念与视觉信息关联的视觉组件，以及一个基于视觉组件的输出来生成动作的策略模块。最近的端到端框架 [2], [5], [6], [7], [8] 试图利用可供性将语义含义与视觉信息集成。这种集成有助于机器人识别特定物理环境中的可行动作，通过将语言背景与视觉线索合并来解决哪些动作是可能的以及它们可以在给定场景中执行的位置的问题。与此相反，另一条研究方向，以 CALVIN[4] 为例，倡导向适应性强、任务无关的操作策略转变，这些策略采用通用、非结构化的语言来定义任务。这种方法允许更通用的策略，因为广泛的语言规范能够实现任务执行的多功能性。

然而，现有的研究经常将视觉模型和行动策略的学习过程分开，每个组件独立优化。这种划分通常导致感知模型提供的表示并不适合政策模型有效使用。此外，行动策略面临学习广泛技能的挑战。这种复杂性源于解释开放语言指令以及有效地利用视觉表示的必要性，因此需要更复杂和集成的学习方法。因此，实现将视觉观察与语言概念关联起来以适应机器人操作的更灵活策略面临着两阶段的挑战：(i) 增强视觉与语言概念之间的对齐。(ii) 优化两种模态的集成信息以制定行动策略。

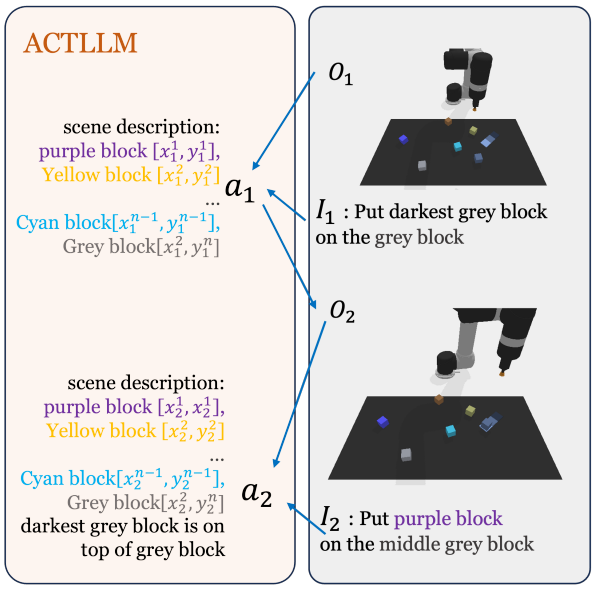

Fig. 1. 我们将用于操作任务的马尔可夫决策过程重新表述为一个多回合视觉对话框架，在此框架中，模型根据给定指令和历史观察生成当前视图以及潜在未来状态的描述。从这些描述中生成动作以完成任务。这种方法确保了在场景描述之间生成的动作与场景变化保持一致，使大型语言模型（LLMs）能够分析操作轨迹中的空间和时间关系。这有助于复杂任务的有效执行。

由于基础模型的强大零样本性能，一些研究将其集成到机器人应用中以克服第一个挑战。这些方法涵盖了从生成自然语言场景描述 [9]、利用预训练表示进行反应控制 [10]、将动作与符号表征关联以适应测试时间 [11], [12], [13] 到自动化密集奖励生成 [14], [15], [16], [17], [14] 等各种任务。此外，一些方法调整了基础模型架构来应对第二个挑战。例如，ReAct [18] 规划器生成条件序列来指导低级策略动作。PaLM-E [6] 集

This work is a work in progress.

[1]University of Rochester, Rochester, NY, USA. Email: jing.bi@rochester.edu, chenliang.xu@rochester.edu

[2]Corning Inc., Corning, NY, USA. Email: {wenl, liuzh3}@corning.com

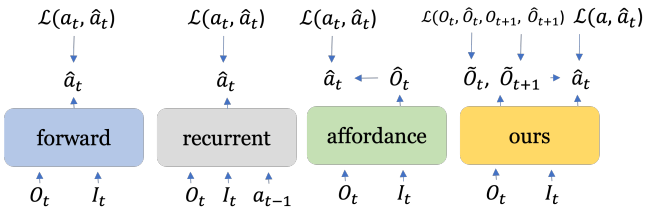

Fig. 2. 与之前的模型相比：(a) 传统前向模型使用当前的观测和指令来输出动作，可能通过额外的历史信息进行增强。(b) 可用性方法则不同，它们将观测转化为热图以通过 argmax 进行简单的动作计算。(c) 我们的方法通过从大语言模型生成动作，利用基于文本的场景描述来规范化并提高动作嵌入的准确性，从而推进了这些概念。

成了多模态输入——包括视觉、文本和状态估计——以产生低级指令性文本来驱动控制器。然而，先前的方法通常将任务规范和环境理解视为不同的挑战，这种分离可能妨碍了视觉概念与其语义意义的整合，从而为有效指导机器人制造障碍。此外，视觉观察经常被处理成隐藏表征，这不仅难以解释，而且对于策略优化也颇具挑战性。

为克服这些挑战，我们引入了 ACTLLM 方法，该方法统一了视觉信息的解释与策略学习。我们的方法利用结构化的场景描述，这不仅对人类更具可访问性，还允许我们构建新的损失函数以更有效地优化模型。通过纳入动作一致性约束损失，我们将动作策略与场景描述生成联合优化，显著提升了这些元素的融合，并解决了上述问题。此外，遵循所提出的方法，我们能够将操作任务的马尔科夫决策过程重新表述为一个多轮视觉对话框架，以帮助长期任务学习。总结而言，我们的贡献在于三个方面：

结构化场景描述：通过将观察结果明确表示为结构化的场景描述，我们可以将空间推理与指令理解相结合，创建可操作的表征，融合语言理解和视觉特征。

新颖的政策优化方法：建立在结构化场景描述的基础上，我们提出了一种新的损失函数。这增强了来自文本指令和视觉观察的信息整合，在动态环境中优化策略学习。

增强大型语言模型的调优以进行操控 ACTLLM 将传统的马尔可夫决策过程转换为可视化的多轮对话框架。这使得能够采用一种新颖的方法来增强大语言模型的微调，通过整合操纵控制数据来优化决策过程。

II. 相关工作

A. 操作愿景

传统的机器人感知方法主要依赖于明确的“对象”表示，例如实例分割、对象类别和姿态 [19], [20]。然而，这些方法在处理诸如衣物和豆类等可变形和颗粒状物品时遇到挑战，因为使用几何模型或分割来表征它们非常困难。为了克服这些限制，近期的一些方法 [21], [22] 开始对对象和任务做出更少的假设，通常将问题表述为图像到动作预测任务。然而，直接在 RGB 图像上训练具有 6 自由度（6-DoF）的任务往往效率低下，通常需要大量的演示或阶段来获取基本技能如物体重新排列。针对这些挑战，出现了一些方法。例如，在 3D 环境中，C2FARM [23] 提出了一种以动作为中心的强化学习 (RL) 代理，其后端是一个粗到细粒度的 3D-UNet。然而，这种方法在最精细级别上的感受野有限，无法涵盖整个场景。另一条研究路线，例如像 [24], [2], [25] 这样的方法，则专注于学习带有能力的动作中心表示，强调对象交互方面的内容。与这些方法不同的是，我们的模型仅依赖于 RGB 图像，并且不需要像以前的工作那样对“物体”进行特殊建模。

B. 语言赋能的机器人技术

基于指令的策略一直是机器人领域的一个热门研究方向 [26], [27], [28]。一个值得注意的发展是 CLIPORT [29]，它通过 CLIP 文本编码增强了 Transporter [24] 的语义理解和对象操作能力 [30]。这一概念进一步扩展到了 Perceiver-Actor 模型的三维环境，利用了体素化的观察和三维动作空间 [31]。LLMs 在机器人中的另一个创新应用涉及为行动策略生成代码 [32]，使机器人能够使用视觉 API 完成如分割、检测和互联网访问等任务。Instruct2Act [9] 展示了这一点，通过将机器人技能与大型语言模型相结合，在灵活性和专业性之间取得了平衡，并在零样本设置中表现出高绩效。语言在高级机器人规划 [33], [34], [35] 和低级策略开发中也扮演着关键角色，基于模型的方法正在获得关注 [36] 。值得注意的是，CaP [32] 和苏格拉底模型 [37] 分别通过生成详细的策略代码和将感知数据纳入大型语言模型中取得了显著进展。此外，大型语言模型正被用作机器人系统中的奖励或反馈来源，如 [34] 和 [38] 的方法所示，这些方法通过闭环反馈和强化学习增强了机器人的操作。PAFF [11] 通过利用基础模型的反馈，采用回顾经验回放过程来应对这

一挑战，使模型能够响应陌生环境中随机生成的指令，并随后收集和重新标记这些动作以微调基础模型进行测试时适应。

VIMA [1] 提出了一种视觉运动注意力代理，使用 Transformer 编码器-解码器架构解决来自多模态提示的机器人操作问题。

III. 方法

A. 初步的

我们有权访问一个数据集 $\mathscr{D} = \{\zeta_1, \zeta_2, \ldots, \zeta_n\}$，其中包含 n 个专家演示轨迹 ζ 和相关的离散时间输入-动作对 $\zeta_i = \{(o_1, I_1, a_1), (o_2, I_2, a_2), \ldots\}$。值得注意的是，每一步的指令 I 可能仅由文本组成、仅由图像组成或如图所示的两者的组合 3。动作空间包括原始运动技能，其中包括诸如拾取和放置、清除、推送 [24] 等动作。此外，对于每个动作，两个位置向量表示初始和目标姿态，标记为 $(p_{\text{initial}}, p_{\text{target}})$。

我们的目标是学习一个策略模型 π，能够有效地生成一个动作 a_t 以完成基于当前执行历史的多模态指令 I。受到近期模仿学习进展 [39] 的启发，我们设计了我们的策略为 $\hat{a} = \pi_\theta(x_t, x_{t+1})$。这使我们能够推断出最有可能的动作，该动作将代理从其当前状态过渡到后续状态，其中 x_t 和 x_{t+1} 代表从观察中提取的状态。对于一步轨迹，表示为 (o_t, a_t, o_{t+1}, I_t)，我们将通过观察模型将观测映射到各自的状态表示，得到 $x_t = \mathscr{M}(o_t), x_{t+1} = \mathscr{M}(o_{t+1})$。在推理过程中，为了使策略能够利用未来状态 x_{t+1}，我们需要结合一个前向模型来预测给定当前观测和指令的未来状态，形式化为 $\tilde{x}_{t+1} = \mathscr{F}_\beta(o_t, I_t)$。这种表述的本质在于关注期望的最终状态而非达到它的具体步骤，因为多条路径可以导致相同的结局。在实践中，这种方法意味着只要提出的替代动作能导致相同的预期后续状态，模型就不会严格惩罚偏离真实动作的行为。

B. 结构场景描述

在优化我们的模型时，关键在于为 $\mathscr{M}$ 和预测 $\mathscr{F}$ 模型选择状态表示。

我们利用 LLM 作为模型 $\mathscr{M}$ 和函数 $\mathscr{F}$ 的关键见解，这不仅增强了模型优化，而且具有稳定的真实值，因为先前使用隐藏向量进行状态表示的方法可能不稳定，尤其是在训练的早期阶段。

为了使 LLM 提供结构化表示而非非结构化的语言，我们利用预定义的 JSON 模式来操控解码阶段。在此阶段中，我们预先填充数据模式的固定标记，并将内容标记的生成完全委托给语言模型。这种方法确保了结构的一致性，而模型动态地填写内容。因此，我们将状态表示定义为一组 $\{(\mathfrak{o}_1, \mathfrak{c}_1, p_1), (\mathfrak{o}_2, \mathfrak{c}_2, p_2), (\mathfrak{o}_3, \mathfrak{c}_3, p_3), \ldots\}$，其中 $\mathfrak{o}$、$\mathfrak{c}$ 和 p 分别代表对象、颜色和坐标。

此设计提供了若干优点：(i) 可解释性：这种方法简化了对模型决策过程的理解和验证，使某些结论是如何以及为什么得出的更加透明。(ii) 清晰性：我们可以基于状态表示制定一个清晰的损失函数，而不是依赖与隐藏状态相关的模糊回归损失。(iii) 稳定的地面真相：通过利用对模拟器内部状态的访问，我们可以精确地获得所有对象的状态。为了使大语言模型能够使用坐标描述场景，我们采用边界框的中心位置来表示对象的位置。坐标在 [0,1] 范围内进行归一化，并保留两位小数；[0.50,0.50] 表示对象位于视图的中心。由于我们希望扩大大语言模型的词汇空间，我们将 Adapter 机制 [40] 整合进来以建模空间标记。一旦有了场景描述，我们可以计算损失为 $\mathscr{L}(x_t, x_{t+1}, \tilde{x}_{t+1}, \tilde{x}_t)$

我们采用了来自 GLIP [41] 的方法，使用对比损失来比较预测对象与地面真实文本标记。这包括直接匹配预测对象及其相应的文本描述符。该过程包括：(i) 对于每个预测对象（查询），我们计算与文本描述的特征向量的点积，以生成对应于每个文本标记的对数几率，然后对每个对数几率应用焦点损失。(ii) 由于对象集缺乏内在顺序，我们使用边界框回归和分类成本来在预测与真实数据之间执行二分图匹配。(iii) 最后，我们计算匹配预测及其相应真实值的损失值，确保精确对齐和准确性。

通过将对象属性及其位置与语言标记对齐，我们可以提高模型在场景预测任务中的可靠性和准确性。

C. 动作一致性损失

我们现在专注于如何根据状态表示生成动作 $a_t = \pi_\theta(x_t, x_{t+1})$。将另一个语言模型连接起来，基于生成的或真实的状态表示来预测动作似乎是直观的。然而，这可能会带来更多的缺点而不是优点。主要是因为在初始训练阶段场景描述可能存在不准确，导致策略梯度更新中的噪声。因此，我们提出通过使用可学习参数 W 和注意力模块聚合状态表示的标记嵌入序列来构

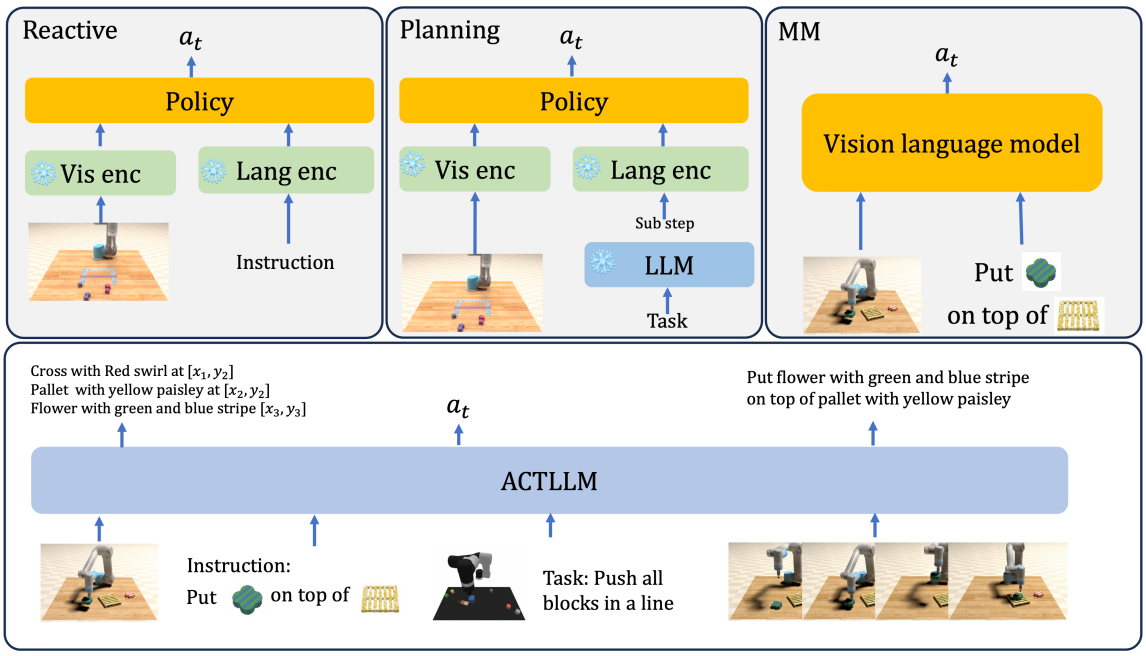

Fig. 3. 我们视觉语言操作方法与现有解决方案的比较：**反应性的**表示利用基础模型提取特征以生成后续动作 [10]。相比之下，**规划**策略通常采用大语言模型将任务分解为子步骤并解决它们。MM 意味着直接对视觉语言模型进行微调用于机器人操作，正如在 [1], [3] 直接微调一个视觉语言模型用于机器人操作所示。我们的模型可以处理作为特定任务输入的视觉和语言令牌，使我们能够生成更加一致且准确对应场景描述的动作。

建策略

$$Q = EW, \quad x_{\text{agg}} = \text{softmax}(QQ^T)Q$$
$$\hat{a}_t = \text{MLP}(x_{\text{agg}})$$

其中 E 表示 x_t, x_{t+1}、$E = [e_1, e_2, \ldots, e_n]$ 的标记嵌入序列，而 n 是该序列中的标记数量。通过这种方法，动作是从一个聚合的标记中得出的，这个标记巧妙地封装了状态的句子级表示。此外，由于其轻量设计，模型表现出更好的适应性，这对于平稳容纳随时间改进的状态表示至关重要。

一旦我们有一个步骤元组 (s_t, a_t, s_{t+1}, I_t)，我们可以设计优化函数来使动作与状态转换对齐。训练的联合目标形式化为：

$$\begin{aligned} \min_{\theta,\beta} \; & \mathscr{L}(x_t, x_{t+1}, \tilde{x}_{t+1}, \tilde{x}_t) + \mathscr{L}(a_t, \hat{a}_t), \\ \text{s.t.} \quad & \tilde{x}_t, \tilde{x}_{t+1} = \mathscr{F}_\beta(o_t, I_t) \\ & \hat{a}_t = \pi_\theta(x_t, x_{t+1}) \end{aligned} \tag{1}$$

其中动作损失包括原始技能的分类损失和场景描述的损失。如图 4所示，动作一致性损失有效地利用了额外信息以更准确地将动作和视觉数据对齐在一起。这种动作生成方法的优势在于它紧密集成了动作与场景描述，确保动作更加密切地关联于任务和场景内的上下文。此外，通过采用如下的多轮调优，动作得益于其对应场景变化的特性。因此，场景中的变化也影响了动作的生成方式。

D. 马尔可夫决策制定作为可视对话之一

最近的进步，如在 vicuna fintuning[42] 中所强调的那样，表明多轮互动显著增强了大型语言模型（LLMs），使它们能够进行更复杂且富含上下文的对话。与单轮对话不同，单轮对话生成响应时不考虑过去的交互，而多轮对话则依赖于之前的交流，使模型能够全面理解上下文和用户的目标。

在之前介绍的概念基础上，我们提出将我们的单步优化扩展为如图 1所示的动作可变长度序列。该框架涉及多轮对话调优，其中每一步包括当前观察和指令，从而导致一个独特状态。模型需要根据当前状态和整体任务目标选择最合适的动作，并结合预测状态的历史背景。

联合损失在每个时间步计算，并在整个轨迹上与动作预测损失一起进行优化。带有特征空间动态的最

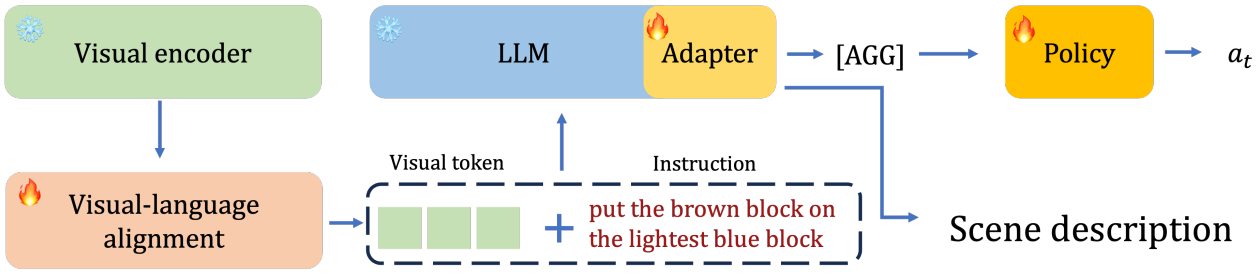

Fig. 4. 我们的模型的说明。我们的模型由三个主要组件组成：一个视觉编码器，一个视觉-语言对齐层和一个仅解码器 LLM。边界框的坐标被转换成特定格式的文本。在训练过程中，我们冻结了视觉编码器和 LLM，并且只更新适配器和对齐层。接收到指令后，LLM 最初生成一个聚合标记。这个标记告知策略为这一步骤生成相应的动作并有助于环境描述。

终多步骤目标定义如下：

$$\min_{\theta,\beta} \sum_{t=1}^{T-1} \Big(\mathscr{L}(x_t, x_{t+1}, \tilde{x}_{t+1}, \tilde{x}_t) + \mathscr{L}(a_t, \hat{a}_t) \Big), \quad \text{s.t.} \quad \tilde{x}_t, \tilde{x}_{t+1} = \mathscr{F}_\beta(o_t, I_t) \quad \hat{a}_t = \pi_\theta(x_t, x_{t+1}) \tag{2}$$

该方法允许模型随着时间学习和适应，培养出更高级且以用户为中心的对话体验。它使用户能够在任务之间切换或提供即时干预，增强互动质量和效果。如图3所示，与其它方法相比，我们提出的模型可以处理各种类型的信息并为机器人操作生成关键输出。轨迹具有长上下文的优势在于能够确保动作与场景描述之间的更好对齐。具体来说，在调整步元组 (s_t, a_t, s_{t+1}, I_t) 时，动作和场景描述紧密关联于当前状态。然而，在优化多轮交互时，动作生成可以利用扩展的上下文作为提示。这允许采取考虑更全面历史的动作。

E. 实现细节

我们的模型架构由两个组件组成 (i) 多模态大模型既作为 $\mathscr{F}$ 也作为 $\mathscr{M}$。该组件包含一个视觉编码器 Φ_V、一个投影层 Φ_P 和一个语言模型 Φ_L。(ii) 政策头 π，其中包括一个注意力模块和一个多层感知器。视觉编码器用于将指令或观察中发现的所有图像转换为一系列视觉标记，表示为 $Z_V = \Phi_V(I)$。为了适应这一表示以便在 LLM 中使用，采用了线性层 Φ_P，将 Z_V 转换到 LLM 的输入空间，从而得到 $Z_T = \Phi_P(Z_V)$。随后，Z_T 和 Q_T 被连接并通过 Φ_L 生成场景描述 x_t, x_{t+1}。最后，策略头 π 处理场景描述标记并将它们翻译成动作。正如在 [43] 中讨论的那样，即使使用一个适度的视觉指令调优数据集对视觉编码器进行微调也会导致语义损失。这种损失会不利地影响视觉编码器的图像表示能力。因此，我们选择保持视觉编码器冻结，特别是考虑到模拟可能会加剧语义损失。我们将 LLAMA3-8B [44] 模型作为我们的大语言模型，特别调优以遵循指令。为了将视觉标记与静态的大语言模型相连接，我们使用了一个对齐层。这由一个动作投影模块补充，该模块对于将模型见解转换为可执行输出至关重要。选择了 AdamW 作为优化器。更多细节将在补充材料中提供。

IV. 实验

我们的实验集中在两个基准测试 (i) VIMA-BENCH：一个新的任务套件和基准测试，旨在通过多模态提示促进通用机器人操作的学习。(ii) 克莱港口：该数据集提供了与自上而下的 RGB-D 观察配对的步骤级文本说明，作为学习多任务操作的平台。我们精心设计评估来评价 (i) 组成泛化，评估模型处理具有新形状、颜色以及完全新颖的对象的能力。(ii) 上下文泛化，其中模型通过新任务进行评估 请注意，在测试阶段，我们不提供状态表示作为模型的输入。因此，我们不需要依赖任何对象检测器来提供结构化的屏幕描述，因为它是由模型生成的。

A. 组合泛化

组合泛化评估模型将学到的知识应用于熟悉元素或概念的新组合的能力。我们在 CLIPORT 和 VIMA-BENCH 环境中展示了这一点。

在 CLIPORT 环境中，我们遵循 PAFF 框架 [11]。我们在 10 个不同的场景中报告了 100 个评估实例的任务成功率。这些场景包含各种方块、物体和碗，从而测试模型准确放置物体的能力。我们采用了以下两种评估协议：1) 打包未见过的对象：在此协议中，我们训练一个策略将不同形状的物体打包进一个棕色盒子 ('pack-shapes')，然后评估其对'pack-unseen-objects' 的能力。2) 将形状放入碗中 另一项评估涉及将不同颜色的积木放入不同颜色的碗中（“把积木放进碗里”），然后我们要求模型将不同形状的物体放入不同颜色的

碗中。我们包含了 PAFF 中的所有基线模型，其中 MdetrORT [45] 是通过替换视觉和语言编码器而形成的 CLIPORT [2] 的一种变体，并且 AugORT [46] 包含了更多的数据增强到 MdetrORT 中。

TABLE I

在 CLIPORT 平台上进行的组成和分布外泛化评估的结果被展示出来。主要的评价指标是成功率。每个步骤都会在左侧列提供一个新的指令；只有在前一个指令成功执行后才会发出后续指令。

Method	将形状放入碗中		打包未见对象	
CLIPORT	28.0%	16.8%	58.9%	46.1%
MdetrORT	33.8%	17.8%	62.0%	48.4%
AugORT	34.4%	18.9%	63.1%	49.0%
PAFF	51.0%	35.0%	72.8%	63.8%
ACTLLM	64.0%	66.2%	85.8%	79.6%

我们在表 I中展示了组合泛化的评估结果。我们的方法在两种评估协议中都显著超越了基线，显示出我们场景表示模型在实现卓越的组合泛化方面的有效性。与 PAFF 不同，ACTLLM 采取了一种全面的方法，不仅仅关注任务相关的对象；它考虑了所有存在的对象。这种更广泛的视角使它能够在涉及新颖物体和形状的任务中表现出色。此外，ACTLLM 可以准确地生成包含各种概念（如物体和容器）的场景描述，在组合设置下实现这一点。另外，它可以学习感知物体的表示而不需特定的对象及其相关的边界框。

在 VIMA-BENCH 中，组合泛化的评估更具挑战性，可以分为三个等级：1) **放置：**在训练过程中，所有提示都会出现，而在测试时仅随机化桌面上物体的放置位置。2) **组合的：**在训练阶段，所有纹理和对象都是熟悉的，但在测试阶段，这些元素的新组合被引入。3) **新对象：**测试提示和模拟工作区都包含了训练过程中未遇到的新纹理和对象。我们包含了 Gato [47]、Flamingo [48] 和 GPT [1] 等模型的结果，这些结果直接来自 VIMA 论文。各种评估协议级别的评估结果呈现在表 II中。由于空间有限，我们仅报告那些不同方法显示出显著性能差异的代表性任务的成功率。平均值（Avg）表示特定评估级别下所有任务的成功率。我们可以看到，在前三个需要学习熟悉元素或概念的新组合的等级中，我们的方法优于所有基线方法。

B. 上下文泛化

此挑战在 VIMA 中首次引入，其中新任务通过新颖的提示模板在测试阶段定义。这些模板不仅包括新的动作，还包括训练数据中未出现的新对象。我们的评估采用两步方法：首先，我们遵循 VIMA 协议，直接执行第 4 级任务，如表 II所示；其次，我们在测试过程中使用情景演示视频传达任务的本质。在测试阶段包含情景学习旨在考察模型的零样本适应能力。这种方法强调了为模型提供动态、现实世界情境的重要性，以增强其学习和适应过程。遵循 VIMA，我们保存 T9: 扭曲 T10: 跟随动作 T11: 跟随顺序作为新任务。然后我们进行了一项消融研究来评估各个模块的有效性。

如表 III所示，移除任务调优显著降低了所有任务的性能，导致上下文学习的准确性明显下降。这表明模型难以解释来自视频的任务，这是可以预见的结果，因为训练主要包含多模态提示而不是视频内容。这样的发现强调了任务调优在提升模型泛化能力方面的重要作用。未来状态预测模块的缺失严重影响了性能，特别是在任务 T10 中，突显了模型预测能力对于有效行动规划的重要性。此外，移除多轮调优也降低了有效性，尽管程度不如缺乏未来状态预测严重，这可能起到了正则化机制的作用。

V. 结论、未来工作和局限性

我们的方法，ACTLLM，在基准数据集上通过引入动作一致性损失展现了优越的性能，显著提高了组合性和零样本泛化能力。我们的研究推动了智能机器人系统的发展，这些系统能够直观地理解和执行人类语言命令，改善人机交互。未来的研究可以探索实际应用和情境学习能力，扩大智能机器人系统的部署范围。

虽然我们的模型显示出有希望的潜力，但仍存在一个关键挑战：二维图像往往无法捕捉到有效机器人控制所需的精确三维位置和期望状态。

TABLE II

我们使用 VIMA-BENCH 评估在四个不同级别上对我们的方法进行了比较分析。‘Avg’ 表示每个级别内所有任务的平均成功率。为了确定每种方法的成功率，我们从每个任务中抽取了 200 个片段。我们的方法相对于基线方法显示出显著改进。

	L1					L2					L3					L4	
Method	Avg	T5	T9	T16	T17	Avg	T5	T9	T16	T17	Avg	T5	T9	T16	T17	Avg	T10
Gato	57.0	44.5	14.0	43.0	1.5	53.9	46.0	10.5	42.0	1.0	45.6	36.0	17.0	41.5	0.0	13.5	0.0
Flamingo	47.2	41.0	3.0	38.0	2.0	47.1	43.0	4.5	40.0	1.0	42.1	36.5	6.0	45.5	0.5	11.1	0.0
GPT	47.9	45.0	8.0	33.0	1.0	47.4	43.0	10.5	34.0	3.0	42.6	32.0	5.0	37.5	0.0	12.1	0.5
VIMA	87.2	65.0	13.5	88.0	77.0	87.0	61.0	12.5	87.5	77.5	84.0	63.0	12.0	58.5	78.0	49.6	0.0
主动语言模型	90.5	78.3	65	96.0	86.0	90.9	78.2	61.5	93.0	84.3	93.4	84.0	70.9	88.2	87.0	64.8	12.1

TABLE III

评估 ACTLLM 在上下文中的泛化能力，将扭曲和遵循顺序作为新的测试任务。

Task	T9	T10	T11	Overall
Our Method	20.1%	63.4%	10.0%	36.2%
w/o task tuning	14.5%	38.9 %	6%	26.7%
w/o future state	12.4%	23.9 %	4.7%	22.3%
w/o multi-turn	16.1%	29.9 %	3.5%	24.3%

References

[1] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu, and L. Fan, “Vima: General robot manipulation with multimodal prompts,” arXiv preprint arXiv:2210.03094, 2022.

[2] M. Shridhar, L. Manuelli, and D. Fox, “Cliport: What and where pathways for robotic manipulation,” in CoRL. PMLR, 2022, pp. 894–906.

[3] J. Li, Q. Gao, M. Johnston, X. Gao, X. He, S. Shakiah, H. Shi, R. Ghanadan, and W. Y. Wang, “Mastering robot manipulation with multimodal prompts through pretraining and multi-task fine-tuning,” arXiv preprint arXiv:2310.09676, 2023.

[4] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard, “Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks,” IEEE Robotics and Automation Letters, vol. 7, no. 3, pp. 7327–7334, 2022.

[5] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, et al., “Do as I can, not as I say: Grounding language in robotic affordances,” arXiv preprint arXiv:2204.01691, 2022.

[6] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu, et al., “Palm-e: An embodied multimodal language model,” arXiv preprint arXiv:2303.03378, 2023.

[7] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al., “Rt-1: Robotics transformer for real-world control at scale,” arXiv preprint arXiv:2212.06817, 2022.

[8] S. Tellex, T. Kollar, S. Dickerson, M. Walter, A. Banerjee, S. Teller, and N. Roy, “Understanding natural language commands for robotic navigation and mobile manipulation,” in Proceedings of the AAAI Conference on Artificial Intelligence, vol. 25, no. 1, 2011, pp. 1507–1514.

[9] S. Huang, Z. Jiang, H. Dong, Y. Qiao, P. Gao, and H. Li, “Instruct2act: Mapping multi-modality instructions to robotic actions with large language model,” arXiv preprint arXiv:2305.11176, 2023.

[10] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, “R3m: A universal visual representation for robot manipulation,” arXiv preprint arXiv:2203.12601, 2022.

[11] Y. Ge, A. Macaluso, L. E. Li, P. Luo, and X. Wang, “Policy adaptation from foundation model feedback,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 19 059–19 069.

[12] R. Wang, J. Mao, J. Hsu, H. Zhao, J. Wu, and Y. Gao, “Programmatically grounded, compositionally generalizable robotic manipulation,” arXiv preprint arXiv:2304.13826, 2023.

[13] J. Zhang, K. Pertsch, J. Zhang, and J. J. Lim, “Sprint: Scalable policy pre-training via language instruction relabeling,” arXiv preprint arXiv:2306.11886, 2023.

[14] Y. J. Ma, W. Liang, V. Som, V. Kumar, A. Zhang, O. Bastani, and D. Jayaraman, “Liv: Language-image representations and rewards for robotic control,” arXiv preprint arXiv:2306.00958, 2023.

[15] M. Kwon, S. M. Xie, K. Bullard, and D. Sadigh, “Reward design with language models,” in The 11th International Conference on Learning Representations, 2023.

[16] W. Yu, N. Gileadi, C. Fu, S. Kirmani, K.-H. Lee, M. G. Arenas, H.-T. L. Chiang, T. Erez, L. Hasenclever, J. Humplik, et al., “Language to rewards for robotic skill synthesis,” arXiv preprint arXiv:2306.08647, 2023.

[17] E. Bıyık, D. P. Losey, M. Palan, N. C. Landolfi, G. Shevchuk, and D. Sadigh, “Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences,” The International Journal of Robotics Research, vol. 41, no. 1, pp. 45–67, 2022.

[18] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, “React: Synergizing reasoning and acting in language models,” arXiv preprint arXiv:2210.03629, 2022.

[19] X. Deng, Y. Xiang, A. Mousavian, C. Eppner, T. Bretl, and D. Fox, “Self-supervised 6d object pose estimation for robot manipulation,” in 2020 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2020, pp. 3665–3671.

[20] C. Xie, Y. Xiang, A. Mousavian, and D. Fox, “The best of both modes: Separately leveraging rgb and depth for unseen object

instance segmentation," in Conference on robot learning. PMLR, 2020, pp. 1369–1378.

[21] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, et al., "Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation," arXiv preprint arXiv:1806.10293, 2018.

[22] K. Kang, C. Xie, C. He, M. Yi, M. Gu, Z. Chen, K. Zhou, and H. Wu, "Learning efficient illumination multiplexing for joint capture of reflectance and shape." ACM Trans. Graph., vol. 38, no. 6, pp. 165–1, 2019.

[23] S. James, K. Wada, T. Laidlow, and A. J. Davison, "Coarse-to-fine q-attention: Efficient learning for visual robotic manipulation via discretisation," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 13 739–13 748.

[24] A. Zeng, P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, et al., "Transporter networks: Rearranging the visual world for robotic manipulation," in Conference on Robot Learning. PMLR, 2021, pp. 726–747.

[25] E. Stengel-Eskin, A. Hundt, Z. He, A. Murali, N. Gopalan, M. Gombolay, and G. Hager, "Guiding multi-step rearrangement tasks with natural language instructions," in Conference on Robot Learning. PMLR, 2022, pp. 1486–1501.

[26] A. Brohan, Y. Chebotar, C. Finn, K. Hausman, A. Herzog, D. Ho, J. Ibarz, A. Irpan, E. Jang, R. Julian, et al., "Do as i can, not as i say: Grounding language in robotic affordances," in Conference on Robot Learning. PMLR, 2023, pp. 287–318.

[27] L. Shao, T. Migimatsu, Q. Zhang, K. Yang, and J. Bohg, "Concept2robot: Learning manipulation concepts from instructions and human demonstrations," The International Journal of Robotics Research, vol. 40, no. 12-14, pp. 1419–1434, 2021.

[28] M. Shridhar and D. Hsu, "Interactive visual grounding of referring expressions for human-robot interaction," arXiv preprint arXiv:1806.03831, 2018.

[29] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," in Conference on Robot Learning. PMLR, 2022, pp. 894–906.

[30] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al., "Learning transferable visual models from natural language supervision," in International conference on machine learning. PMLR, 2021, pp. 8748–8763.

[31] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," in Conference on Robot Learning. PMLR, 2023, pp. 785–799.

[32] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," arXiv preprint arXiv:2209.07753, 2022.

[33] W. Huang, P. Abbeel, D. Pathak, and I. Mordatch, "Language models as zero-shot planners: Extracting actionable knowledge for embodied agents," in International Conference on Machine Learning. PMLR, 2022, pp. 9118–9147.

[34] W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar, et al., "Inner monologue: Embodied reasoning through planning with language models," arXiv preprint arXiv:2207.05608, 2022.

[35] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, et al., "Do as i can, not as i say: Grounding language in robotic affordances," arXiv preprint arXiv:2204.01691, 2022.

[36] S. Nair, E. Mitchell, K. Chen, S. Savarese, C. Finn, et al., "Learning language-conditioned robot behavior from offline data and crowd-sourced annotation," in Conference on Robot Learning. PMLR, 2022, pp. 1303–1315.

[37] A. Zeng, A. Wong, S. Welker, K. Choromanski, F. Tombari, A. Purohit, M. Ryoo, V. Sindhwani, J. Lee, V. Vanhoucke, et al., "Socratic models: Composing zero-shot multimodal reasoning with language," arXiv preprint arXiv:2204.00598, 2022.

[38] M. Kwon, S. M. Xie, K. Bullard, and D. Sadigh, "Reward design with language models," arXiv preprint arXiv:2303.00001, 2023.

[39] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi, "A survey of imitation learning: Algorithms, recent developments, and challenges," 2023.

[40] N. Houlsby, A. Giurgiu, S. Jastrzebski, B. Morrone, Q. de Laroussilhe, A. Gesmundo, M. Attariyan, and S. Gelly, "Parameter-efficient transfer learning for nlp," 2019.

[41] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang, K.-W. Chang, and J. Gao, "Grounded language-image pre-training," in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, June 2022. [Online]. Available: http://dx.doi.org/10.1109/CVPR52688.2022.01069

[42] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez, et al., "Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality," See https://vicuna. lmsys. org (accessed 14 April 2023), 2023.

[43] G. Wang, Y. Ge, X. Ding, M. Kankanhalli, and Y. Shan, "What makes for good visual tokenizers for large language models?" arXiv preprint arXiv:2305.12223, 2023.

[44] A. Dubey, A. Jauhri, and A. e. a. Pandey, "The llama 3 herd of models," 2024. [Online]. Available: https://arxiv.org/abs/2407.21783

[45] A. Kamath, M. Singh, Y. LeCun, G. Synnaeve, I. Misra, and N. Carion, "Mdetr-modulated detection for end-to-end multi-modal understanding," in ICCV, 2021, pp. 1780–1790.

[46] A. Pashevich, R. Strudel, I. Kalevatykh, I. Laptev, and C. Schmid, "Learning to augment synthetic images for sim2real policy transfer," in IROS). IEEE, 2019, pp. 2651–2657.

[47] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, and N. de Freitas, "A generalist agent," 2022.

[48] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, R. Ring, E. Rutherford, S. Cabi, T. Han, Z. Gong, S. Samangooei, M. Monteiro, J. Menick, S. Borgeaud, A. Brock, A. Nematzadeh, S. Sharifzadeh, M. Binkowski, R. Barreira, O. Vinyals, A. Zisserman,

and K. Simonyan, “Flamingo: a visual language model for few-shot learning,” 2022.