arXiv:2611.32053v2 [cs.AI] 25 Nov 2026

EffectWAM：学习用于机器人操作的动作落地效果

Yuan Zhang[1,†], **Hao Cui**[1,†], **Jingfei Ni**[1,†], **Chi Liu**[1], **Xin Nie**[1], **Jiajia Wu**[1], **Meiyi Wang**[1], **Wenjie Xu**[1], **Lin Gao**[1], **Hao Li**[1], **Xianhe Zhang**[1], **Diyuan Liu**[2], **Jia Pan**[1,*], **Chao Ji**[2,*]

[1]科大讯飞, [2]LindenBot

摘要

世界–动作模型将视觉预测与可执行的机器人控制连接起来。我们提出 EffectWAM，通过三个互补组件发展这种连接：Effect、相机位置编码（CaPE）和动作落地。Effect 将视觉变化表示为独立编码的未来观测与共享的当前观测锚点之间的差异。当可靠的外参可用时，CaPE 根据相对相机位姿对端点池化进行条件化；对于未经标定的轨迹，则与无位姿池化共享相同的学习参数。动作落地训练视觉表征，使其保留测得的末端执行器位姿和夹爪状态变化。观测编码器的指数移动平均为 Effect 提供目标。共享生成专家通过流匹配联合预测 Effect 和可执行动作块，并复用采样得到的二者，按需生成未来视频。这使控制器无需等待视频合成即可执行动作。我们在超过 20,000 小时的异构具身数据上进行训练，并适配特定于数据集的关节、末端执行器和手部位姿表示。我们的评估在全部五个基准测试上使用 Small（8B）配置，并在 RoboDojo 上额外评估 Large（26B）配置。实验考察了互补基准测试上的闭环操作；与多种基线策略的比较表明，动作落地能够带来一致的性能提升。

日期： 2026年11月27日
联系邮箱： jiapan@iflytek.com；chaoji@iflytek.com
项目代码： https://github.com/haocui1715/xx-wam-mem-0.5-preview
项目模型： https://huggingface.co/haocui/xx-wam-mem-0.5-preview

1 引言

语言条件下的机器人操作要求策略将任务指令转化为一系列动作，从而在环境中实现预期变化。在闭环执行过程中，策略会反复观察场景，并随着任务推进调整动作。预测场景在一个动作时域内可能如何演变，有助于策略将当前指令与预期结果联系起来。

因此，选择预测未来的哪些内容，是学习用于机器人控制的视觉动态的核心。

视觉-语言-动作（VLA）模型学习将观测和指令映射为机器人动作 [1], [2]。其动作监督使这些指令产生的视觉后果处于隐含状态。世界-动作模型（WAM）增加了预测视觉动态的显式目标 [3], [4]。未来视频提供了丰富的训练

† 表示共同第一作者，* 表示通讯作者

信号，但在每次决策时合成视频会为闭环控制增加延迟。Fast-WAM 仅在训练期间使用未来视频，从而避免了这一成本 [5]。在匹配实验中，LAWA 发现，与视频和动作联合预测相比，这种方法的泛化能力较弱，尤其是在机器人示范有限的情况下；同时还表明，在 RoboCasa 上，在动作预测的同时预测紧凑的潜变量，可以在更低延迟下取得与视频和动作联合预测相当的成功率 [6]。这促使我们采用一种策略：在测试时保留对动作后果的紧凑预测，以支持操作性能，同时将视频合成排除在控制路径之外。

目标的选择决定了策略学会预期哪些变化。例如，La-JEPA 使用冻结的 V-JEPA 2.1 编码器，在监督机器人动作的同时监督未来预测 [7], [8]。冻结的视频特征继承了预训练期间学到的区分能力。通用的视频预测目标不会直接监督模型保留夹爪开合度或末端执行器位姿的细微变化，即使这些变化对控制很重要。机器人示范通过同步图像和测量的状态变化提供了这种信号。我们利用这些测量来塑造表征，并以此为策略提供未来预测目标。

为此，我们提出 Effect，这是一种紧凑的表征，用于表示动作时域内的视觉变化，并从同一示范中的成对观测和机器人测量数据中学习。共享编码器分别处理当前观测和选定的未来观测。我们从每个未来特征中减去当前特征，使不同时间跨度上的变化都相对于机器人执行动作时所处的状态来表示。辅助头从特征差中读出末端执行器位姿和夹爪状态的测量变化，从而鼓励编码器保留有关已执行运动的信息。在策略训练期间，编码器的指数移动平均（EMA）提供由此得到的 Effect目标。

在不同机器人构型和相机配置下，相似的操作动作可能产生不同的图像变化，因为摄像头的安装方式和运动方式各不相同。例如，腕部相机接近抽屉时，抽屉尚未移动，图像中的把手就可能发生偏移。Qwen-RobotManip 利用显式的相机几何，在异构机器人数据中对齐视觉观测和末端执行器运动 [9]。为支持在不同机器人构型和相机配置之间进行转移，当可靠的外参可用时，我们根据相对于当前观测的相机位姿来调节 Effect 的注意力池化。这为编码器提供几何上下文，使其能够区分由相机运动引起的变化与场景中的变化。当外参不可用时，同一编码器则使用共享权重进行无位姿池化，使这两类轨迹都能参与训练。

EffectWAM 通过两个互补的流匹配目标整合 Effect、CaPE 和动作落地。第一个目标根据当前观测和指令，对 Effect 与示范动作块进行联合建模。第二个目标则根据相同的上下文、干净的 Effect目标以及示范动作，对未来视频潜变量进行建模。在推理时，策略联合采样 Effect 和动作块，并在每次控制决策中保留对视觉变化的预测。联合采样一完成，控制器便可执行采样得到的动作块。需要视觉预测时，视频模式会使用同一效果–动作对。

我们使用涵盖异构动作表示的 20,000 多小时具身数据进行训练。在 LIBERO [10], LIBERO-Plus [11], RoboTwin 2.0 [12], Robo-Casa365 [13], 和 RoboDojo [14], 上评估 Small（8B）配置，并进一步在 RoboDojo 上评估 Large（26B）配置。实验对比区分了这些配置，并将 Small（8B）与现有策略进行比较。我们还在多个基线模型上考察了动作落地，其性能均得到稳定提升，证明了动作落地的价值并不局限于单一架构。

我们的贡献如下。

- **效果：对视觉变化的紧凑预测**。我们以共享的当前观测锚点为基准表示未来观测，并联合预测由此产生的效果序列与可执行动作块。同一效果–动作对还可作为条件用于可选的视频生成，从而将控制与视觉预测连接起来，而无需在每个决策时刻进行视频合成。

- **CaPE：几何感知视觉转换编码**。我们根据相对相机位姿对端点池化进行条件化处理，以适应机器人和相机配置下的视点变化。已标定和未标定

轨迹共享相同的学习所得池化参数，从而支持从异质示范中进行联合学习。

- **动作落地：来自测得运动的监督信号**。我们使末端执行器和夹爪的变化能够从效果中恢复，从而围绕与机器人控制相关的运动来塑造视觉表征。在多个基线策略上的评估表明，这一目标带来了稳定的改进，说明其益处并不局限于我们的完整模型。

2 相关工作

世界-动作模型的预测目标。未来视频为控制提供了学习信号，但不同模型会在策略的不同环节使用这一信号。视频优先策略从预测帧中推断控制信号 [15], ，而 DreamZero 则联合生成视频和动作 [3]。Fast-WAM 在训练期间保留视频预测，并在动作推理期间跳过该过程 [5]。在紧凑特征空间中进行预测，则提供了另一种利用未来观测的方式。V-JEPA 2.1 通过自监督视频预训练学习稠密的时空特征 [7]。La-JEPA 使用预训练的 V-JEPA 2.1 编码器来定义未来特征，其策略在训练期间会在预测动作的同时预测这些特征 [8]。Intern W0-Δ 则更明确地利用视频特征的变化。其因果印记回归相邻视频潜变量之间的差异，并在以单独的动作专家为条件之前，将中间特征与未来视频特征对齐 [4]。在每种情况下，用于构建预测目标的特征空间都会塑造策略所接受训练、需要预期的视觉变化。

学习视觉转变的表征。潜在动作模型学习转变本身的表征。LAPA 将帧对之间的变化压缩为离散编码，以支持未来帧重建，并利用这些编码进行策略预训练 [16]。后续工作采用额外信号来捕获操作相关信息。WALA 通过预测预训练语义特征和估计深度中以当前帧为锚点的变化来学习潜在动作，然后利用其预训练的潜在动作编码器为策略提供监督信号 [17]。LAWA 使用通过下一帧重建和操作区域掩码训练的分词器，然后对连续潜变量和可执行动作进行联合去噪 [6]。机器人测量数据也可以塑造表征。WLA^3 将同步视图与可用的本体状态变化共同编码，并通过重叠窗口之间的多模态重建和一致性进行学习 [18]。在我们的方法中，末端执行器和夹爪测量数据监督一个从视觉特征差中进行辅助读出的模块。观测编码器在策略训练期间进行学习，其指数移动平均副本提供 Effect目标。最终得到的效果与动作联合预测，也可以与采样动作块共同作为视频预测的条件。我们将动作落地视为一种独立的监督机制，并在多个基线策略上评估其贡献，同时评估完整模型。

视觉转变编码中的相机几何。视觉转变既能反映物理交互，也能反映视点变化。EscherNet 引入相机位置编码（CaPE），以在注意力中表达相对相机位姿；PRoPE 则将相对相机编码扩展至包含相机内参和外参 [19], [20]。我们使用 CaPE 在池化每个端点时，为 Effect 编码器提供明确的视点信息。我们将可用的相机位姿表示为相对于当前观测的位姿，并对具有可靠外参和不具有可靠外参的轨迹使用相同的池化权重。

3 模型设计

3.1 问题表述

在每个决策时刻 t_0，策略都会观测可用的 RGB 相机视图 $I_0^{1:V}$，并接收一条语言指令 ℓ。我们将条件信息记为 $C=(I_0^{1:V},\ \ell,\ \mathcal{S},\ \mathcal{Q})$，其中 $\mathcal{S}$ 指定机器人的动作表示，$\mathcal{Q}$ 指定相对于 t_0 的预测时域。模型会预测一个可执行动作块 $A=(a_1,\ \ldots\ldots,\ a_{N_A})$ 和一个效果序列 $E=(e_1,\ \ldots\ldots,\ e_{N_E})$。每个 e_k 表示从当前观测到指定未来时刻的变化。动作和效果分别根据各自的时间支撑集进行索引。我们将未来视频的潜变量记为 Y。

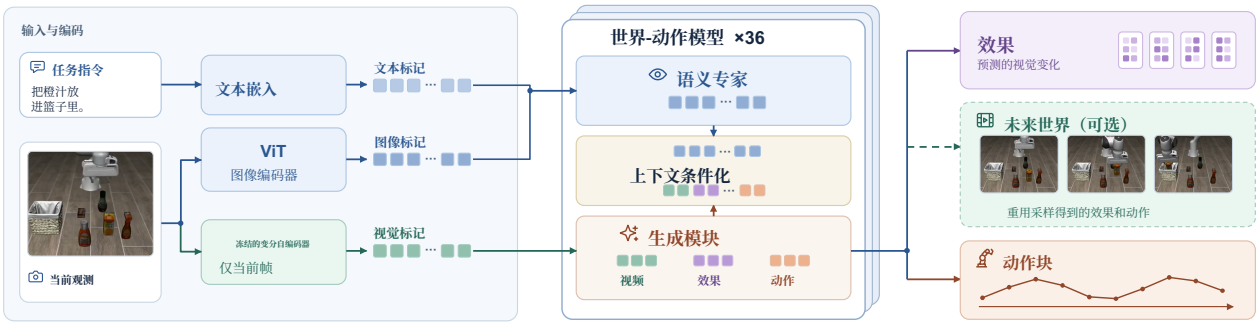

图 1 世界–动作推理。世界–动作混合变换器（MoT）通过掩码自注意力结合语义专家（SEM）和共享生成专家（GEN）。SEM 接收指令和图像标记，GEN 则接收由冻结的视频变分自编码器编码的当前观测中的视觉标记。GEN 联合采样一个效果序列 $\hat{E}$ 和一个动作块 $\hat{A}$。根据需要，第二次生成过程使用同一组采样结果生成未来视频。

我们通过以下条件分解对这三个变量进行建模

$$p_\theta(E, A, Y \mid C) = p_\theta(E, A \mid C)\, p_\theta(Y \mid C, E, A). \tag{1}$$

第一个因素为策略提供一个动作块，以及对伴随视觉变化的预测。根据需要，第二个因素利用这一对结果生成视觉预测。训练期间，未来观测为 E 和 Y提供目标。策略根据 C中的当前信息进行条件化，联合预测效果–动作。

3.2 架构概述

EffectWAM 为公式（1）中的两个因素构建共同的观测–语言上下文，将联合效果–动作预测与条件视频生成联系起来（图1）。任务指令被嵌入为文本标记，可训练的视觉变换器（ViT）将可用的当前相机视角映射为合并图像词元。SEM 接收这两组标记，并将中间视觉特征注入早期的语义专家模块。同时，冻结的视频变分自编码器仅编码当前观测。学习得到的投影将其潜变量映射为供 GEN 使用的视觉标记。

混合变换器（MoT）使用掩码自注意力来耦合语义专家（SEM）和 GEN。GEN 在效果、动作和视频之间共享其变换器主干网络，并为不同模态分别设置输入和输出投影。它首先联合采样效果流和动作流。可选的视频处理阶段会复用采样得到的二元组和原始观测上下文，详见第 3.5 节。训练期间，ViT 还会向第 3.3 节中的观测端点编码器提供合并前的图像块特征。Small（8B）和 Large（26B）配置的组件参数数量列于表 3 中。

3.3 物理基础效果学习

效果表示。 我们通过将未来观测表示为相对于共享当前观测锚点的结果（图 2），从主摄像头演示片段中学习效果。参考配置以 4、8、12 和 16 帧的偏移量采样未来帧。ViT 独立编码每个采样帧。给定帧 I_k的合并前图像块特征，观测端点编码器（OBE）使用一个学习查询和四个注意力头对图像块进行池化。输出投影和一个两层多层感知机生成 $u_k \in \mathbb{R}^d$，其中 $d = 256$。径向映射 $z_k = \sqrt{d}\, u_k / \sqrt{d + \|u_k\|_2^2}$ 给出有界端点特征。每个效果词元都会将一个未来端点与相同的当前端点 z_0进行比较，

$$e_k = \frac{z_k - z_0}{\sqrt{2}}. \tag{2}$$

差值序列描述了从发出动作块时的状态到每个请求的未来时刻之间的变化。

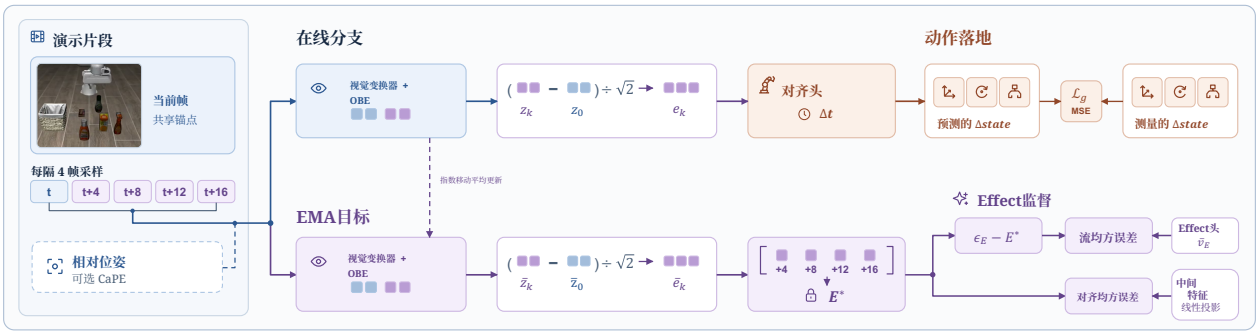

图2 物理基础效果学习。在线分支和指数移动平均（EMA）分支对相同的示范帧进行编码，并相对于共享当前帧形成端点差异。可选的摄像头位置编码（CaPE）为两个分支提供条件。在线接地头使用 e_k 和经过时间间隔 Δt_k 来预测测量的状态变化。EMA分支将其差异堆叠为分离目标 $E^\star$。最终的GEN 头以速度目标 $\epsilon_E - E^\star$进行监督，并使用 $E^\star$ 直接监督中间效果状态的线性投影（参考配置中的方块9）。

相机位姿感知池化（CaPE）。这些端点差异既可以反映相机运动，也可以反映物理交互。当可靠的外参可用时，我们将相机位置编码（CaPE） [19] 纳入端点池化。令 $T_{C_k \leftarrow W}$ 表示主相机在时刻 t_k的世界到相机变换。我们将位姿表示为相对于当前相机的形式，

$$P_k = T_{C_k \leftarrow W} \left(T_{C_0 \leftarrow W}\right)^{-1}, \qquad P_0 = I_4. \tag{3}$$

对于帧 k的图像块 j 处的投影注意力键 k_j ，我们将每个键头拆分为四维行向量方块，并将每个方块右乘 P_k。等价地， $k'_j = k_j \,\mathrm{diag}(P_k, \ldots, P_k)$。帧中的所有图像块共享相同的相对变换。查询和值保留其原始投影。对于未采用相机位姿条件化的轨迹，池化使用未经变换的键。在线分支和EMA分支在可用时使用相同的相对位姿。在每个分支内，两条池化路径使用相同的学习查询、投影和端点映射，并且在 $P_k = I_4$时二者重合。

动作落地与指数移动平均目标。为将这些视觉差异与测得的物理变化关联起来，对齐头接收在线Effect e_k 以及从当前帧到对应未来帧的经过时间间隔 Δt_k 。它预测该时间间隔内末端执行器平移、旋转和夹爪状态的变化。物理接地损失 $\mathcal{L}_g$ 在每个测量类别内进行归一化后，使用平方误差将这些预测与可用量的测量值进行比较。我们还通过对同一观测的两个光度视图施加一致性约束，并对特征方差和协方差进行约束，来正则化当前端点。

为从学习到的端点表示构建预测目标，指数移动平均（EMA）跟踪合并前视觉变换器和OBE。EMA分支对相同帧进行编码，并相对于其自身的当前端点 $\bar{z}_0$计算端点差异。以 $\bar{z}_k$ 表示未来EMA端点，并以sg表示停止梯度算子，我们按照时间顺序堆叠这些差异，以获得干净目标，

$$e_k^\star = \mathrm{sg}\left(\frac{\bar{z}_k - \bar{z}_0}{\sqrt{2}}\right), \qquad E^\star = (e_1^\star, \ldots, e_{N_E}^\star). \tag{4}$$

如图2所示， $E^\star$ 同时提供Effect流匹配目标和中间对齐目标。在训练期间，它还对视频模式进行条件化，详见第3.4节。

3.4 共享生成与训练目标

我们将式（1）中的两个因子实现为 GEN 的两种前向模式，并共享参数。联合效果–动作模式（`ACT_EA`）接收带噪声的效果和动作标记，并利用当前上下文预测它们的速度。视频模式（`VIDEO_EA`）接收带噪声的未来视频标记，并以相同的

表1 训练期间的生成模式。两种模式都接收当前上下文 C。

Mode	额外的干净条件	带噪预测变量
`ACT_EA`	None	E_σ, A_σ
视频 _EA	$E^\star, A^\star$	Y_σ

上下文和一个干净的效果–动作对为条件。训练期间，该效果–动作对由停止梯度的指数移动平均 Effect目标$E^\star$ 和演示动作块 $A^\star$组成。表 1 总结了这两种模式。

在一个样本内，语义令牌在各自的流中进行因果注意。生成令牌关注语义令牌，以及当前生成模式中的干净令牌和带噪令牌。扩散时间步嵌入应用于带噪令牌。两种模式通过独立的前向传递进入同一个优化窗口。联合效果–动作模式使用观测到的上下文及其自身的带噪变量。冻结的视频变分自编码器对连续的演示片段进行编码，以获得未来视频目标 $Y^\star$。在参考配置中，其 $4\times$时间压缩将未来潜在切片置于与图中效果端点相同的帧偏移处。2视觉条件 C 来自仅对当前观测进行的单独编码。

对于每个预测变量 $x \in \{E, A, Y\}$，我们采样高斯噪声 $\epsilon_x \sim \mathcal{N}(0,\ I)$，并在参考配置中采样流时间 $\sigma = 5u/(1+4u)$，其中 $u \sim \mathcal{U}(0,\ 1)$。我们在干净目标 $x^\star$ 与噪声之间进行插值，并回归该路径上的速度，

$$x_\sigma = (1-\sigma)x^\star + \sigma\epsilon_x, \qquad v_x^\star = \epsilon_x - x^\star. \tag{5}$$

效果和动作共享流时间，并接收相互独立的高斯噪声。仅在视频模式下，未来视频潜变量会被添加噪声。我们将 $\mathcal{L}_e$、 $\mathcal{L}_a$ 和 $\mathcal{L}_y$ 定义为相应的平方速度误差：先对有效坐标取平均，再对符合条件的请求取平均。在图 2 中，最终的 GEN 头预测 $\hat{}v_E$，并将其与速度目标 $\epsilon_E - E^\star$进行比较。

在参考配置中，中间对齐损失 $\mathcal{L}_{\text{align}}$ 会对 GEN 第 9 个模块在效果令牌位置处的隐藏状态应用学习到的线性投影，并针对干净目标使用平方误差$E^\star$。根据训练计划启用 $\sigma \geq 0.5$ 后，该损失会应用于符合条件的带噪效果词元。观测损失 $\mathcal{L}_{\text{obs}}$ 将每个光度视图的在线端点与另一视图经过分离梯度处理的指数移动平均端点进行匹配。同时，它还会惩罚兼容组内较低的逐维方差和非对角协方差。结合第 3.3 节中的落地损失，组合目标为

$$\begin{aligned}\mathcal{L} = \lambda_e\mathcal{L}_e + \lambda_a\mathcal{L}_a + \lambda_y\mathcal{L}_y + \lambda_g\mathcal{L}_g \\ + \lambda_{\text{obs}}\mathcal{L}_{\text{obs}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}}.\end{aligned} \tag{6}$$

3.5 闭环推理与按需视频生成

在每个决策时刻，策略都会根据最新的 RGB 观测 C 、任务指令、动作接口和所需的预测时域构建上下文。ViT 派生的语义特征以及当前观测的冻结 VAE 潜变量，为采样提供视觉上下文。训练期间会使用观测端点编码器、物理读出和 EMA 目标编码器。推理时，GEN 根据当前上下文联合采样效果序列和动作块，如图 1 所示。

在联合效果–动作模式下，我们用相互独立的高斯噪声初始化效果和动作变量，并在共享流时间网格上将它们视为一个状态 $s_\sigma = (E_\sigma,\ A_\sigma)$。在每次求解器评估时，生成专家都会读取两条带噪数据流，并共同预测它们的速度，

$$\frac{\mathrm{d}s_\sigma}{\mathrm{d}\sigma} = \big(\hat{v}_E(s_\sigma, \sigma, C), \hat{v}_A(s_\sigma, \sigma, C)\big). \tag{7}$$

我们朝着 $\sigma = 0$ 进行积分，以获得 ($\hat{}E,\ \hat{}A$)。参考配置在每种生成模式下使用二阶 Flow UniPC 求解器，并进行四步积分。使用无分类器引导时，条件和

无条件速度估计在同一噪声状态下进行评估，并在每次求解器更新前合并。由于生成令牌隐藏状态可以关注不断变化的噪声变量，因此每次评估时都会重新计算这些状态。

ˆ 对于闭环控制，我们对 A 的有效坐标进行反归一化，并将其转换为 $\mathcal{S}$指定的命令表示。控制器会执行该动作块中预先配置的一段内容，这段内容可以是整个动作块，然后获取新的观测并构建下一个上下文。下一次决策会利用新的上下文采样一个新的效果–动作对。因此，联合采样器一旦完成集成，就会立即提供执行所需的命令。

需要进行视觉预测时，视频模式会保留用于生成 $(E,\ A)$ 的观测快照和预测时域 ˆ ˆ。它将这对确切内容作为生成器表征中的干净条件，并从独立的视频噪声样本开始。相同的生成专家会对视频速度 ˆ 进行积分，以获得 Y；冻结的变分自编码器则将当前潜变量和预测的未来潜变量解码为 RGB 帧。该预测从原始观测出发，描述采样动作块，包括控制器在重新规划前仅执行前缀时计划的后续部分。此流程使视频生成保持可选，并确保预测与用于控制的同一动作提案相对应。

4 训练策略

4.1 数据准备

EffectWAM 学习预测视觉变化和可执行动作。因此，训练结合了机器人示范、人类操作记录和第一人称视角视频，这些数据分别为机器人控制、手物交互和视觉动态提供互补的覆盖范围。该数据混合涵盖不同的实体形态、相机配置、操作技能以及不同程度的动作标注。

我们使用超过 20,000 小时的具身交互数据进行预训练。数据清单涵盖 34 个来源，共 25,292.8 小时。图 3 按记录的回合时长汇总了这份清单；图中显示的占比并非训练采样权重。附录 A 列出了完整的数据来源。

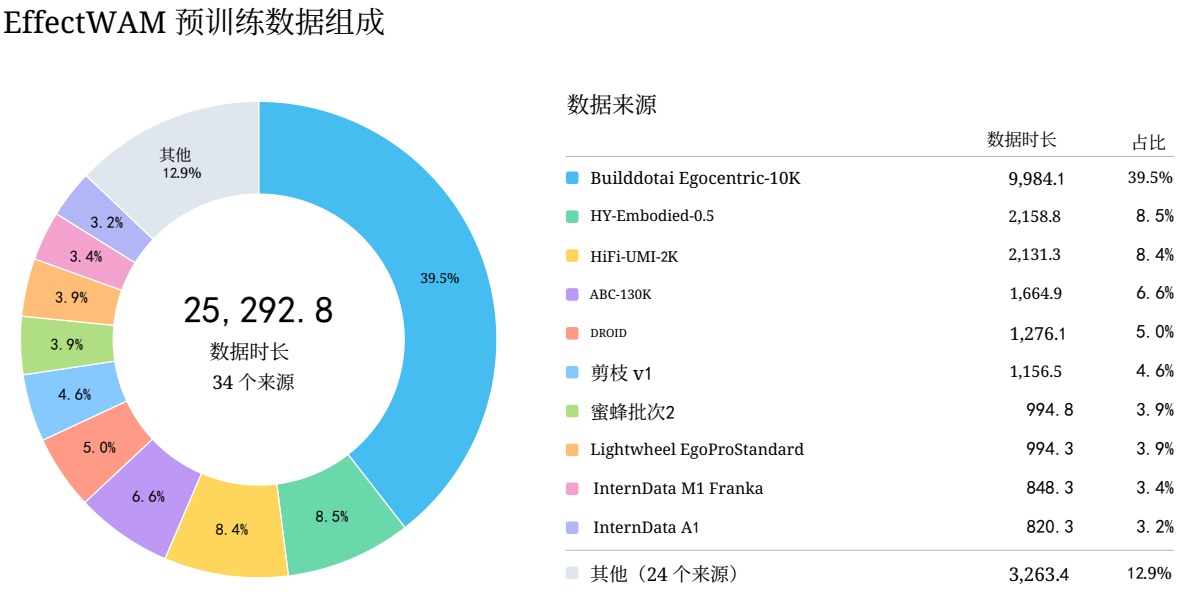

图3 EffectWAM按记录时长统计的预训练数据组成。环形图分别展示了占比最大的十个来源，并将其余24个来源归为其他（3,263.4小时；12.9%）。数据时长表示记录的回合时长，所有占比均以25,292.8小时的总计为基准进行归一化。

规模最大的来源包括 Builddotai Egocentric-10K、HY-Embodied-0.5、HiFi-UMI-2K、ABC-130K 和 DROID。其余数据混合了 InternData、RH20T、EgoDex、RoboCOIN、RoboMIND、Furni- tureBench，以及其他人类与机器人交互数据集。“其他”还包括官方 RoboDojo 仿真训练数据集（20.5 小时） [14]以及 GPT-Astra 机器人操作数据（10.5 小时）。

机器人示范为动作学习提供同步的视觉观测和控制标注。其原始记录可能包括关节位置、末端执行器位姿以及夹爪状态或指令；灵巧机器人示范还可能提供指关节或手部姿态坐标。我们通过相应的动作接口保留这些标注的物理含义。

人类操作记录扩展了所观测到的物体交互和手部动作范围。部分记录包括跟踪得到的腕部和关节式手部姿态，这些姿态描述的是人类动作，而非可执行的机器人指令。纯视频记录在缺少动作或物理状态标签的情况下，提供时间视觉监督。表 2 总结了具有代表性的记录类型及其原始标注。这些标注差异决定了训练期间可用的监督信号。

表 2 具有代表性的记录类型中的原始动作／状态标注。可用字段因数据来源和处理版本而异。

数据来源／记录类型	原始动作／状态标注
机械臂与夹爪轨迹	关节位置、末端执行器平移和方向，以及夹爪状态或在有提供的情况下，使用相应指令。
灵巧机器人示范	机械臂姿态，以及手指关节或手部姿态坐标。
人类手部记录	跟踪得到的腕部和关节式手部姿态。
仅视频第一人称记录	没有动作或物理状态标注；RGB序列和可用的任务文本。

4.2 训练过程

训练遵循第 3.4 节中的目标，并结合每条轨迹中可用的监督信号。当前相机观测、任务指令、动作接口和预测时域共同定义了条件上下文。可训练的视觉变换器向语义专家提供视觉特征，而冻结的变分自编码器则单独编码当前观测，以提供观测到的视觉潜变量。

预测目标由同一示范构建。EMA分支通过合并前视觉变换器和观测端点编码器，分别编码当前观测和选定的未来观测。根据公式（4），将未来端点与其共享的当前端点进行比较，以形成已停止梯度的效果目标。冻结的变分自编码器对演示片段进行编码，以提供未来视频目标。当前视觉上下文与这些未来预测目标保持独立。

共享生成专家在同一优化窗口内通过两种前向模式进行训练。联合Effect–动作训练使用共享的流时间，并为Effect和动作分别使用独立噪声。视频训练对未来视频潜变量添加噪声，同时以干净设置下的指数移动平均Effect目标和示范动作块作为条件。两种模式共享生成参数，并优化各自对应的速度预测目标。

每种模式都需要表1中规定的标注和条件变量。可用的物理测量值通过预测区间内末端执行器位姿和夹爪状态的变化，为动作落地提供监督信号。可靠的相机外参支持CaPE条件化端点池化；对于未标定轨迹，则采用无位姿路径，并使用相同的学习池化参数。观测一致性和中间Effect对齐提供了第3.4节所定义的额外监督信号。需要不可用标注的项将被排除。纯视觉记录仅贡献适用的视觉目标，不提供机器人动作标签。

在多源数据训练后，模型使用相应的动作接口，根据每个基准测试的示范进行适配。这既保留了动作标注的特定于来源的含义，又应用了共享的训练目标。

表 3 EffectWAM Small (8B) 和 Large (26B) 的模型配置。参数数量指所列模块的参数量。

规模	模块	参数量（B）
Small（8B）	VLM	2.2
	视频 DiT	5.0
	动作 DiT	1.2
Large（26B）	VLM	8.3
	视频 DiT	16.4
	动作 DiT	2.0

5 实验

我们在五个操作基准上评估 EffectWAM，以研究任务执行、鲁棒性、组合泛化和特定能力表现。比较内容包括相对于 InternW0-Δ [4] 的任务成功率、对环境扰动的鲁棒性，以及对不熟悉任务组合的泛化能力。

5.1 实验设置

我们评估两种配置：**Small（8B）**和**Large（26B）**，各组件的规模列于表3中。Small 在全部五个基准测试上进行评估，Large 还额外在 RoboDojo 上进行评估。我们在单独的表格中，将两种 RoboDojo 配置与模型规模相近的基线方法进行比较。

第5.2 节评估标准任务执行情况，以及面对观测、机器人初始状态和场景配置变化时的鲁棒性。LIBERO 衡量相对于 InternW0-Δ的任务表现，而 LIBERO-Plus 以及 RoboTwin 2.0 上的干净设置/随机化设置对比，则评估分布偏移下的表现。第5.3节考察这些提升能否从原子任务扩展到 RoboCasa365 上已见和未见的任务组合。RoboDojo 则通过能力层级分析补充这些比较，利用成功率和部分得分任务分数，刻画五个能力维度上的完整任务完成情况和中间进展。

在 LIBERO 和 RoboCasa365 上，我们使用相同的任务集和回合预算评估 InternW0-Δ (8B) 与 EffectWAM。EffectWAM 在 RoboTwin 2.0 上仅使用干净设置示范数据，在 RoboCasa365 上仅使用 Human300，且不使用 Target训练数据。RoboTwin 上的比较仅限于使用该基准测试中的干净设置示范数据进行适配的策略。模型规模为近似值，包括冻结的条件编码器，但不包括变分自编码器。

我们遵循各基准测试的聚合协议。LIBERO-Plus 和 RoboCasa365 根据评估回合汇总成功率；RoboTwin 2.0 对干净设置和随机化设置的成功率赋予相同权重；RoboDojo 则对其五个能力维度赋予相同权重。成功率以百分比表示；RoboDojo 还报告部分得分任务分数。Small 使用一个随机种子进行评估，Large 则报告三个随机种子上的变化。

5.2 任务表现与鲁棒性

标准任务执行。在引入分布偏移之前，我们使用标准LIBERO来确定任务执行表现。我们在四个套件上评估 EffectWAM 和 InternW0-Δ，每个套件包含 10 项任务，每项任务进行 50 个回合，每个模型总计 2,000 个回合。EffectWAM Small 的总体成功率为 99.20%，而 InternW0-Δ (表 4)。此比较涵盖监督策略及采用额外后训练的方法，为标准任务执行提供了更广泛的参照。

EffectWAM 在原本已很高的基线上进一步提升了总体成功率，其中 Goal套件的提升最为明显：成功率从 98.00% 上升至 99.20%。Object成功率也提升至 100.00%，而 Spatial 则维持在 99.20%。总体提升主要由 Goal套件带动，同时在物体和空间任务上也保持了较高的成功率。其 99.20% 的总体成功率与 SRPO 持平，且每个套件的成功率均不低于 98.40%。

分布偏移下的鲁棒性。接下来，我们使用 LIBERO-Plus 评估这种出色的标准任务表现能否适应相机视角、机器人初始状态、语言、光照、背景、视觉噪声和场景的变化

表 4 标准LIBERO成功率（%）。最高得分以**红色粗体**显示，第二高且不同的得分带下划线。

方法	规模（B）	Spatial	物体	目标	长程	总体
OpenVLA-OFT（联合） [21]	7.0	97.7	98.0	96.1	95.3	96.8
UniVLA [22]	7.0	96.5	96.8	95.6	92.0	95.2
RIPT-VLA [23]	7.0	98.6	98.6	99.0	93.8	97.5
NORA-1.5（DPO） [24]	4.0	98.0	96.0	95.4	90.5	95.0
SRPO [25]	7.0	98.8	**100**	**99.4**	98.6	**99.2**
AVA-VLA（联合） [26]	7.0	97.4	99.4	97.4	97.6	98.0
InternW0-Δ [4]	8.0	**99.2**	99.8	98.0	**99.0**	99.0
EffectWAM Small	8.0	**99.2**	**100**	99.2	98.4	**99.2**

表 5 LIBERO-Plus 成功率（%）。总体成功率根据评估回合汇总。最高得分以**粗体红色**显示，第二高且不同的得分加下划线。

方法	规模（B）	摄像头	机器人	语言	光照	背景	噪声	布局	总体
SRPO（增强版） [25]	7.0	83.4	62.0	73.6	**97.2**	97.7	85.7	75.2	82.1
OpenVLA-OFT [21]	7.0	56.4	31.9	79.5	88.7	93.3	75.8	74.2	69.6
RIPT-VLA [23]	7.0	55.2	31.2	77.6	88.4	91.6	73.5	74.2	68.4
NORA-Long [27]	3.0	2.20	37.0	65.1	45.7	58.6	12.8	62.1	39.0
π_0 [2]	3.3	13.8	6.0	58.8	85.0	81.4	79.0	68.9	53.6
π_0-FAST [28]	3.0	65.1	21.6	61.0	73.2	73.2	74.4	68.8	61.6
InternW0-Δ [4]	8.0	90.6	91.1	92.9	95.8	99.4	**94.0**	88.4	92.8
EffectWAM Small	8.0	**91.7**	**91.4**	**93.4**	96.5	**99.6**	93.4	**88.7**	**93.1**

SRPO（增强版）使用增强数据配置。对 OpenVLA-OFT、RIPT-VLA、NORA 以及 π_0 各变体的评估来自 LIBERO-Plus [11]。

布局。EffectWAM Small 在 10,030 个回合中成功完成 9,338 个，总体成功率为 93.10%（表5）。该成功率根据所有回合计算，以计入类别规模不等的情况。总体成功率 93.10%，比报告的 InternW0-Δ 结果高 0.30 个百分点。类别级结果展现了这一表现如何应对不同的变化来源。

高成功率覆盖七个扰动类别：其中六类超过 91%，所有类别均至少达到 88.66%。在背景和光照变化下，性能尤为出色，成功率分别达到 99.54% 和 96.50%；语言变化和视觉噪声变化下的成功率也都超过 93%。摄像头和机器人变化下的成功率同样保持在 91% 以上。这些结果表明，无论是外观变化，还是摄像头视点或机器人初始状态的变化，任务完成率都能保持较高水平；这也以广泛的扰动覆盖范围补充了标准任务结果。

随机化设置下的双臂操作。 RoboTwin 2.0 测试在相同任务集于场景随机化条件下进行评估时，策略能否保持双臂操作性能。我们在干净设置和随机化设置下分别使用 50 个任务，每个任务包含 100 个回合，总计 10,000 个回合。EffectWAM Small 在干净设置下的成功率为 91.82%，在随机化设置下为 74.22%，等权平均值为 83.02%（表 6）。该比较中的所有方法均仅使用 Clean示范数据进行基准适配。因此，随机化评估测试的是向这些示范数据中未包含的场景条件进行转移的能力。

EffectWAM 在两种设置下的表现均优于已报告的 InternW0-Δ 结果：在干净设置下高出 1.82 个百分点，在随机化设置下高出 2.32 个百分点。因此，相较于这一参照方法，EffectWAM 在随机化双臂操作中的优势依然明显，且在随机化设置下优势更大。其 83.02% 的平均成功率也高于 PICA-WAM 已报告的 82.7%，且在干净设置下的成功率更高。仅使用 Clean 训练后，在随机化设置下仍保持 74.22% 的成功率，证明其转移能力超出了基准训练条件。

各项任务进一步体现了模型在不同设置下的稳定执行能力。例如，shake_瓶子在干净设置下的成功率为 100%，在随机化设置下的成功率为 99%。该任务几乎未发生变化的完成率表明，成功的

表6 仅使用 Clean 训练时，RoboTwin 2.0 在 Clean 和随机化测试条件下对 50 项任务进行评估所得的成功率（%）。最高得分以**红色粗体**显示，第二高的不同得分带下划线。

方法	规模（B）	干净设置→干净设置	干净设置→随机化设置	平均值
Fast-WAM [5]	11.7	77.8	1.9	39.9
4D-WAM [29]	11.0	81.5	41.8	61.7
OpenWAM-α [30]	11.7	89.4	48.7	69.0
InternW0-Δ [4]	8.0	90.0	71.9	81.0
PICA-WAM [31]	8.0	<u>90.7</u>	**74.7**	<u>82.7</u>
EffectWAM Small	8.0	**91.8**	<u>74.3</u>	**83.1**

Fast-WAM 和 4D-WAM 的结果来自 InternW0-Δ [4]。4D-WAM 使用 LingBot-VA 架构 [32]。

表 7 RoboCasa365 在 50 项评估任务上的成功率（%）。最高得分以**红色粗体**显示，次高且不重复的得分带下划线。

方法	规模（B）	原子已见	已见组合任务	未见组合任务	总体
Xiaomi-Robotics-1 [33]	5.1	**80.2**	**57.1**	**32.1**	**57.4**
RLDX-1 [34]	≈ 7	67.6	27.9	8.5	36.0
InternW0-Δ [4]†	8.0	76.5	47.7	17.3	48.3
EffectWAM Small	8.0	<u>76.9</u>	<u>48.2</u>	<u>19.4</u>	<u>49.3</u>

†：由我们评估。

执行能力能够在场景随机化下保持不变。完整的逐任务结果详见附录 B。

5.3 组合操控与能力分析

组合泛化。我们使用 RoboCasa365 来考察任务执行能力的提升是否能扩展到行为组合，包括训练期间未见过的组合。其原子已见、已见组合任务和未见组合任务划分，分别衡量原子任务、熟悉组合和陌生组合上的表现。

我们在 50 项任务上评估版本 1.0.1，每项任务包含 30 个回合。这 1,500 个回合由 540 个原子已见回合、480 个已见组合任务回合和 480 个未见组合任务回合组成。InternW0-Δ 与 EffectWAM Small 使用相同的任务分组和回合预算。EffectWAM 仅使用 Human300 进行训练；Target训练数据未被使用。总体成功率按各划分中的回合数加权。表 7比较了三个划分上的表现及按回合数加权的总体成功率。

EffectWAM 在三个划分上的成功率均有所提升，成功回合数为 739 个，而 InternW0-Δ 为 725 个。总体成功率从 48.33% 提高到 49.27%，根据回合数计算，提升了 0.93 个百分点。提升最大的是未见组合任务：新增的 14 个成功回合中，有 10 个来自这一划分，相当于提升了 2.08 个百分点。原子已见和已见组合任务各增加了 2 个成功回合。因此，这一收益也扩展到了陌生任务组合，其提升幅度高于表现更好的原子任务。

特定能力的表现。我们使用 RoboDojo 来评估 EffectWAM 在泛化、Precision、长时程、记忆和开放能力方面的表现。成功率（SR）用于衡量任务是否完整执行，而基准测试任务得分也能反映任务取得的部分进展。结合各项任务的结果，这些指标揭示了该策略最擅长的能力及其任务行为。

我们在全部 42 个 RoboDojo 仿真任务上评估这两种配置：Small 在训练步骤 3,000 时使用随机种子 0，Large 在步骤 7,500 时使用三个评估随机种子。对于 12 个泛化任务中的每一个随机种子，均使用 25 个标准回合和 25 个随机化回合；对于其余 30 个任务中的每一个，则使用 50 个回合。由此，Small 共包含 2,100 个回合，Large 共包含 6,300 个回合。遵循官方协议 [14], ，我们合并每个泛化任务的标准回合和随机化回合，计算每项能力内的任务指标平均值，

表 8 RoboDojo 在两种模型规模下的结果。能力列和总体报告成功率（%）；得分为总体部分得分任务分数。结果以 ± 的均值 ± 表示，标准差基于三个随机种子计算；EffectWAM Small 使用一个随机种子。最高得分以 **粗体红色**显示，第二高的不同得分加下划线。

方法	规模（B）	泛化	Precision	长时程	记忆	Open	总体	得分
小规模								
Awomo-0.5V [35]	≈ 10.7	20.80	**28.10**	25.00	12.50	**28.10**	22.90	28.70
PrismWAM（小规模） [36]	≈ 11.7	19.56	19.50	22.08	30.11	1.500	18.55	25.10
EffectWAM Small	≈ 8.40	**24.33**	20.75	**27.00**	**55.00**	12.25	**27.87**	**33.53**
大规模								
VPP2 [37]	≈ 24.7	29.50	**35.17**	**37.00**	**55.56**	4.08	32.26	39.26
PrismWAM（大规模） [36]	≈ 25.0	29.33 ± 0.98	31.08 ± 2.66	31.00 ± 0.74	50.67 ± 0.27	4.42 ± 0.66	29.30 ± 0.61	36.23 ± 0.55
EffectWAM Large	≈ 26.7	**33.22** ± 1.14	34.67 ± 0.72	36.92 ± 1.12	54.22 ± 1.37	**11.00** ± 1.24	**34.01** ± 0.12	**40.29** ± 0.32

模型规模根据模型组件估算。

并为五个能力均值赋予相同权重。Small 的结果是单种子估计值，不提供跨随机种子的标准差；Large 的结果则报告跨随机种子的均值和总体标准差。

表 8 按模型规模对比结果分组，每组内的参数范围均小于 5B。EffectWAM Small 的总体成功率为 27.87%，任务分数为 33.53。其总体成功率高于报告的 Awomo-0.5V 和 PrismWAM（小规模）结果。记忆是其表现最强的能力，成功率为 55.00%，任务分数为 55.38；其记忆成功率也高于选定的小规模基线方法。长时程任务的成功率为 27.00%，得分为 41.29，表明即使未完整执行任务，也能获得相应的进度分数。其 Precision 和开放能力仍弱于 Awomo-0.5V。单种子评估无法量化运行间变异。

在更大的规模下，EffectWAM Large 的总体成功率为 34.01%，任务分数为 40.29，分别高于 VPP2 1.75 和 1.03 分（表 8）。与 PrismWAM* （Large）相比，其总体成功率提高了 4.71 个百分点，任务分数提高了 4.06 分，且五项能力的成功率均更高。相较于 VPP2，其在开放任务上的成功率优势最大，从 4.08% 提升至 11.00%；泛化能力也从 29.50% 提升至 33.22%。因此，相较于 VPP2，其优势主要体现在开放和泛化能力上。任务分数范围为 0 至 100，可计入部分完成的得分；而成功则要求完整完成任务。在本次评估中，记忆是表现最强的能力，成功率为 54.22%，与之相近的任务分数为 54.62。个别任务的完成率尤其高：按压_按_数量在 150 个回合中成功 149 次，而 覆盖_方块 在 150 个回合中成功 144 次。这两项指标高度一致，表明该类别的得分主要来自任务完整完成；这些代表性任务也具体展现了其可靠的表现。

长时程任务的任务分数为 53.44，完整任务成功率为 36.92%。任务分数还涵盖了在计为完整成功的回合之外获得的额外进展，从而为策略执行扩展任务的表现提供了互补视角。Memory 和长时程任务的结果共同凸显了 EffectWAM Large 能力概况的两个方面：在 Memory 任务中频繁实现完整执行，以及在长时程任务中取得大量计入得分的进展。

执行可视化。为将能力得分与具体的操作行为联系起来，图 4 展示了 EffectWAM Large (26B) 在 RoboDojo 五个能力类别中的每一类里成功执行的示例 [14]。泛化示例展示了机器人如何在带有彩色光照的杂乱场景中，将旋转衣物的袖子和下摆折叠起来，将可变形物体处理与视觉外观变化结合起来。在长时程序列中，机器人处理了四个具有不同初始姿态的瓶子。一个瓶子返回桌面后被重新抓取，随后，其他瓶子在处置前由夹爪之间相互传递，在重复的抓取和转移操作中保持进展。

“记忆”示例首先展示三个彩色积木，随后按照所需的红色—绿色—蓝色顺序将它们逐一显露出来。这一顺序不同于它们的空间顺序，因此，在遮挡前观察到的颜色—位置关联在选择要抬起的杯子时仍然具有参考意义。在“开放”示例中，机器人使用尺子将积木排列成一条直线，然后

将其从左侧末端重新定位，以纠正剩余的错位。“Precision”示例交替进行单个支撑块放置与水平平台的协同搬运，随后添加两个顶部组件，完成一个八块塔。每一层新结构都依赖于其下方的支撑排列，并且结构在松手后仍保持站立。这些序列通过展示完成任务所需的交互，为表 8中的汇总结果提供补充示例。附录 C 展示了同五个类别中的另外 25 条成功轨迹。

注意力可视化。为考察预测过程中强调的视觉信息，图 5 比较了完整模型中未来Effect预测的注意力与移除Effect监督信号后的未来RGB注意力。两者均采用 Small（8B）配置。在 AgiBot World [38], 中，推动过程中，Effect响应仍集中在夹爪与门的交互附近，其他区域的响应稀疏且微弱；RGB注意力图还突出显示了远处的人、附近的固定装置和地面区域。在 LIBERO [10], 中，两种变体都关注着被持握的碗向盘子下降的过程。仅RGB注意力图还额外强调了非目标碗和左侧的固定橱柜。在 Ego4D [39], 中，随着第一人称视角变化，被持握的塑料盖仍是Effect注意力图的主要焦点；而RGB预测还突出显示了柜台上的金属盖、容器和外围设备。在这些示例中，Effect的非目标响应更少、更小且通常更暗，同时在被操纵物体附近保留了显著的活动。这一模式符合其着重关注与动作相关的视觉变化这一设计目标，也从定性角度补充了鲁棒性结果。



图4 EffectWAM Large（26B）成功完成的 RoboDojo 仿真执行示例：(a) 在随机化设置的场景中叠衣服，(b) 将瓶子放入垃圾桶，(c) 覆盖并显露彩色积木，(d) 使用尺子对齐方块，以及 (e) 搭建塔。每一行从左至右对应一条记录的头部摄像机轨迹。帧以不等间隔捕捉与任务相关的事件；集编号和从零开始的源视频帧索引用于标识记录。每行的主视图均采用固定裁剪区域。蓝色轮廓标出同一帧中以蓝色边框插图显示的区域；Memory 细节裁剪区域在各阶段保持固定。

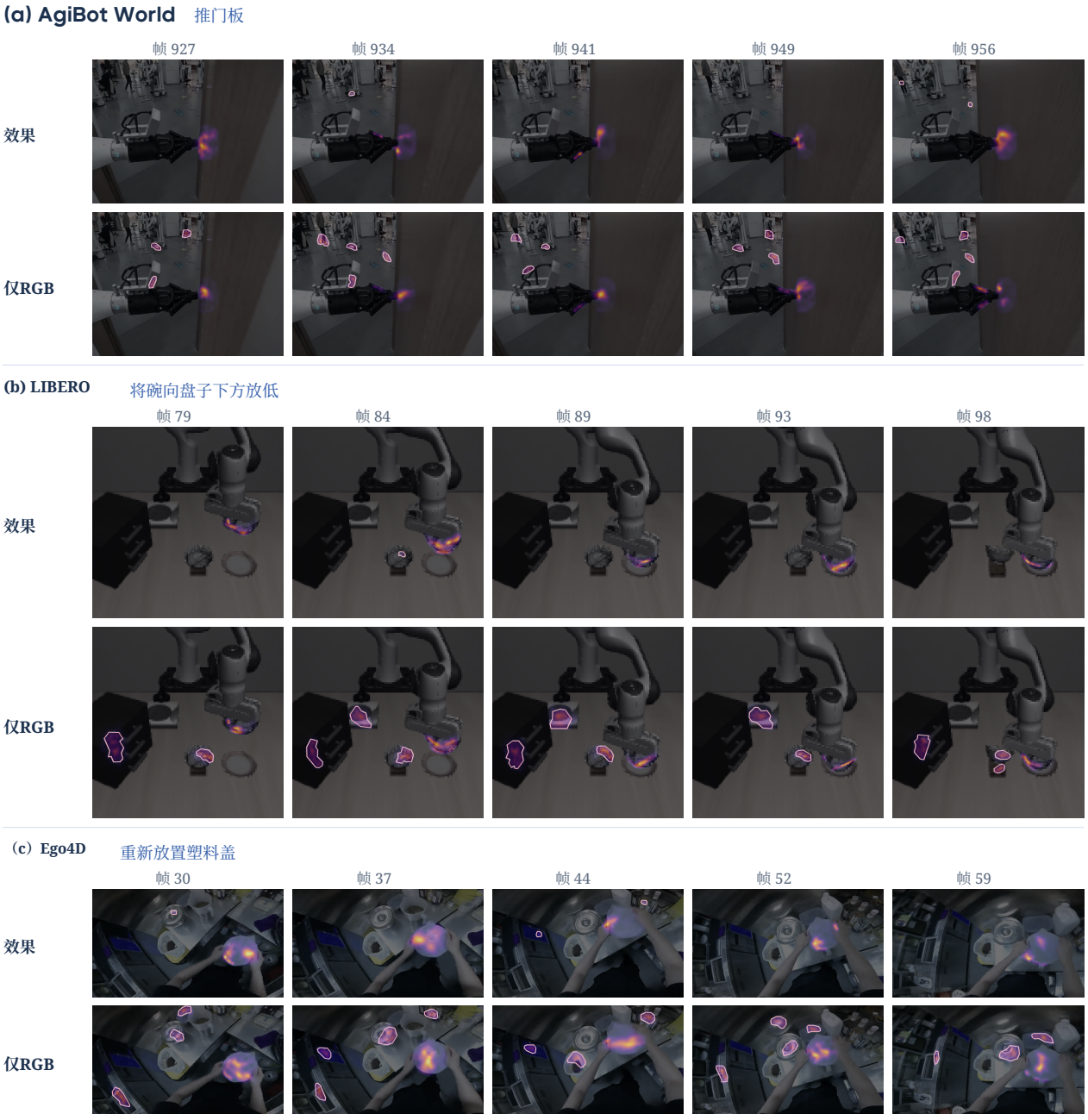

图5 EffectWAM Small（8B）的定性注意力比较：(a) 推门板，(b) 将碗向盘子放低，(c) 重新放置塑料盖。Effect表示完整模型中用于未来Effect预测的注意力；仅RGB表示未使用Effect监督信号进行未来RGB预测时的注意力。热图汇总了最后八层中指向历史输入帧的注意力：对各注意力头取平均，并对未来预测查询标记求和；两个变体使用相同的归一化方式和颜色刻度。成对的行按时间顺序展示相同帧，每列上方标注从零开始的源帧索引；采样帧并非连续帧。颜色越亮表示显示的注意力越强。浅粉色多边形轮廓标出具有非目标注意力的背景区域，以便更直观地比较。这些轮廓是额外添加的标注；注意力强度由底层热图表示。

实验凸显了较高的标准任务成功率、广泛的扰动覆盖范围以及更出色的组合式执行表现。EffectWAM Small 在 LIBERO 上的成功率达到 99.20%，在 LIBERO-Plus 上达到 93.10%；在 RoboTwin 的两种设置中均超过已报告的 InternW0-Δ 结果，并在 RoboCasa365 的三个划分中均提升了成功率。RoboCasa365 上的提升在未见过的组合中最为显著。在 RoboDojo 上，Small 的总体成功率达到 27.87%，记忆表现出色；Large 则达到 34.01%。

6 局限性与结论

局限性。在不同评估条件下，性能仍不均衡。RoboTwin 随机化条件下的性能下降、在未见过的 RoboCasa365 组合上的低成功率，以及 RoboDojo 开放任务的执行难度，都表明鲁棒性和泛化能力仍面临挑战。在 RoboDojo 的长时程任务中，部分得分与完整任务成功率之间的差距进一步凸显了持续保持有效行为、直至任务完成的必要性。

该方法还依赖几何和物理监督信号的可用性。CaPE 需要可靠的相机外参；对于未标定轨迹，则采用共享的无位姿路径。动作落地仅对可用且一致对齐的测量值进行监督，因此其对异构数据源中各类物理量的覆盖范围有限。在推理阶段，可选的视频生成使视频合成不进入控制路径，但联合进行效果–动作采样仍会带来计算成本。预测时域仍是需要配置的输入，不会根据子目标持续时间自动调整。

结论。我们提出了**EffectWAM**，这是一种世界–动作模型，通过 Effect、CaPE 和动作落地，将视觉变化预测与可执行的机器人控制联系起来。Effect 将视觉变化编码为相对于共享当前观测锚点的端点差异。具备标定信息时，CaPE 会在端点池化中纳入相对相机几何，而动作落地则使表征与测得的运动对齐。共享生成专家通过流匹配联合预测 Effect 和动作，并复用采样得到的动作与效果对，以支持可选的视频预测。在超过 20,000 小时的异构具身数据上进行训练，使这些组件能够在采用不同动作表示的机器人示范与人类演示中协同发挥作用。

在五个操作基准上的评估表明，该方法在标准任务上表现出色，并在 RoboCasa365 上取得了高于原始策略的成功率，其中对未见过的组合的提升最大。跨基线评估进一步表明，动作落地能够提升所有受评估基线策略的表现，支持其作为可迁移训练信号的作用。该方法仍易受环境变化影响，且难以完成复合任务，这促使我们进一步研究稳健的物理锚定、自适应动作时域以及更高效的联合采样。

参照方法

[1] M. J. Kim 等人，《OpenVLA：一种开源视觉-语言-动作模型》，载于机器人学习会议论文集（*CoRL*），Proc. Mach. Learn. Res. 系列，第 270 卷，2025年，第 2679–2713 页。[2] K. Black 等人，《π_0：一种用于通用机器人控制的视觉-语言-动作流模型》，载于 *Robotics: Science and Systems*（*RSS*）会议论文集，2025年，doi: 10.15607/RSS.2025.XXI.010。[3] S. Ye 等人，《世界动作模型是零样本策略》，2026年，arXiv:2602.15922。[4] X. Miao 等人，《InternW0-Δ：一种以 20K+ 小时开放数据连接预测动力学与动作的世界动作模型》，2026年，arXiv:2609.31394。[5] T. Yuan、Z. Dong、Y. Liu 和 H. Zhao，《Fast-WAM：世界动作模型需要测试时未来想象吗？》，2026年，arXiv:2603.16666。[6] X. Li 等人，《将潜在动作视为意图，使世界动作模型能够高效地进行未来想象》，2026年，arXiv:2608.24882。[7]L. Mur-Labadia 等人，《V-JEPA 2.1：释放视频自监督学习中的密集特征》，2026年，arXiv:2603.14482。[8] M. Clémente 等人，《La-JEPA：基于潜在世界建模的紧凑型机器人策略》，2026年，arXiv:2611.08156。

[9] H. Yuan 等人., “Qwen-机器人Manip tec hnical report : Ali gnment unlocks sca le for robotic manipulation foundation 模型”，2026年，arXiv:2606.17846.[10] B. Liu 等，“LIBERO：面向终身机器人学习的知识迁移基准测试”，载于《Adv. Neural Inf. Process. Syst.》，第36卷，2023年，第44 776–44 791页，doi: 10.52202/075280-1939.[11] S. Fei 等，“LIBERO-Plus：视觉-语言-动作模型的渐进式鲁棒性基准”，载于《Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)》，2026年，第38 574–38 583页。[12] T. Chen 等，“RoboTwin 2.0：一种用于鲁棒双臂机器人操作、具备强领域随机化能力的可扩展数据生成器与基准测试”，载于《Proc. 43rd Int. Conf. Mach. Learn. (ICML)》，丛书《Proc. Mach. Learn. Res.》，第306卷，2026年，第13 673–13 699页。[13] S. Nasiriany、S. Nasiriany、A. Maddukuri 和 Y. Zhu，“RoboCasa365：用于训练和评估通用机器人的大规模仿真框架”，载于《Proc. Int. Conf. Learn. Represent. (ICLR)》，2026年。[14] T. Chen 等，“RoboDojo：用于全面评估通用机器人操作策略的统一仿真与现实基准”，2026年，arXiv:2607.04434。[15] Y. Du 等，“通过文本引导的视频生成学习通用策略”，载于《Adv. Neural Inf. Process. Syst.》，第36卷，2023年，第9156–9172页，doi: 10.52202/075280-0403。[16] S. Ye 等，“基于视频的潜在动作预训练”，载于《Proc. Int. Conf. Learn. Represent. (ICLR)》，2025年。[17] J. Liu 等，“WALA：从带动作标签的演示和无动作标注视频中学习可执行潜在动作”，2026年，arXiv:2607.11397。[18]P. Liu 等，“WLA^3：面向语义、动力学和运动学的世界潜在动作建模”，2026年，arXiv:2609.15870。[19] X. Kong、S. Liu、X. Lyu、M. Taher、X. Qi 和 A. J. Davison，“EscherNet：用于可扩展视图合成的生成模型”，载于《Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)》，2024年，第9503–9513页。[20] R. Li、B. Yi、J. Liu、H. Gao、Y. Ma 和 A. Kanazawa，“作为相对位置编码的摄像头”，载于《Adv. Neural Inf. Process. Syst.》，第38卷，2025年，第18 020–18 045页，doi: 10.52202/085713-0540。[21] M. J. Kim、C. Finn 和 P. Liang，“视觉-语言-动作模型微调：优化速度与成功率”，载于《Proc. Robotics: Science and Systems (RSS)》，2025年，doi: 10.15607/RSS.2025.XXI.017。[22] Q. Bu 等，“利用任务中心的潜在动作学习在任意场景中行动”，载于《Proc. Robotics: Science and Systems (RSS)》，2025年，doi: 10.15607/RSS.2025.XXI.014。[23] S. Tan、K. Dou、Y. Zhao 和 P. Krähenbühl，“视觉-语言-动作模型的交互式后训练”，2025年，arXiv:2505.17016。[24] C.-Y. Hung 等，“NORA-1.5：使用世界模型与基于动作的偏好奖励训练的视觉-语言-动作模型”，2025年，arXiv:2511.14659。[25] S. Fei 等，“SRPO：面向视觉-语言-动作模型的自指策略优化”，2025年，arXiv:2511.15605。[26] L. Xiao 等，“AVA-VLA：通过主动视觉注意力改进视觉-语言-动作模型”，载于《Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)》，2026年，第13 453–13 463页。[27] C.-Y. Hung 等，“NORA：面向具身任务的小型开源通用视觉-语言-动作模型”，2025年，arXiv:2504.19854。[28] K. Pertsch 等，“FAST：视觉-语言-动作模型的高效动作词元化”，载于《Proc. Robotics: Science and Systems (RSS)》，2025年，doi: 10.15607/RSS.2025.XXI.012。[29] L. Yang 等，“4D-WAM：通过轨迹场为世界动作模型注入时空感知能力”，2026年，arXiv:2608.08023。[30] Y. Wang 等，“OpenWAM：面向系统化世界-动作模型预训练的开放式模块化探索”，2026年，arXiv:2609.07398。[31] Y. Zou、Y. Yao、G. Jiao、J. Ma、G. Zheng 和 X. Gao，“PICA-WAM：通过世界动作模型中的跨外观自对齐实现视觉扰动不变性”，2026年，arXiv:2610.26813。[32] L. Li 等，“面向机器人控制的因果世界建模”，2026年，arXiv:2601.21998。[33] 小米机器人团队，“Xiaomi-Robotics-1：利用超过10万小时的真实世界轨迹扩展视觉-语言-动作模型”，2026年，arXiv:2607.15330。[34] D. Kim 等，“RLDX-1技术报告”，2026年，arXiv:2605.03269。

[35] Awomo-WAM Team，《Awomo-0.5 技术报告》，Awomo-WAM Team，技术报告，2026年10月。 [在线]。获取地址：https://awomo-westlakedi.github.io/Awomo-0.5-page/assets/awomo-0.5-report.pdf[36] T.Xin 等，《PrismWAM：解耦世界动作模型中的状态表征》，2026年，arXiv:2610.26564。[37] Y.Guo 等，《视频预测策略2：预测得更好，行动得更好》，2026年，arXiv:2610.10270。[38] Q. Bu 等，《AgiBot World Colosseo：用于可扩展智能具身系统的大规模操作平台》，载于 *IEEE/RSJ* 智能机器人与系统国际会议论文集（*IROS*），2025年，第3549–3556页。[39] K. Grauman 等，《Ego4D：在3,000小时第一人称视角视频中环游世界》，载于 *IEEE/CVF* 计算机视觉与模式识别会议论文集（*CVPR*），2022年，第18 995–19 012页。

预训练数据统计

预训练数据包含 34 个来源，记录总时长为 25,292.8 小时。表 9列出了构成图 3 中数据的各来源时长。未使用的思维链数据不计入统计，同一来源的不同处理版本仅计数一次。

表 9 各预训练数据来源的记录时长（单位：小时）。

来源	小时数
Builddotai Egocentric-10K	9,984.1
HY-Embodied-0.5	2,158.8
HiFi-UMI-2K	2,131.3
ABC-130K	1,664.9
DROID	1,276.1
Jianzhi v1	1,156.5
Mifeng batch2	994.8
Lightwheel EgoProStandard	994.3
InternData M1 Franka	848.3
InternData A1	820.3
RH20T	784.7
EgoDex v1	724.2
Fractal	350.2
RoboCOIN	215.6
BC-Z	165.8
GM100	151.9
BitRobot HIW-500	127.9
Mifeng 批次1	111.6
FurnitureBench	109.7
EgoVerse Lightwheel 第1部分	100.4
EgoVerse Aria v1	74.6
InternData M1 AgileX	62.5
PineLab ManuFacet	49.3
EgoVerse Mecka v1	38.3
Imitator 机器人	37.0
FineArt 90Hz	33.3
Imitator 仿真	30.9
RoboMIND2 AgileX	28.4
RoboDojo（仿真）	20.5
EgoVerse Lightwheel 第 2 部分	12.6
GPT-Astra（机器人操作）	10.5
Stanford Hydra v2	10.0
Imitator 人类 v1	8.1
伯克利 UR5	5.4
总计	25,292.8

B RoboTwin 2.0 各任务结果

表10 EffectWAM Small（8B）在 RoboTwin 2.0 各任务上的成功率（%）。

Task	干净设置→干净设置	干净设置→随机	Task	干净设置→干净设置	干净设置→随机
调整 _瓶子	100	96	放置 _容器 _盘子	99	80
beat _方块 _锤子	99	74	放置 _dual_鞋子	92	61
方块 _排名 _rgb	91	85	放置 _空的 _cup	100	75
方块 _排名 _size	71	42	放置 _fan	96	77
点击 _闹钟	100	90	放置 _汉堡包 _薯条	100	98
点击 _bell	100	91	放置 _鼠标 _pad	91	71
dump_bin_大垃圾箱	99	94	放置 _物体 _篮子	87	64
grab_滚轮	100	77	放置 _物体 _规模	98	83
交接 _方块	93	67	放置 _物体 _支架	98	87
交接 _mic	99	82	放置 _手机 _支架	92	64
lift_pot	100	81	move_药瓶 _pad	100	71
move_can_pot	94	65	放置 _shoe	96	80
move_扑克牌 _away	98	73	按压 _订书机	91	80
move_订书机 _pad	74	65	put_瓶子 _垃圾桶	93	88
悬挂 _mug	56	40	put_物体 _柜子	67	38
open_笔记本电脑	94	78	旋转 _二维码	93	48
open_微波炉	68	43	scan_物体	90	59
pick_多样的 _瓶子	91	78	摇动 _瓶子	100	99
pick_dual_瓶子	100	81	摇动 _瓶子 _水平地	100	97
放置 _a2b_left	92	70	堆叠 _方块 _三个	99	81
放置 _a2b_右侧	94	64	堆叠 _方块 _two	100	89
放置 _面包 _篮子	93	91	堆叠 _碗 _三个	75	70
放置 _面包 _煎锅	96	78	堆叠 _碗 _two	88	77
放置 _can_篮子	75	67	邮票 _seal	94	65
放置 _cans_塑料盒	98	93	turn_开关	77	44

C AdditionalRoboDojo 执行

图6–11展示了 EffectWAM Large (26B) 的另外 25 次成功执行，与图 4 中的五个示例相辅相成。这些序列遵循 RoboDojo 的能力分组，涵盖标准和随机化设置下的衣物折叠任务。每一行代表一个单独的记录回合；所选帧保留了初始状态、中间交互和完成状态。索引对应从零开始计数的源视频帧，所选事件之间的间隔并不相等。蓝色边框细节取自同一帧中以轮廓标出的区域。



图6 在 fold_clothes 任务中的其他泛化执行，回合为 0、7、10、19 和 24。面对不同的服装外观，机器人先向内折叠袖子，再用两个夹爪折叠下摆。中间帧展示了两次袖子折叠；最后一列展示松开的衣物。

场景变化下的泛化。随机折叠回合会改变衣物姿态和周围环境的外观，同时保持相同的折叠目标。图 7 展示了彩色照明、不同的桌面纹理以及附近物体。因此，抓取位置会随衣物而变化，但操作序列仍依次完成两次袖子折叠，最后折叠下摆。



图7 折叠_衣物_随机场景的更多泛化执行，回合5、7、13和21。工作空间裁剪图保留了周围的物体和视觉条件，而不是仅截取衣物。每行按照时间顺序展示一个回合，最后一帧显示释放后的折叠结果。

重复转移过程中的进展。 瓶子处理任务要求将全部四个瓶子送入垃圾桶。图 8 中的回合改变了瓶子形状、位置和初始姿态。靠近垃圾箱的瓶子可以直接转移，而工作空间另一侧的瓶子则需要在夹爪之间传递。中间帧记录了剩余物体数量的减少过程，其中包括在工作空间清空前对最后一个瓶子的交接。



图8 将_瓶子_放入_垃圾桶的更多长时域执行，回合 1、30、42 和 47。事件帧展示了弃置进度和机械臂间交接。终止视图展示了四瓶序列结束后清空的桌面；每一行都保留了一个固定裁剪区域，其中包含左侧的垃圾桶。

遮挡后的颜色顺序。 在图 9中，初始颜色排列各不相同，但机器人首先从左侧到右侧遮挡方块，随后依次揭示红色、绿色和蓝色。全遮挡帧移除了可见的颜色线索。后续帧展示了由此产生的空间上不同的揭示序列，明确体现了其对先前颜色—位置排列的依赖。每一行中的细节窗口始终使用相同的图像区域，因此可以直接比较可见方块和被遮挡位置。

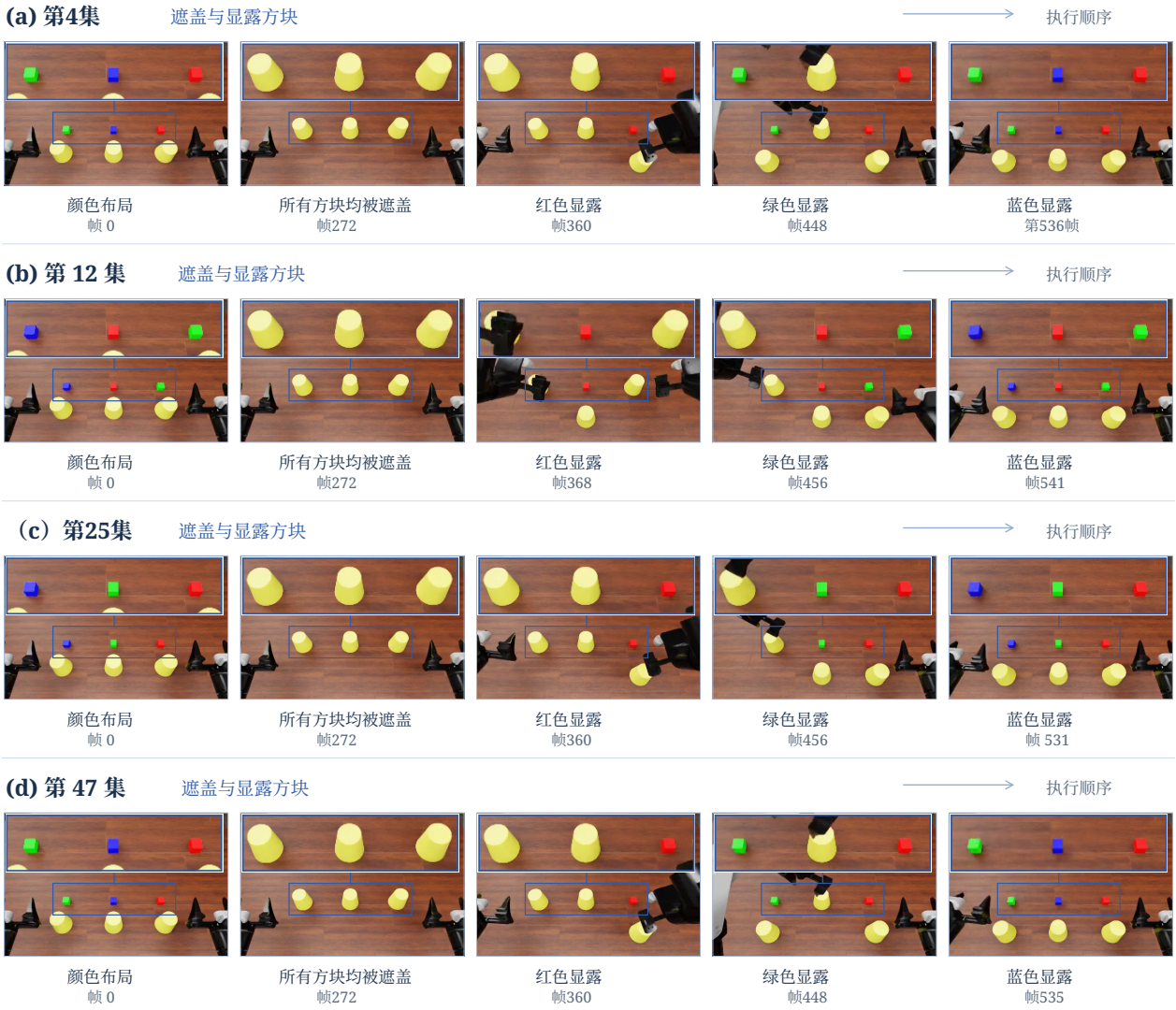

图9 在覆盖_方块的回合4、12、25和47中执行附加记忆。各列展示初始颜色、完全遮挡状态，以及显露红色、绿色和蓝色后的状态。揭示顺序遵循颜色而非固定的从左到右扫描；最后一帧还显示杯子已移回，远离方块。

通过工具接触实现对齐。 图 10 展示了不同的初始尺子方向和方块偏移。尺子提供了一个共享接触面，使方块无需逐个抓取即可对齐。第4集还包括从尺子一端进行的额外调整，而其他三个回合则通过连续的推送序列达到对齐构型。放大的终端视图让最后的方块排列更易于检查。



图10 关于对齐_方块的其他开放执行，回合为 4、6、12 和 49。事件帧保留了初始排列、工具接触、推动或调整，以及最终对齐。蓝色边框细节放大显示末端方块及其与尺子的关系。

逐层搭建支撑结构。 在图 11中，机械臂分别放置四个白色支撑件，并将水平平台一起搬运。随着结构不断增高，它们的位置必须支撑下一层。两个顶部组件的位置也各不相同：在这些最后的放置操作中，负责执行每次放置的机械臂会随回合而变化。所选的中间状态展示了已完成的下层平台、上部支撑结构和上层平台；最后一帧展示了夹爪撤回后的完整的塔。



图11 build_tower 上的更多精准执行，回合 0、6、10 和 18。每个序列都会组装四个支撑块、两个水平平台和两个顶部组件。中间帧展示了逐步形成的支撑层级，最后的细节图突出显示已完成的八件结构。