Bootstrapping World Models from Dynamics Models in Multimodal Foundation Models

Anonymous Author(s)
Affiliation
Address
email

Abstract

To what extent do vision-and-language foundation models possess a realistic world model (observation $\times$ action $\rightarrow$ observation) and a dynamics model (observation $\times$ observation $\rightarrow$ action), when actions are expressed through language? While open-source foundation models struggle with both, we find that fine-tuning them to acquire a dynamics model through supervision is significantly easier than acquiring a world model. In turn, dynamics models can be used to bootstrap world models through two main strategies: 1) weakly supervised learning from synthetic data and 2) inference time verification. Firstly, the dynamics model can annotate actions for unlabelled pairs of video frame observations to expand the training data. We further propose a loss-weighting mechanism for the image tokens weighted by the its importance predicted by a recognition model. Secondly, the dynamics models can assign rewards to multiple samples of the world model to score them, effectively guiding search at inference time. We evaluate the world models resulting from both strategies through the task of *action-centric image editing* on AURORA-BENCH. Our best model achieves a performance competitive with state-of-the-art image editing models, improving on them by a margin of 15% on real-world subsets according to GPT4o-as-judge, and achieving the best average human evaluation across all subsets of AURORA-BENCH.[1].

1 Introduction

World models (observation $\times$ action $\rightarrow$ observation) [1, 2, 3, 4] can be successfully trained to simulate future trajectories given the history of past observations and actions. World models are instrumental in training embodied agents to endow them with specific abilities [5], such as grounding on affordances [6], spatio-temporal reasoning [7, 8], and planning [9, 10, 11]. However, learning a specialised world model is challenging. Firstly, it requires a large amount of real-world data [12] and even this data volume may be insufficient within the confines of the current training paradigm [13]. Secondly, the benefit of creating a separate world model to train a downstream embodied agent remains unclear because of possible compounding errors between the two models. Conversely, foundation models, such as vision-language models (VLMs), are already imbued with plenty of real-world knowledge of both action (in language form) and perception (in vision form), because of their large-scale pre-training. While such knowledge is not straightforward to elicit [14, 15, 16], we propose investigating a promising alternative to specialised world models, by enhancing the knowledge implicitly stored inside foundation models.

Firstly, we probe whether native VLMs already contain reliable world models, facilitated by model designs that combine various modalities into a unified representation, i.e., sequences of tokens [17]. In particular, we frame the assessment of world models as the ability to solve *action-centric image editing*

[1]The code and models used in this paper will be available at `[anonymised]`.

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

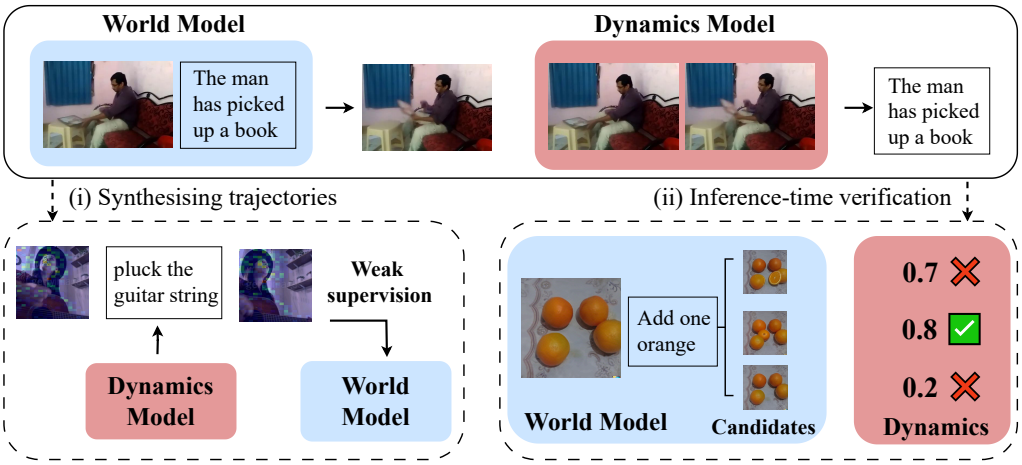


Figure 1: Illustration of our two strategies to bootstrap a world model from a dynamics model in Vision-Language Models: (i) synthesising trajectories for weak supervision (**left**) and (ii) inference-time verification of candidate observations (**right**).

tasks [18]. In such tasks, the model predicts the next observation given the previous observation and an action expressed as a language instruction. Based on our evaluation, we empirically demonstrate that existing open-source models do not prefer ground-truth trajectories compared to adversarially generated ones. Hence, we verify that the world model implicit in the original VLMs *per se* is not well grounded on real-world trajectories [14, 15, 16].

Surprisingly, we also find that acquiring a dynamics model (observation $\times$ observation $\rightarrow$ action) via supervised fine-tuning is substantially easier than directly acquiring a world model (observation $\times$ action $\rightarrow$ observation). Inspired by this observation, we propose two strategies to bootstrap the world model from the dynamics model in a given VLM, namely (i) **learning from synthetic trajectories** in videos automatically labelled with actions by the dynamics model; and (ii) **test-time verification** of predicted observations sampled from the world model through the dynamics model.

For the weak supervision strategy, which is reminiscent of [19], we use a dynamics model fine-tuned on the AURORA dataset [18] to annotate motion key-frames pairs extracted from real-world videos with actions (in language form). Around 45 hours of unlabelled videos are sourced from movements-in-time [20], Kinetics700 [21, 22] and UCF-101 [23]. Together with the ground-truth trajectories in AURORA, the synthesised trajectory triplets (observation $\times$ annotated action $\rightarrow$ observation) are then used for supervised fine-tuning of the VLM world model. To effectively train the world model, we additionally propose a *loss-weighting method* which weights the loss of each image token according to the visual difference between the ground-truth source and target observations, as estimated by a recognition model. In the verification strategy, we show how using the VLM dynamics model to assign rewards to multiple samples generated by the VLM world model can effectively guide search at inference time.

We conduct an extensive evaluation on MagicBrush, Action-Genome, Something-Something, What-sUp and Kubric in AURORA-BENCH [18]. We focus on Chameleon-7B as the best available open-source foundation model, and transform it into a world model (**CWM**; **C**hameleon **W**orld **M**odel). We show that thanks to the synthetic data strategy to bootstrap world models from dynamics models, our general-purpose CWM can achieve an overall performance superior to state-of-the-art diffusion models specialised for image editing. In particular, CWM improves GPT4o-as-a-judge scores on the Something-Something, Action-Genome, and Kubric subsets of AURORA by 15%, 15% and 7%, respectively. Similarly, human evaluators rate CMW image editing consistently better. Inference-time verification can also improve AURORA-finetuned Chameleon to a comparable degree as data synthesis, providing an effective training-free bootstrapping method. In some cases, it can even be combined with data synthesis for compounded gains.

To summarise our contributions:

Table 1: Model preference percentages (Reference vs. Various Negatives) across tasks for 9 VLMs. Higher values indicate stronger preference for reference.

Model	World Modelling (WM)				Inverse-dynamics Modelling (IDM)			
	Rand. Act.	Inv. Obs.	Copy. Obs.	Rand. Obs.	Rand. Act.	Inv. Obs.	Copy. Obs.	Rand. Obs.
Qwen2-VL-2B [24]	36.69	53.23	54.03	42.74	58.87	53.63	54.03	47.58
Qwen2-VL-7B [24]	36.69	50.40	56.85	38.31	59.68	54.84	64.52	50.40
Qwen2.5-VL-3B [25]	31.04	52.42	53.63	42.74	56.45	55.24	64.91	42.34
Qwen2.5-VL-7B [25]	43.55	55.24	81.05	36.29	60.08	50.40	67.34	43.15
LLaVA-Next-7B [26]	48.79	54.44	48.79	51.61	55.65	49.60	48.39	52.02
LLaVA-Interleave [27]	56.45	46.77	31.85	58.87	56.85	47.58	30.65	58.06
Qwen-Omni-3B [28]	29.84	46.37	74.19	35.89	59.27	48.39	55.65	54.44
Qwen-Omni-7B [28]	40.32	48.39	66.94	39.92	58.06	50.81	55.24	50.40
Chameleon-7B [17]	44.80	52.00	100.0	46.40	55.60	50.80	42.70	58.10

- We empirically show that VLMs like Chameleon-7B do not exhibit a clear preference for ground-truth real-world trajectories over heuristic-generated incorrect ones.
- We propose two strategies to bootstrap a world model from a dynamics model inside VLMs: (i) learning from unlabelled videos annotated with actions by a dynamics model, and (ii) verifying the generated observations with the dynamics model at inference time.
- We conduct extensive evaluations on AURORA-BENCH: both GPT4o-as-a-judge and human raters demonstrate the effectiveness of our methods with a considerable margin compared to the state-of-the-art image editing models.

2 VLMs Lack a Consistent Preference for Real-World Trajectories

The first research question we investigate in this paper is: **To what extent do VLMs exhibit a preference for token sequences of actions and observations that align with real-world trajectories?**

To address this question, we evaluate 9 VLMs on ground-truth trajectories from 5 subsets of AURORA-BENCH [18]: MagicBrush, Something-Something, Action-Genome, Whatsup, and Kubric. Each subset contains 50 trajectory triplets of the form (o_s, a, o_t), where o_s is the source observation, a the action text, and o_t the next observation. [2]

We then manually curate four types of negative trajectories using rules: two that manipulate the observation of the trajectory triplet, and two that manipulate the action. We design two kinds of action-level manipulation: 1) **Random Action**: for a given pair of observations, we substitute the original action with another randomly sampled within the same subset. 2) **Random Observation**: we randomly substitute the target observation with another in the same subset. We also test the following observation-level manipulations. 3) **Copy Observation**: we directly copy the source observation as the target observation. 4) **Inverse Observation**: we swap the source and target observations.

In Table 1, we compare the negative log-likelihood VLMs assign to each ground-truth trajectory against its corresponding manipulated one. We evaluate the VLMs in two tasks: action prediction (i.e., as a dynamics model) and next-observation prediction (i.e., as a world model). For each kind of negative trajectory, we report the percentage of samples where the model favours the reference trajectory over the negative trajectory. From Table 1, it emerges that VLMs display a very limited preference for the ground-truth trajectories in a zero-shot setting (around 50%). In the action prediction task (right panel), there is a slightly higher tendency to favour the ground-truth over the group with random actions; however, even in the best case, Qwen2.5-VL-7B prefers the reference in only 60.08% of the samples. The only negative group that seems to be identifiable for VLMs is the inverse observation, where Qwen2.5-VL-7B has 67.34% of correct preference. In the next-observation prediction task (left panel), the VLM mostly fails in effectively differentiating the ground truth from the negatives. An exception to this is the copy manipulation, where the Chameleon can always tell them apart. Although the underlying reason remains uncertain, one plausible explanation for this behaviour is that the model's ability to solve next-observation prediction tasks depends on their alignment with training sequences: for instance, it is plausible that Chameleon's data rarely features two identical adjacent images. We provide a breakdown discussion for Chameleon in Appendix A.5.

[2] We choose these 9 VLMs with the consideration of 1) they are public accessible and 2) we ensure that they have been exposed to interleaved data during their pre-training.

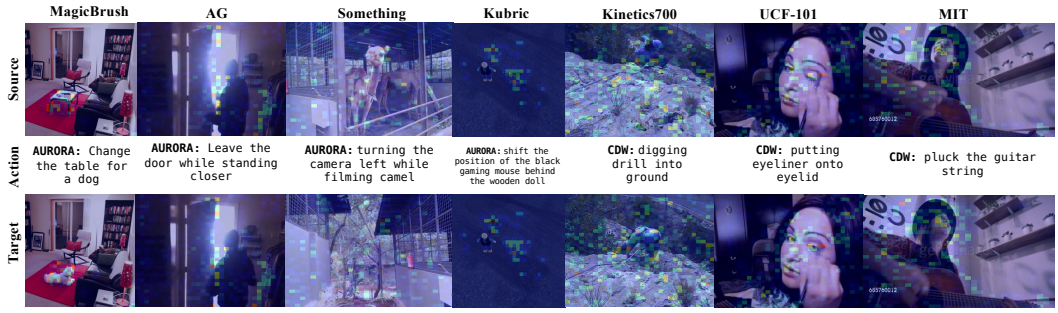


Figure 2: Heatmap visualization of image token weights predicted by the recognition model on examples from UCF-101, Something-Something, MagicBrush, and Kubric.

3 Bootstrapping a World Model from a Dynamics Model in VLMs

Since we showed in Section 2 that Chameleon-7B displays a higher proclivity as for action prediction than next-observation prediction, we first verify that this tendency is intensified when Chameleon-7B is fine-tuned on image editing trajectories (Section 3.1), as this results in the VLM acting reliably as a dynamics model. Motivated by this, we propose two strategies to leverage the VLMs as dynamics models to enhance VLMs as world models: (i) generating synthetic trajectories by annotating large-scale key-frame pairs from videos with actions predicted by the dynamics model, then using these as weak supervision to train the world model (Section 3.2); and (ii) using the dynamics model as a verifier at test time to score candidate next observations sampled from the world model (Section 3.3).

3.1 Fine-tuning Chameleon as a Dynamics Model

First, we fine-tune Chameleon as a Dynamics Model (**CDM**) $p_{\text{CDM}}(a \mid o_s, o_t)$, which predicts the probability of an action given the previous and next observations. As training data, we rely on high-quality triplets from AURORA [18] and the action recognition track of EPIC-Kitchen [29], which is based on videos with an egocentric view. We use the first and last frame in the EPIC-Kitchen video clips as the source and target observation o_s and o_t and the annotated action as a. We provide full details on CDM training data and experimental setting Appendix A.9.1. Foreshadowing the results in Section 4.2, this significantly enhances action-prediction capabilities of Chameleon by a wide margin.

3.2 Weakly Supervised Learning from Unlabelled Videos

Synthetic Trajectories. Taking advantage of the resulting high-quality CDM, we then explore the first of our strategies to bootstrap a world model in VLMs: we annotate pairs of motion key-frames of unlabelled videos with a textual description of the action with the CDM. To ensure both scale and quality, we collect approximately 45 hours of video from Moments-in-Time [20], Kinetics-700 [21, 22], and UCF-101 [23], all of which consist of curated clips focused on human actions. To ensure the selected pairs of motion key-frames are meaningful, i.e., they express a valid action, we then calculate the optical flow to quantify the dynamics per frame in the video clips, and select the top-K_f frames while ensuring that the interval between two selected frames is I_f. Specifically, we set $I_f = 20$ and $K_f = 6$ for all three datasets. This results in approximately 20K, 46K, and 21K annotated trajectory triplets from Moments-in-Time, Kinetics-700, and UCF-101, respectively. Finally, we apply a filtering strategy to further guarantee the quality of the resulting triplets. We use the CDM's predicted likelihood for each trajectory triplet $(o_s, a_{\text{CDM}}, o_t)$ as a score, and apply stratified Top-K sampling[3] to select a subset of CDM-annotated trajectory triplets. We show statistics of the scores and action classes for the selected triplets in Figure 10. We also provide one example for each dataset in Figure 2.

Fine-tuning Chameleon as a World Model. Afterwards, we fine-tune Chameleon as a World Model (**CWM**), $p_{\text{CWM}}(o_t \mid a, o_s)$ on both AURORA's supervised triplets $\mathcal{D}_{\text{sup}}$ and unsupervised triples $\mathcal{D}_{\text{unsup}}$ with actions sampled from the CDM. The world model CWM is trained with maximum

[3]The details of this algorithm are provided in Appendix A.6.

likelihood estimation as an objective:

$$\min_{\theta} \mathbb{E}_{(a,o_s,o_t)\sim\mathcal{D}_{\text{sup}}}\left[-\log p_\theta(o_t \mid a, o_s)\right] + \mathbb{E}_{(o_s,o_t)\sim\mathcal{D}_{\text{unsup}}}\left[\mathbb{E}_{\hat{a}\sim p_{\text{CDM}}(a|o_s,o_t)}\left[-\log p_\theta(o_t \mid \hat{a}, o_s)\right]\right], \tag{1}$$

where θ are the parameters for CWM, and $\hat{a}$ is action sampled from the CDM.

Recognition-Weighted Training Loss. Nevetheless, the objective in Equation 1 is limited by treating all regions of the target observation equally, even if some of them remain identical to the source whereas others change. This may result in degenerate solutions such as always copying the source. As an alternative, we therefore propose a novel training objective for world models that overcomes this assumption. This objective weights the loss of next-observation image tokens based on their importance. The intuition is that not all image patches in source and target observations contribute equally to modelling real-world transitions; instead, the model should focus on patches most indicative of the action's consequences. To this end, we leverage a recognition model $f_{\text{rec}}(w|o_s, o_t)$, which outputs token-level weights aligned with Chameleon's image token representations. These weights modulate the loss during training, emphasising learning on semantically meaningful regions and down-weighting irrelevant ones. We formulate our alternative objective as:

$$\min_{\theta} \sum_{l=1}^{L} f_{\text{rec}}(w|o_s, o_t)^{(l)} \cdot \left(-\log p_\theta(o_t^{(l)} \mid o_t^{(<l)}, o_s, a)\right), \tag{2}$$

where θ are the parameters of CWM and a L is the number of tokens used to represent an image in Chameleon. $o_t^{(l)}$ and $o_t^{(<l)}$ represent the image tokens of o_t at position l and the history of previous positions, respectively. For simplicity, we use the pre-trained vector-quantised model of Chameleon as the recognition model, by computing the squared L_2 norm of pre-quantized features $\mathbf{z}_{o_s} \in Z_{o_s}$ and $\mathbf{z}_{o_t} \in Z_{o_t}$ where Z_{o_s} and Z_{o_t} are the sets of features of source and target observations, respectively. We visualise the token weights in Figure 2, which capture the effects of acting on the source observation to yield the target one.

3.3 Test-time Verification

Finally, we introduce an inference-time strategy which harnesses the CDM as a verifier to enhance CWM performance. Inspired by recent work on scaling test-time compute [30, 31], we let the CWM generate N candidate observations. Each candidate is paired with the source and scored by the CDM, which assigns each a predicted likelihood, interpreted as a reward. The final prediction of the CWM is selected by maximising the CDM's reward:

$$\hat{o}_t = \underset{i\in\{1,\ldots,N\}}{\text{argmax}}\ p_{\text{CDM}}\left(a \mid o_s, o_t^{(i)}\right), \quad \text{where } o_t^{(i)} \sim p_{\text{CWM}}(o_t \mid o_s, a),$$

where $\hat{o}_t$ is the selected prediction.

4 Experiments and Results

4.1 Experimental Setting

Benchmarks. We select AURORA-BENCH [18] for evaluation of both dynamics and world models. This dataset provides high-quality data for action-centric edits, covering a wide array of phenomena and assessing a model's alignment with the physical world, including temporal and spatial reasoning. We choose 5 subsets: **MagicBrush** for specialised image editing, **Action-Genome (AG)** and **Something-Something (Something)** for real-world actions and scenarios. **Whatsup** focuses on spatial reasoning, whereas **Kubric** contains synthetic samples from a physical engine [32].

Baselines. We report Chamelon's zero-shot performance (**C-ZS**). We also fine-tune Chameleon on AURORA's training set as our first baseline (**C-FT**). We compare CWM with C-FT in both a single-prediction setting and in a best-of-N setting. The latter provides a ceiling performance for inference-time verification with CDM. Additionally, we include three state-of-the-art diffusion models specialised for image editing as baselines, such as **PixInstruct** [33], **GoT** [34] and **SmartEdit** [35]. As a sanity check, we also report the metric scores obtained by simply copying the source observation input as the next-observation prediction (**Copy**).

Table 2: Performance of dynamics models performance on action prediction, measured by text similarity metrics: BERTScore (BS; [36]), ROUGE-1/2/L (R-1, R-2, R-L; [37]) and BLEU [38].

	BS	R-1	R-2	R-L	BLEU
VILA-U Fine-tuned	0.40	0.38	0.20	0.37	0.15
Chameleon Zero-Shot (C-ZS)	0.05	0.09	0.02	0.08	0.00
Chameleon Fine-Tuned (CDM)	0.40	0.39	0.20	0.37	0.17
Chameleon Fine-Tuned (CDM) + DS	**0.45**	**0.45**	**0.27**	**0.44**	**0.20**

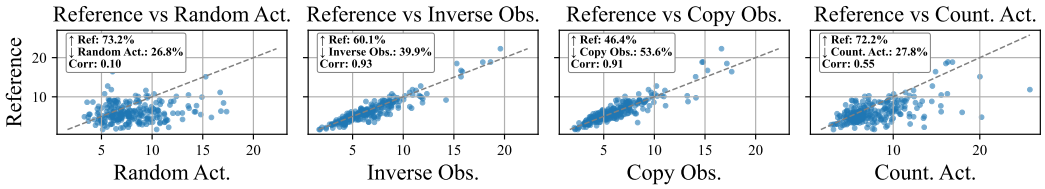


Figure 3: Comparison of negative log-likelihoods (lower values indicate stronger model preference) of the action predicted by CDM for ground-truth trajectories versus four types of negative trajectories.

Metrics. For next-observation prediction evaluation, following [34], we rely on **GPT4o-as-a-judge** as it is the only metric that reliably penalises Copy. In Appendix A.3, we show four other metrics, e.g., CLIP, which assign high scores to Copy. GPT4o-as-a-judge scores consider two criteria, one for the editing success rate and one for visual consistency with the original. We take the minimum of the two as the final score. The prompt for GPT4o-as-a-judge is provided in Appendix A.7.

4.2 Chameleon Dynamics Model

We evaluate the dynamics models based on the textual similarity of the predicted action with the ground-truth action in AURORA-BENCH, as shown in Table 2. Our results demonstrate that fine-tuning is necessary to elicit Chameleon's ability to verbalise the dynamics from two observations. We then compare Chameleon fine-tuned on action prediction (CDM) with the fine-tuned version of another state-of-the-art VLM, VILA-U [39]. CDM is on par or superior to VILA-U fine-tuned, justifying our choice of Chameleon as a foundation model for our experiments. Table 2 also provides an ablation showing that downsampling trajectories from Kubric (DS) in the training data further boosts CDM performance (CDM + DS). This suggests that data sourced from simulations do not necessarily translate into better dynamics modelling in real-world examples. We use the DS version of CDM in the rest of the experiments as the best-performing dynamics model. In Figure 3, we further evaluate CDM on discriminating between ground-truth and negative trajectories, as in Section 2. Now, we observe that CDM is mostly successful in identifying manipulated trajectories as such, except for Copy. These results corroborate the feasibility of annotating actions for key-frame pairs.

4.3 Chameleon World Model

Automatic evaluation. Next, we test CWM on next-observation prediction for each of the AURORA-BENCH subsets, reporting GPT4-as-a-judge scores in Figure 3. We first notice that the state-of-the-art image editing models (i.e., PixInstruct, GoT, SmartEdit) tend to specialise in the image editing benchmark, MagicBrush (5.96 and 6.71 GPT4o scores for GoT and SmartEdit). Nevertheless, in the action-centric subsets, including Action-Genome (AG), Something and Kubric, they are mostly behind CWM and even C-FT. In particular, CWM outperforms all other models in these 3 subsets, achieving gains of 18%, 4%, and 86%, respectively, over the best diffusion baselines. In addition, it boasts the highest average performance across subsets, with an 8% increase. Crucially, comparing CWM and C-FT reveals the benefit of augmenting the training data with synthetic triplets bootstrapped from the CDM, as it yields a 13% performance margin. CWM also outperforms C-FT on the best-of-N setting [40], indicating the potential for inference-time verification as best-of-N is effectively an oracle for its performance.

Table 3: Model performance on MagicBrush, AG, Something, WhatsUp, and Kubric from AURORA-BENCH in terms of GPT-4o scores. For C-FT and CWM, we report their performance for both single prediction and *best-of-N*. The average scores for each model are shown at the bottom. We **bold** the best model overall for each subset and highlight the best and worst scores among our variants for each setting. SE: SmartEdit.

Datasets	Models								
	Copy	**PixInstruct**	**GoT**	**SE**	**C-ZS**	**C-FT**	***+Best-of-3***	**CWM**	***+Best-of-3***
MagicBrush	0.000	3.120	5.960	**6.710**	0.000	2.520	3.270	3.920	3.920
AG	0.000	1.200	1.610	3.080	0.170	2.480	2.740	**3.640**	**3.640**
Something	0.000	0.957	2.620	2.810	0.370	3.110	3.110	2.920	**3.310**
WhatsUp	0.000	0.000	**1.580**	0.755	0.146	0.880	0.980	0.540	0.540
Kubric	0.000	1.880	3.920	3.700	0.140	7.300	7.300	7.320	**7.780**
Average	0.000	1.430	3.140	3.410	0.165	3.260	3.480	3.670	**3.840**

Figure 4: Ablation study of synthetic trajectories (Synth.) and loss weighting (LW) in CWM. Numbers are GPT-4o-as-judge scores (↑, average of 3 runs). MB: MagicBrush, AG: Action-Genome, ST: Something-Something, WU: WhatsUp, KU: Kubric.

	CWM	*w/o Synth.*	*w/o LW*
MB	**3.48**	-0.28	-0.22
AG	**3.02**	-0.35	-0.08
ST	**3.06**	-0.18	-0.19
WU	**0.46**	0.40	0.08
KU	**7.14**	-0.03	-0.33
All	**3.43**	-0.09	-0.15

Figure 5: Human evaluation results. † indicates all results whose gap with respect to CWM is significant, based on a Wilcoxon signed-rank test ($p = 0.05$). MB: MagicBrush, AG: Action-Genome, ST: Something-Something, WU: WhatsUp, KU: Kubric.

	GoT	**SE**	**C-FT**	**CWM**
MB	0.06†	**0.29**†	-0.32†	-0.03
AG	-0.23†	-0.46†	0.32	**0.37**
ST	0.00	-0.37†	0.18	**0.20**
WU	**0.25**	-0.38†	0.14	0.00
KU	-0.52†	-0.22†	0.34	**0.40**
All	-0.09†	-0.23†	0.13	**0.19**

Human Evaluation. Following [18], we conduct a blind human evaluation comparing GoT, SmartEdit, C-FT, and our proposed CWM. We randomly sample 5 examples from each subset within AURORA-BENCH and present the outputs generated by each of the four models. Human annotators are asked to identify the best and worst generated observations based on three criteria: (1) *Realism*: the generated image should exhibit natural textures and lighting while remaining faithful to the input scene; (2) *Instruction-Following Ability*: the edit should clearly reflect the given instruction; and (3) *Over-Editing*: the modification should be minimal and focused, altering only what is necessary. Each model receives +1 point for being selected as the best, -1 for the worst, and 0 otherwise. We compute the average scores over 350 annotated samples, as reported in Table 5. The results align with automatic evaluations: image-editing models excel in the MagicBrush domain, but fall short on action-centric datasets such as Action-Genome, Something-Something, and Kubric. In contrast, CWM outperforms C-FT on all three of these datasets, highlighting its strength in next-observation prediction in real-world, action-centric trajectories.

Table 4: Detailed scores of GPT4o-as-a-judge evaluation for loss-weighting and standard training. We report the scores for **E**diting **S**uccess (**ES**) and **M**inimal **E**diting (**ME**). MB: MagicBrush, AG: Action-Genome, ST: Something-Something, WU: WhatsUp, KU: Kubric. We highlight the best and worst scores for each category.

	Weighted		**Standard**	
	ES (↑)	ME (↑)	ES (↑)	ME (↑)
MB	3.73	8.17	3.68	8.46
AG	3.18	8.03	2.37	8.13
ST	3.32	7.01	2.78	7.20
WU	0.54	7.25	0.76	7.19
KU	7.75	8.49	7.24	8.70
Avg. GPT4o	3.71	7.80	3.37	7.94
	3.67		3.58	

Ablation Study on Synthetic Trajectories. To assess the importance of extra supervision from CDM-synthetic trajectories, Table 4 reports GPT-4o's scores for this ablation. We see performance

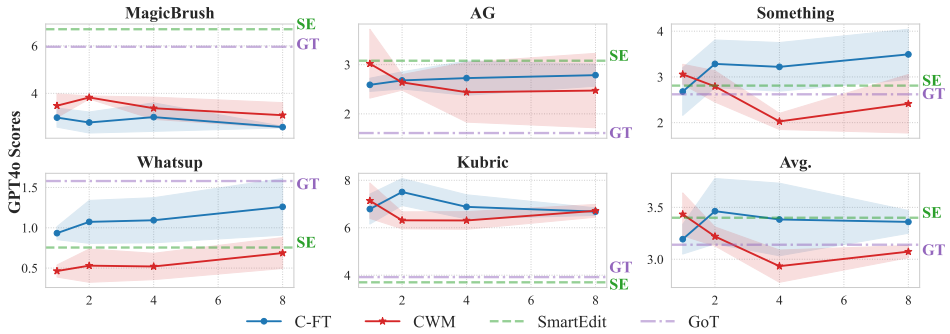


Figure 6: GPT-4o scores for test-time verification with K samples, where $K \in \{1, 2, 4, 8\}$. We use a blue line for C-FT and a red line for CWM, plotting the standard deviation as the shaded area. We indicate the scores for GoT (GT) and SmartEdit (SE) as horizontal lines.

drops on most datasets—particularly on Something and AG—when the additional training data from unlabelled videos is removed, highlighting the effectiveness of bootstrapping CWM with large-scale real-world data via CDM. An exception is the WhatsUp dataset, which focuses on specific actions within a fixed scene; in this case, training in an open-domain setting may not transfer effectively.

Ablation Study on Loss Weighting. Based on Table 4, we also observe consistent degradation when loss weighting is removed, demonstrating the benefit of explicitly incorporating the recognition model into visual next-token prediction. To better understand the effect of loss weighting, Table 4 reports the average scores for two criteria used in the GPT-4o-as-a-judge evaluation separately: Editing Success (ES), which measures how well the model captures the intended action and performs the corresponding edit, and Minimal Editing (ME), which assesses whether the model introduces unnecessary modifications. The full distribution of GPT-4o scores is provided in Appendix A.8. Our analysis reveals that the primary bottleneck for CWM remains its ability to reliably follow the instruction, as reflected by the fact that ES scores are significantly lower than ME scores. Loss weighting partly solves this problem, increasing the editing success and reducing copying behaviour, albeit at the cost of sometimes over-editing the source observation.

Verification at Test Time. We evaluate test-time verification using CDM in Figure 6, comparing C-FT and CWM with $K \in 1, 2, 4, 8$. Each experiment is repeated three times, and we visualise the mean and standard deviation. By increasing exploration on more candidate next observations, C-FT benefits from test-time verification on most datasets with real-world trajectories (e.g., AG, Something, WhatsUp), suggesting the effectiveness of CDM's trajectory preferences. Increasing K does not always improve performance (MagicBrush, Kubric), suggesting that bootstrapping with a dynamics model that shares the same foundation model backbone may be limiting. In contrast, CWM shows no clear gain, likely because it was trained with the synthetic trajectories and has already internalised CDM's preferences—as is evident from its strong $K = 1$ performance. In summary, CDM-based verification boosts C-FT's performance to the same level as CWM, by leveraging more diverse samples at inference time rather than during training.

	SMQA	ESB
C-ZS	26.1	15.0
CM-F	25.8	**21.2**
CWM	**27.2**	17.5

Table 5: Performance on SpatialMQA (SMQA) and EmbodiedSpatial-Bench (ESB).

Image Editing as an Auxiliary Task. Training on action-centric image editing task exposes the model to interactive supervision, where it receives action and predicts subsequent observations. It should encourages the model to ground concepts more effectively and to generalise beyond editing. Since AURORA includes a rich variety of spatial relations (e.g., left/right orientation), we further evaluated our trained models on two spatial reasoning benchmarks: SpatialMQA (SMQA) [41] and EmbodiedSpatial-Bench (ESB) [42]. Table 5 shows our trained models with the world-modelling objective achieves improvements over the baseline. These findings demonstrate that CWM transfers beyond image editing to broader spatial reasoning tasks, underscoring world modelling as a valuable training signal for strengthening model's other fundamental capabilities such as spatial understanding.

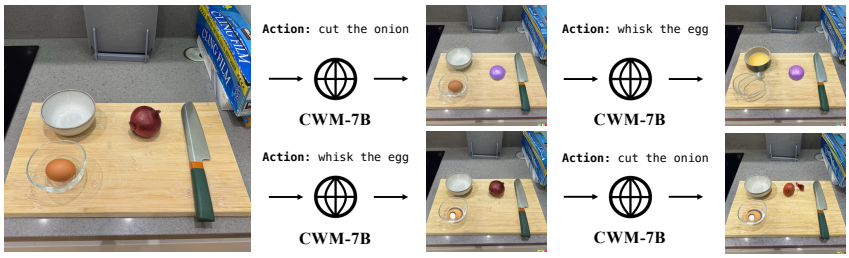


Figure 7: A qualitative case of real-world observation prediction, demonstrating CWM's ability to steer predictions using language and perform sequential predictions. More cases from AURORA-BENCH are in Appendix A.4.

Qualitative Example. Figure 7 presents a real-world example demonstrating that CWM's predicted observations can be guided through language expressing actions. CWM is also capable of iteratively generating future observations in multiple steps while maintaining consistency with previous frames.

5 Related Work

Despite the surge in interest for world modelling [1, 43, 44], previous works focused mostly on building specialised *ad-hoc* world models. These world models can be explicitly learnt as a visual simulator [2, 3, 4], or enable planning with model-based reinforcement learning [45, 46, 47, 48, 11]. Instead, we focus on leveraging large-scale multimodal foundation models [49, 50, 39] to develop world models, which is more attractive due to the inductive bias they provide from their extensive training. This is possible thanks to frameworks that integrate observations, actions, and rewards into a unified sequence of tokens in autoregressive Transformers [51], building on pioneering works such as Decision Transformers [52] and GATo [53]. Related to our work, [54] initialise the parameters of RL policies with VLMs, thus taking advantage of the abundant and general world knowledge encoded in their representations. 3D-VLA [55] introduces a set of interaction tokens into a Large Language Model to engage with the environment as an embodied agent. [56, 57] explore large-scale self-supervised learning via next token or frame prediction to build a unified model absorbing internet knowledge, learning from interaction via video.

AURORA-BENCH [18] was the first to approach world modelling through the lens of an action-centric image editing task. With advanced native VLMs capable of the interleaved generation [17, 58], we systematically investigate how this data may help us bootstrap a world model, implicitly stored in the VLMs, with an easier-to-train dynamics model. Most similar to our work, [19] train a dynamics model which aims to uncover the underlying action between video frames in unlabelled video frames from the Minecraft game. Through this model, they synthesise trajectories to train a policy for sequential decision making. In contrast with [19], we focus on next-observation prediction as a task to evaluate world modelling. First, this allows us to port the observation space to real-world frames, rather than simulated ones, hence assessing whether world models are well aligned with the physical environment. Second, this broadens the space of actions from a few choices to the combinatorially infinite and expressive space of language, capturing a significantly more diverse range of dynamics.

6 Conclusion

In this work, we explored whether we can develop word models from VLMs. By evaluating them on action-centric image editing AURORA-BENCH [18], we first show that these models lack a clear preference for ground-truth real-world trajectories. To address this, we induce a dynamics model from the same VLM to bootstrap a world model using automatic annotation of unlabelled real-world videos and inference-time verification. Experiments confirm the effectiveness of both strategies, with our general-purpose world model achieving state-of-the-art performance compared to existing approaches specialised for image editing.

References

[1] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[2] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical AI. *arXiv preprint arXiv:2501.03575*, 2025.

[3] Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

[4] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. 2024. *URL https://openai. com/research/video-generation-models-as-world-simulators*, 3:1, 2024.

[5] Yiran Qin, Zhelun Shi, Jiwen Yu, Xijun Wang, Enshen Zhou, Lijun Li, Zhenfei Yin, Xihui Liu, Lu Sheng, Jing Shao, et al. WorldSimBench: Towards video generation models as world simulators. *arXiv preprint arXiv:2410.18072*, 2024.

[6] Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, et al. Do as I can, not as I say: Grounding language in robotic affordances. In *Conference on robot learning*, pages 287–318. PMLR, 2023.

[7] Wenlong Huang, Fei Xia, Ted Xiao, Harris Chan, Jacky Liang, Pete Florence, Andy Zeng, Jonathan Tompson, Igor Mordatch, Yevgen Chebotar, et al. Inner Monologue: Embodied reasoning through planning with language models. In *6th Annual Conference on Robot Learning*, 2022.

[8] Chengzu Li, Wenshan Wu, Huanyu Zhang, Yan Xia, Shaoguang Mao, Li Dong, Ivan Vulić, and Furu Wei. Imagine while reasoning in space: Multimodal visualization-of-thought. *arXiv preprint arXiv:2501.07542*, 2025.

[9] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022.

[10] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 1(2):6, 2023.

[11] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse control tasks through world models. *Nature*, 640(8059):647–653, 2025.

[12] Hao Liu, Wilson Yan, Matei Zaharia, and Pieter Abbeel. World model on million-length video and language with blockwise RingAttention. *arXiv preprint arXiv:2402.08268*, 2024.

[13] Saman Motamed, Laura Culp, Kevin Swersky, Priyank Jaini, and Robert Geirhos. Do generative video models learn physical principles from watching videos? *arXiv preprint arXiv:2501.09038*, 2025.

[14] Jensen Gao, Bidipta Sarkar, Fei Xia, Ted Xiao, Jiajun Wu, Brian Ichter, Anirudha Majumdar, and Dorsa Sadigh. Physically grounded vision-language models for robotic manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12462–12469. IEEE, 2024.

[15] Yifu Qiu, Zheng Zhao, Yftah Ziser, Anna Korhonen, Edoardo Ponti, and Shay B Cohen. Are large language model temporally grounded? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7057–7076, 2024.

[16] Mostafa Abdou, Artur Kulmizev, Daniel Hershcovich, Stella Frank, Ellie Pavlick, and Anders Søgaard. Can language models encode perceptual structure without grounding? A case study in color. In Arianna Bisazza and Omri Abend, editors, *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 109–132, Online, November 2021. Association for Computational Linguistics.

[17] Chameleon Team. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*, 2024.

[18] Benno Krojer, Dheeraj Vattikonda, Luis Lara, Varun Jampani, Eva Portelance, Christopher Pal, and Siva Reddy. Learning Action and Reasoning-Centric Image Editing from Videos and Simulations. In *NeurIPS*, 2024. Spotlight Paper.

[19] Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video PreTraining (VPT): Learning to act by watching unlabeled online videos. *Advances in Neural Information Processing Systems*, 35:24639–24654, 2022.

[20] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–8, 2019.

[21] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[22] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019.

[23] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[24] Qwen Team. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

[25] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[26] Yuanhan Zhang, Bo Li, haotian Liu, Yong jae Lee, Liangke Gui, Di Fu, Jiashi Feng, Ziwei Liu, and Chunyuan Li. Llava-next: A strong zero-shot video understanding model, April 2024.

[27] Feng Li, Renrui Zhang, Hao Zhang, Yuanhan Zhang, Bo Li, Wei Li, Zejun Ma, and Chunyuan Li. Llava-next-interleave: Tackling multi-image, video, and 3d in large multimodal models. *arXiv preprint arXiv:2407.07895*, 2024.

[28] Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, et al. Qwen2. 5-omni technical report. *arXiv preprint arXiv:2503.20215*, 2025.

[29] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The EPIC-KITCHENS dataset. In *Proceedings of the European conference on computer vision (ECCV)*, pages 720–736, 2018.

[30] Niklas Muennighoff, Zitong Yang, Weijia Shi, Xiang Lisa Li, Li Fei-Fei, Hannaneh Hajishirzi, Luke Zettlemoyer, Percy Liang, Emmanuel Candès, and Tatsunori Hashimoto. s1: Simple test-time scaling. *arXiv preprint arXiv:2501.19393*, 2025.

[31] Charlie Snell, Jaehoon Lee, Kelvin Xu, and Aviral Kumar. Scaling LLM test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

[32] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3749–3761, 2022.

[33] Tim Brooks, Aleksander Holynski, and Alexei A Efros. InstructPix2Pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[34] Rongyao Fang, Chengqi Duan, Kun Wang, Linjiang Huang, Hao Li, Shilin Yan, Hao Tian, Xingyu Zeng, Rui Zhao, Jifeng Dai, et al. GoT: Unleashing reasoning capability of multimodal large language model for visual generation and editing. *arXiv preprint arXiv:2503.10639*, 2025.

[35] Yuzhou Huang, Liangbin Xie, Xintao Wang, Ziyang Yuan, Xiaodong Cun, Yixiao Ge, Jiantao Zhou, Chao Dong, Rui Huang, Ruimao Zhang, et al. SmartEdit: Exploring complex instruction-based image editing with multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8362–8371, 2024.

[36] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. BERTScore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020.

[37] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[38] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[39] Yecheng Wu, Zhuoyang Zhang, Junyu Chen, Haotian Tang, Dacheng Li, Yunhao Fang, Ligeng Zhu, Enze Xie, Hongxu Yin, Li Yi, et al. VILA-U: a unified foundation model integrating visual understanding and generation. *arXiv preprint arXiv:2409.04429*, 2024.

[40] Afra Amini, Tim Vieira, Elliott Ash, and Ryan Cotterell. Variational best-of-n alignment. *The Thirteenth International Conference on Learning Representations*, 2025.

[41] Jingping Liu, Ziyan Liu, Zhedong Cen, Yan Zhou, Yinan Zou, Weiyan Zhang, Haiyun Jiang, and Tong Ruan. Can multimodal large language models understand spatial relations? *arXiv preprint arXiv:2505.19015*, 2025.

[42] Mengfei Du, Binhao Wu, Zejun Li, Xuan-Jing Huang, and Zhongyu Wei. Embspatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 346–355, 2024.

[43] Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3:9–44, 1988.

[44] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019.

[45] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pages 2555–2565. PMLR, 2019.

[46] Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *Deep Reinforcement Learning Workshop NeurIPS 2022*, 2022.

[47] Jan Robine, Marc Höftmann, Tobias Uelwer, and Stefan Harmeling. Transformer-based world models are happy with 100k interactions. *arXiv preprint arXiv:2303.07109*, 2023.

[48] Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos J Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari. *Advances in Neural Information Processing Systems*, 37:58757–58791, 2024.

[49] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. Video-LLaVA: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5971–5984, 2024.

[50] Xiaokang Chen, Zhiyu Wu, Xingchao Liu, Zizheng Pan, Wen Liu, Zhenda Xie, Xingkai Yu, and Chong Ruan. Janus-Pro: Unified multimodal understanding and generation with data and model scaling. *arXiv preprint arXiv:2501.17811*, 2025.

[51] Jialong Wu, Shaofeng Yin, Ningya Feng, Xu He, Dong Li, Jianye Hao, and Mingsheng Long. iVideoGPT: Interactive VideoGPTs are scalable world models. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 68082–68119. Curran Associates, Inc., 2024.

[52] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

[53] Scott Reed, Konrad Zolna, Emilio Parisotto, Sergio Gomez Colmenarejo, Alexander Novikov, Gabriel Barth-Maron, Mai Gimenez, Yury Sulsky, Jackie Kay, Jost Tobias Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022.

[54] William Chen, Oier Mees, Aviral Kumar, and Sergey Levine. Vision-language models provide promptable representations for reinforcement learning. In *Automated Reinforcement Learning: Exploring Meta-Learning, AutoML, and LLMs*, 2024.

[55] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3D-VLA: A 3D Vision-Language-Action Generative World Model. In *International Conference on Machine Learning*, pages 61229–61245. PMLR, 2024.

[56] Sherry Yang, Jacob Walker, Jack Parker-Holder, Yilun Du, Jake Bruce, Andre Barreto, Pieter Abbeel, and Dale Schuurmans. Video as the new language for real-world decision making. *arXiv preprint arXiv:2402.17139*, 2024.

[57] Achint Soni, Sreyas Venkataraman, Abhranil Chandra, Sebastian Fischmeister, Percy Liang, Bo Dai, and Sherry Yang. VideoAgent: Self-improving video generation. *arXiv preprint arXiv:2410.10076*, 2024.

[58] Ethan Chern, Jiadi Su, Yan Ma, and Pengfei Liu. ANOLE: An open, autoregressive, native large multimodal models for interleaved image-text generation. *arXiv preprint arXiv:2407.06135*, 2024.

[59] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[60] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online, October 2020. Association for Computational Linguistics.

A Appendix

A.1 Limitations

While our approach demonstrates the effectiveness of our approaches across AURORA-BENCH, the authors would like to highlight few limitations we have discovered:

- Despite efforts to guide the model via supervised fine-tuning with loss weighting or inference-time verification (Table 3), we observe that the model may still resort to copying the source observation, especially under low sampling temperatures or ambiguous instructions.
- While we show preliminary results of language-steered observation prediction in Figure 7, fine-grained control remains limited, and understanding subtle instructions (e.g., spatial or quantitative edits) remains challenging.
- We observe variance across different runs of experiments, likely due to the sensitivity of sampling for generation in multimodal models. To address this, we report results averaged over multiple runs and include performance under the best-of-N sampling distribution during inference for a robust comparison.
- We mostly conduct experiments using the native and unified VLM, Chameleon, as it is currently the only open-source VLM that supports interleaved image-text generation by default. This choice allows for fair and consistent benchmarking across our tasks. Moreover, Chameleon has demonstrated competitive performance in our settings. For example, its results are comparable to VILA-U in our dynamics prediction task. Future work should explore the generalisation to other multimodal foundation models with stronger capabilities.

A.2 Broader Impact

This work develops models for action-centric image editing for visual world modelling. While our primary aim is to advance fundamental research in world modelling, we acknowledge potential risks, particularly in the generation of realistic future observations.

A core concern is the potential misuse of the models for creating deceptive visual content, including fabricated action sequences or manipulated images that imply false causality. Although the model is not explicitly designed for these tasks, its ability to generate coherent visual predictions from the linguistic action could be adapted for such uses if deployed irresponsibly.

Even in intended use, risks include over-reliance on generated outputs in downstream tasks such as robotic control, or interactive systems. Model failures—e.g., copying artefacts, hallucinations, or broken object continuity—can lead to incorrect inferences or reinforce dataset biases.

To mitigate potential misuse, we limit our model release to research purposes under a non-commercial license and clearly communicate its capabilities and limitations. We urge caution when adapting them for deployment, particularly in settings with high societal or ethical sensitivity.

A.3 Model Performance on AURORA-BENCH with 5 Metrics

In addition to GPT4o-as-a-judge evaluation, we further employ a diverse set of automatic metrics covering both low-level and semantic fidelity: 1) we compute the **L1 distance** between the predicted and target observation as a pixel-level metric. 2) We extract visual features and compute the cosine similarity in their respective embedding spaces for several image encoders, including (**CLIP-I** and **DINO**), to assess semantic similarity. Additionally, to measure alignment between image content and the action semantics, we compute **CLIP-T**, the similarity between the edited image and its BLIP-generated caption. These metrics are evaluated in addition to GPT4o-as-a-judge metric following previous works in image editing [35, 34, 18]. We report the detailed results with 5 metrics in Table 6. We notice that copy baseline exhibits the best performance as measured by the distance-based and visual encoder-based approach, as indicated in Table 3. This poses a challenge to the reliability of the traditional metrics in fairly evaluating the action-centric image editing task. On the other hand, GPT4o-as-a-judge metric robustly assigns 0 score to Copy, indicating its robustness in detecting copying generation while putting GPT-as-a-judge as the most reliable metric to interpret.

Table 6: Model performance at MagicBrush, Action-Genome, Something, WhatsUp and Kubric on AURORA-BENCH. For C-FT and CWM We report both the model performance and their performance in the *best-of-N* distribution. We report the average GPT4o scores for each model at the bottom. We highlight the better GPT-4o scores for C-FT and CWM. We **bold** the best performance among all models, except Copy and *best-of-N* performances. SE: SmartEdit.

Datasets	Metrics	Models								
		Copy	**PixInstruct**	**GoT**	**SE**	**CM**	**C-FT**	***+Best-of-3***	**CWM**	***+Best-of-3***
MagicBrush	L1	0.027	0.114	**0.063**	0.068	0.287	0.075	0.075	0.090	0.078
	CLIP-I	0.959	0.877	0.930	**0.937**	0.671	0.913	0.914	0.906	0.909
	CLIP-T	0.289	0.275	0.286	0.290	0.227	0.289	0.289	**0.291**	0.291
	DINO	0.931	0.761	0.881	**0.894**	0.292	0.883	0.883	0.864	0.864
	GPT-4o	0.000	3.120	5.960	**6.710**	0.000	2.520	3.270	3.920	3.920
AG	L1	0.069	0.220	0.174	**0.137**	0.314	0.170	0.168	0.168	0.167
	CLIP-I	0.943	0.757	0.846	0.811	0.609	0.872	0.872	**0.881**	0.883
	CLIP-T	0.279	0.254	0.280	0.268	0.214	0.280	0.284	**0.284**	0.284
	DINO	0.929	0.557	0.785	0.774	0.258	0.801	0.817	**0.816**	0.816
	GPT-4o	0.000	1.200	1.610	3.080	0.170	2.480	2.740	**3.640**	3.640
Something	L1	0.135	0.232	0.184	**0.163**	0.293	0.184	0.184	0.196	0.184
	CLIP-I	0.870	0.709	0.807	0.773	0.649	**0.820**	0.820	0.804	0.804
	CLIP-T	0.275	0.238	0.269	0.265	0.232	**0.271**	0.269	0.268	0.268
	DINO	0.797	0.453	0.636	0.662	0.297	**0.675**	0.653	0.666	0.666
	GPT-4o	0.000	0.957	2.620	2.810	0.370	**3.110**	3.110	2.920	3.310
WhatsUp	L1	0.039	0.138	0.078	0.067	0.251	**0.066**	0.066	0.070	0.070
	CLIP-I	0.954	0.817	**0.923**	0.888	0.721	0.877	0.880	0.870	0.883
	CLIP-T	0.326	0.287	**0.316**	0.312	0.243	0.309	0.310	0.306	0.307
	DINO	0.908	0.615	**0.850**	0.805	0.424	0.836	0.841	0.831	0.838
	GPT-4o	0.000	0.000	**1.580**	0.755	0.146	0.880	0.980	0.540	0.540
Kubric	L1	0.011	0.104	**0.026**	0.064	0.276	0.044	0.044	0.044	0.044
	CLIP-I	0.963	0.796	0.895	0.868	0.660	0.897	0.899	**0.897**	0.898
	CLIP-T	0.282	0.259	0.281	0.271	0.213	**0.287**	0.287	0.287	0.288
	DINO	0.955	0.676	0.857	0.798	0.161	**0.906**	0.906	0.902	0.902
	GPT-4o	0.000	1.880	3.920	3.700	0.140	7.300	7.300	**7.320**	7.780
All	GPT-4o	0.000	1.430	3.140	3.410	0.165	3.260	3.480	**3.670**	3.840

A.4 Qualitative Cases

In this section, we present additional qualitative examples from AURORA-BENCH in Figure 8. We observe several common failure modes in image editing models. First, they sometimes fail to preserve the scene from the source observation (e.g., PixInstruct on Action-Genome and MagicBrush). Second, some models generate near-identical copies of the source as the target (e.g., GoT on Something-Something). Third, producing realistic outputs remains difficult, as seen in GoT's result on Kubric. Finally, maintaining object consistency is also a challenge—SmartEdit alters the object in WhatsUp, and CWM does so in Something-Something.

Despite the challenges, we also observe several positive editing behaviours from CWM. On Action-Genome, CWM correctly predicts spatial changes, such as *opening and closing a drawer*, which requires a strong understanding of the spatial concepts. In Something-Something, it is the only model to accurately capture the spatial concept of "falling down." On Kubric, it demonstrates basic counting ability by correctly adding one keyboard. In WhatsUp, CWM correctly grounds the action to the laptop, while other models mistakenly edit the monitor.

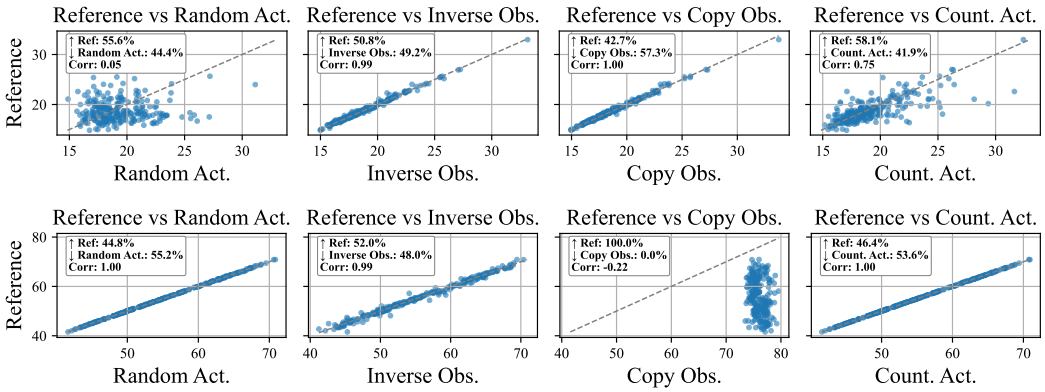


Figure 9: Comparison of predicted negative log-likelihoods (lower values indicate stronger model preference) for ground-truth real-world trajectories versus four types of negative trajectories. **Top**: Action prediction task for the dynamics model (observation $\times$ observation $\rightarrow$ action). **Bottom**: Next observation prediction task for the world model (observation $\times$ action $\rightarrow$ observation). The legend shows the percentage of times the model prefers the ground-truth trajectory ($\uparrow$) over the negatives ($\downarrow$).

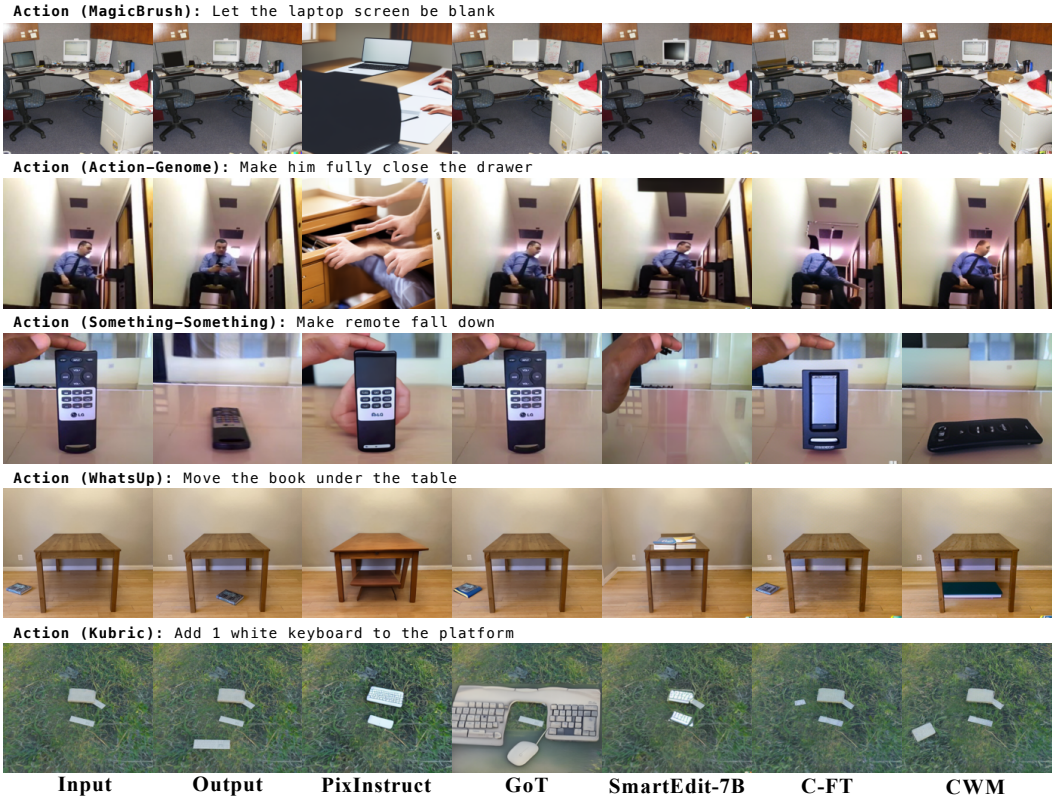


Figure 8: Qualitative examples of the predicted next observation from the state-of-the-art specialised image editing models, and our models including C-FT and CWM, on AURORA-BENCH.

A.5 Detailed Discussion for Chameleon's Predicted Likelihoods

From Figure 9, it emerges that Chameleon-7B displays a very limited preference for the ground-truth trajectories in a zero-shot setting. In the action prediction task (top panel), there is a slightly higher tendency to favour the ground-truth; however, even in the best case (counterfactual action), the model prefers the reference in only 58.1% of the samples. The high correlation in likelihoods indicates that

Table 7: Dataset statistics for the video and triplets from the trajectories annotated by CDM. **OPV**: observations (i.e., extracted key-frames) per video, **APV**: actions per video, **WPA**: words per action.

Dataset	Video		Triplet			
	Avg. Length	**Total Length**	**#Samples**	**#Avg. OPV**	**#Avg. APV**	**#Avg. WPA**
MIT	3.04 seconds	2.57 hours	19,658	2.05	1.05	7.10
UCF-101	7.24 seconds	26 hours	10,965	3.00	2.00	8.96
Kinetics700	9.02 seconds	18 hours	26,959	2.71	1.71	7.39

the VLM struggles also on visual manipulations. In the next-observation prediction task (bottom panel), the VLM mostly fails in effectively differentiating the ground truth from the negatives. An exception to this is the copy manipulation, where the model can always tell them apart. Although the underlying reason remains uncertain, one plausible explanation for this behaviour is that the model's ability to solve next-observation prediction tasks depends on their alignment with training sequences: for instance, it is plausible that Chameleon's data rarely features two identical adjacent images. In summary, Chameleon-7B does not exhibit a preference for ground-truth trajectories over negative ones, constructed through action- or observation-based manipulations.

A.6 Details of Processing CDM Annotations for Unlabelled Videos

Algorithm 1 Stratified Top-K Sampling with Action Class Uniformity

Require: Trajectory triplet set $X = \{(o_s^i, o_t^i, a^i, s^i, c^i)\}_{i=1}^N$, where s_i is the predicted likelihood of a^i, $c_i \in \mathcal{C}$ is the class, number of samples K

Sort X descending by score s_i
Initialize $S \leftarrow \emptyset$, and class_counts$[c] \leftarrow 0$ for all $c \in \mathcal{C}$
while $|S| < K$ **do**
  for all class $c \in \mathcal{C}$ in round-robin order **do**
    $X_c \leftarrow$ top unsampled item from class c in X
    if $X_c \neq \emptyset$ **then**
      $S \leftarrow S \cup \{X_c\}$
      Remove X_c from X
      class_counts$[c] \leftarrow$ class_counts$[c] + 1$
    end if
    if $|S| = K$ **then**
      break
    end if
  end for
end while
return S

We present the raw dataset statistics before sampling for Movements-in-Time, UCF-101 and Kinetics700 in Table 7. Figure 10 shows the distribution of CDM's predicted scores across action classes in Movements-in-Time, Kinetics700, and UCF-101. The predicted likelihoods are nearly uniform within each class, indicating that our sampling method maintains both class diversity and high overall likelihoods. The sampling procedure for CDM-annotated trajectories is detailed in Algorithm 1.

A.7 Prompt template for using GPT4o-as-a-Judge evaluation.

We provide the prompts used for evaluating image editing performance with GPT-4o in Figure A.7. We use `GPT-4o-2024-11-20`. The final score is the average of the minimum value of the two scores for each sample, as in [34].

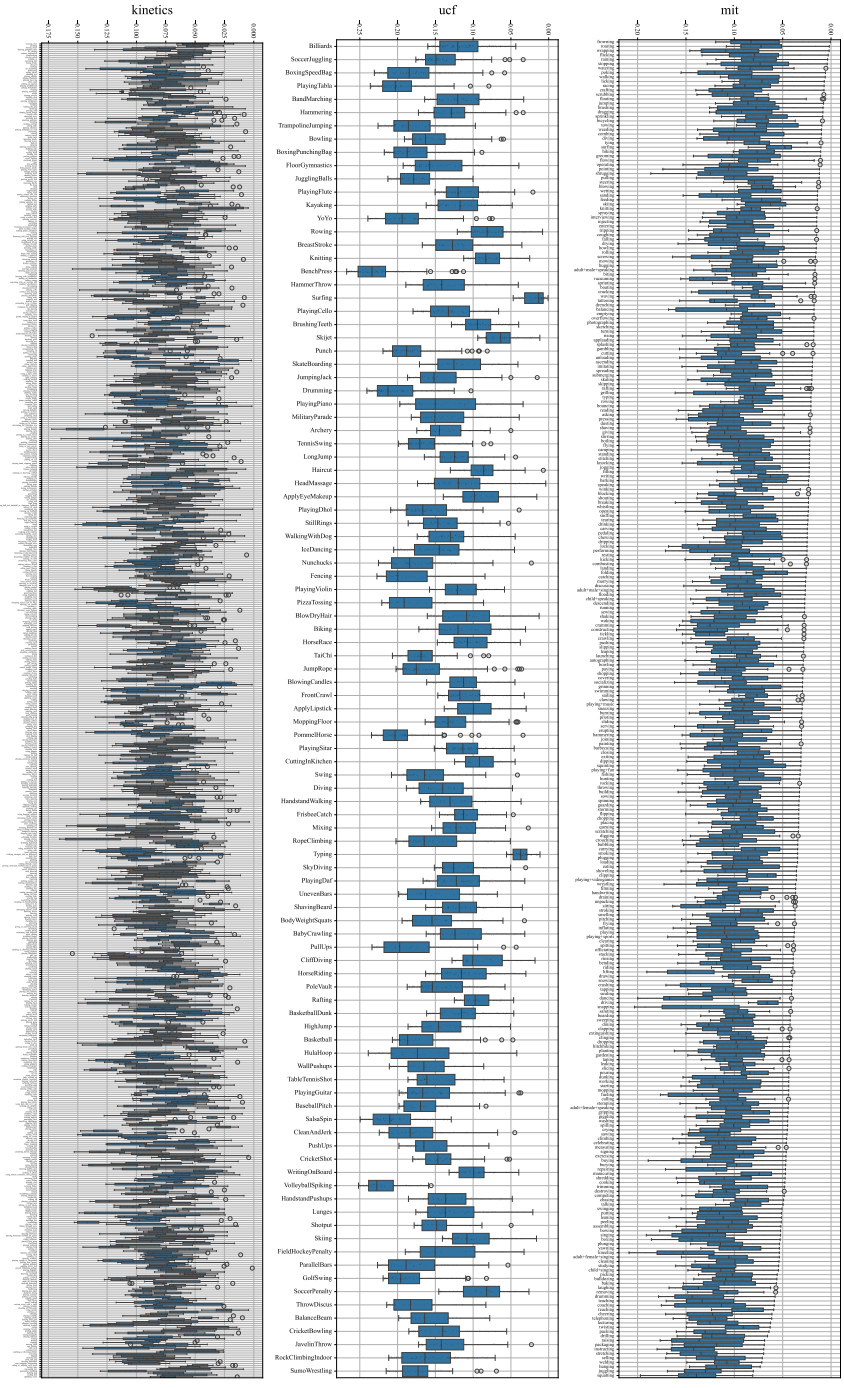


Figure 10: Distributions of triplet log-likelihoods predicted by CDM on Movements-in-Time, UCF-101, and Kinetics-700, based on 7K synthetic triplets per dataset. Triplets are uniformly sampled from each action class while maximising overall predicted likelihoods.

Prompt Template for GPT4o-as-a-judge Evaluation

You are a professional digital artist. You will have to evaluate the effectiveness of the AI-generated image(s) based on the given rules.

You will have to give your output in a valid way of a Python dictionary format (Keep your reasoning concise and short.):

```
{{"score": [...], "reasoning": "..." }}
```

and don't output anything else. Two images will be provided:

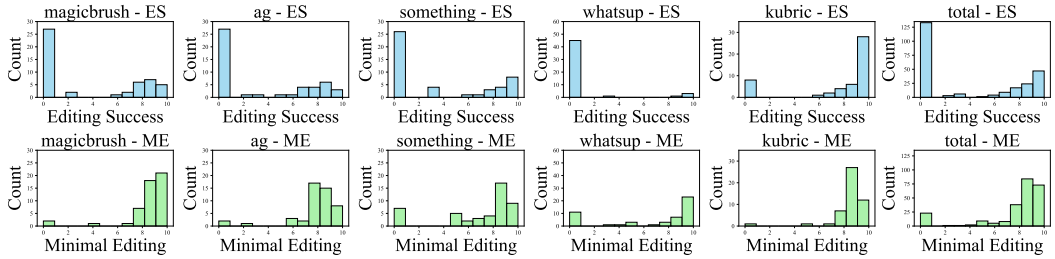
- The first being the original AI-generated image
- The second being an edited version of the first.

The objective is to evaluate how successfully the editing instruction has been executed in the second image. Note that sometimes the two images might look identical due to a failure in image editing. From a scale of 0 to 10:

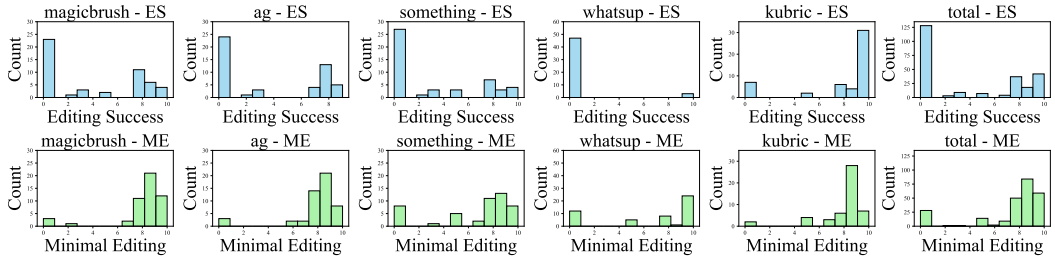
- A score from 0 to 10 will be given based on the success of the editing.
  - 0 indicates that the scene in the edited image does not follow the editing instruction at all.
  - 10 indicates that the scene in the edited image follows the editing instruction text perfectly.
  - If the object in the instruction is not present in the original image at all, the score will be 0.
- A second score from 0 to 10 will rate the degree of minimal editing in the second image.
  - 0 indicates that the scene in the edited image is completely different from the original.
  - 10 indicates that the edited image can be recognised as a minimally edited yet effective version of the original.

Put the score in a list such that: output score = `[score1, score2]`, where `score1` evaluates the editing success and `score2` evaluates the degree of the minimal editing.
Editing instruction: {instruction}

A.8 Detailed GPT4o Scores for Editing Success and Minimal Editing



(a) Detailed GPT4o scores for CWM trained with the standard loss.



(b) Detailed GPT4o scores for CWM trained with the L_2-weighted loss.

Figure 11: GPT4o scores' distributions of editing success (ES) and minimal editing (OE) for CWM trained with standard loss or our loss-weighting method.

Figure 11 shows the distribution of editing success (ES) and minimal editing (ME) scores for standard training and loss-weighted training. Loss weighting tends to improve editing success, with a modest trade-off in minimal editing quality in most of the datasets.

A.9 Implementation Details

A.9.1 Chameleon Dynamics Model

We fine-tune the Chameleon-7B checkpoint from the Anole-7B version [58] to predict the action given a pair of observations, framed as an action-prediction task. The model is trained on a merged dataset from Action-Genome, Kubric, MagicBrush, Something-Something from AURORA's annotated trajectories, and 15K EPIC-Kitchens processed by us. We downsample Kubric's trajectories to 10K. Training is performed for 10 epochs with a batch size of 64, using a learning rate of 2e-4 and cosine scheduling (500 warm-up steps). We use bfloat16 mixed-precision training and apply LoRA [59] for parameter-efficient fine-tuning (rank 16, $\alpha = 32$, dropout 0.05). Only the completion loss is used to optimise the generation of action. Training is conducted on 4 NVIDIA-H100-80GB-HBM3 GPUs using DeepSpeed for distributed optimisation.

A.9.2 C-FT Baseline

We fine-tune the Chameleon-7B checkpoint from the Anole-7B version [58]. The model is trained on a combined dataset from Action-Genome, Kubric, MagicBrush, and Something-Something, formatted as the image editing task. We downsample Kubric's trajectories to 10K. Training is conducted for 40 epochs with a batch size of 96 using the AdamW optimiser and a cosine learning rate scheduler (learning rate of 5e-4, 400 warm-up steps). We use mixed-precision training with bfloat16 and apply LoRA [59] for efficient fine-tuning (rank 16, $\alpha = 32$, dropout 0.05). We only train the model with the truncated loss from the completion part. We use 4 NVIDIA-H100-80GB-HBM3 GPUs with DeepSpeed for distributed training. During inference, we apply a logits processor to mask out non-image tokens, set the temperature to 1, and use top-1 sampling. We observe that temperature is critical in controlling model behaviour: lower values often cause the model to copy the source observation instead of generating meaningful edits.

A.9.3 Chameleon World Model

We fine-tune the Chameleon-7B checkpoint from the Anole-7B version [58]. The model is trained on a combined dataset from Action-Genome, Kubric, MagicBrush, Something-Something from AURORA's annotated trajectories, together with 7K trajectories from Movements-in-Time, 7K trajectories from UCF-101 and 7K trajectories from Kinetics700, formatted as the image editing task. Again, we downsample Kubric's trajectories to 10K. Training is conducted for 40 epochs with a batch size of 96 using the AdamW optimiser and a cosine learning rate scheduler (learning rate of 5e-4, 400 warm-up steps). We use mixed-precision training with bfloat16 and apply LoRA [59] for efficient fine-tuning (rank 16, $\alpha = 32$, dropout 0.05). We only train the model with the truncated loss from the completion part, but we weight the image tokens using L_2 strategy as introduced in Section 3. We use 4 NVIDIA-H100-80GB-HBM3 GPUs with DeepSpeed for distributed training. We use the same hyperparameters as C-FT during the inference time.

A.9.4 Computing Resources

All training experiments were conducted on a compute node equipped with 4× NVIDIA H100 80GB GPUs, 256 CPU cores, and 256GB of memory. The total GPU hours required for training C-FT, CWM, and CDM were approximately 200, 400, and 100 hours, respectively.

For inference, we used a single NVIDIA A100 80GB GPU with 8 CPU cores and 128GB memory. Inference for C-FT and CWM takes approximately 1 GPU hour per model. When applying verification with $K = 8$, inference time increases to around 8 GPU hours. CDM only takes around 0.3 GPU hours for inference.

A.9.5 Assets and Licenses

In this section, we list the public assets we used in this paper and the corresponding links.

Datasets. We include the detailed license and URL for the datasets we used in this paper.

- AURORA and AURORA-BENCH [18]: MIT license, the reader can find the corresponding version we use in this paper in `https://github.com/McGill-NLP/AURORA`.
- Movements-in-Time [20]: BSD-2-Clause license and its own License for Non-Commercial Use, the reader can find the corresponding version we use in this paper in `http://moments.csail.mit.edu/`.
- UCF-101 [23]: unknown license, the reader can find the corresponding version we use in this paper in `https://huggingface.co/datasets/flwrlabs/ucf101`.
- Kinetics700 [21, 22]: Creative Commons Attribution 4.0 International License, the reader can find the corresponding version we use in this paper in `https://research.google/pubs/the-kinetics-human-action-video-dataset/`.
- EPIC-Kitchens [29]: Creative Commons Attribution-NonCommercial 4.0 International License, the reader can find the corresponding version we use in this paper in `https://epic-kitchens.github.io/`.

Implementation. We use the other following code for the implementations:

- Transformers [60]: Apache-2.0 license. We use the 4.47.0 version, following the link at `https://github.com/huggingface/transformers`.
- DeepSpeed: We use the 0.14.4 version, following the link at `https://github.com/deepspeedai/DeepSpeed`.

Model. We use the following models or checkpoints for the implementations:

- Chameleon [17]: Chameleon Research License, the reader can find the corresponding version we use in this paper in `https://github.com/facebookresearch/chameleon`.
- Anole-7B [58]: Chameleon Research License and MIT License, the reader can find the corresponding version we use in this paper in `https://github.com/GAIR-NLP/anole`.

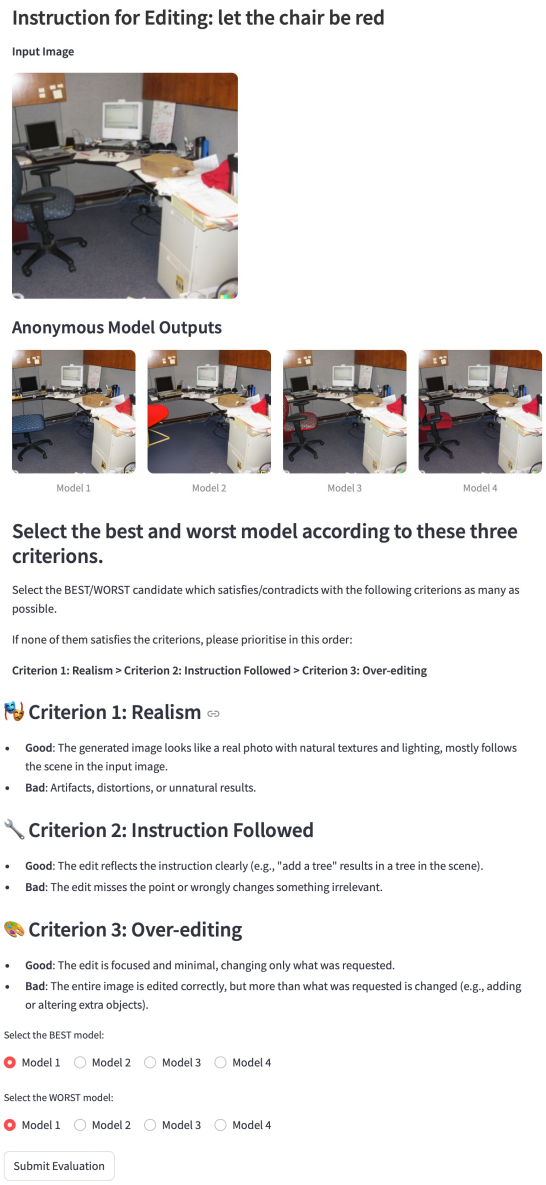


Figure 12: The screenshot for the instructions given to participants and the interface developed for conducting the evaluation.

- VILA-U [58]: MIT License, the reader can find the corresponding version we use in this paper in `https://github.com/mit-han-lab/vila-u`.
- SmartEdit [35]: Apache-2.0, the reader can find the corresponding version we use in this paper in `https://huggingface.co/TencentARC/SmartEdit-7B`.
- GoT [34]: MIT License, the reader can find the corresponding version we use in this paper in `https://github.com/rongyaofang/GoT`.
- PixInstruct [33]: PixInstruct customised license, the reader can find the corresponding version we use in this paper in `https://github.com/timothybrooks/instruct-pix2pix`.

A.10 Details of Human Evaluation

We conducted a human evaluation using a custom-built interface, with the full interface and instructions shown in Figure 12. A total of 14 participants were recruited, all of whom are PhD-level graduate students or higher. Participation was voluntary. Each participant was asked to evaluate 25 samples, which typically required 15–20 minutes to complete.

The evaluation process, including recruitment, instructions, and data processing and storage, followed our institution's ethical guidelines for human subject research. All participants were informed of the purpose of the study and provided consent. No personally identifiable information was collected, and all data were stored and analysed in accordance with privacy standards.

A.11 Safeguards

CWM performs observation prediction through image generation and, while its outputs are task-specific, we acknowledge that any generative model may carry potential for misuse. To mitigate these risks, we commit to the following safeguards upon release:

The model will be released solely for research purposes under a license that prohibits commercial use or any other harmful applications. The GitHub repository will include clear usage guidelines and terms of use, aligned with responsible AI principles.

We will include a disclaimer that the model is intended only for academic research in controlled environments. The datasets used for training are publicly available, action-centric image editing benchmarks that do not include sensitive or personally identifiable content.

Given the targeted nature of our model and the safeguards in place, we believe the risk of misuse is limited. Nonetheless, we encourage responsible use and welcome feedback from the community regarding potential improvements to safety.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS paper checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction include claims made in this paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Yes, we provide the description of limitations of our paper in Appendix A.1.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification: This paper does not include theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: See Appendix A.9 for the implementation details to reproduce the results in this paper. We also provide the detailed creation of training data used in this paper in Section 3.2, and experiment settings in Section 4 for ensuring the reproducibility. To maximise the reproducibility, we will also release the code for reproducing CWM, together with all used data resources we have curated in this paper to the supplementary material and the public.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: We will release the code for reproducing CWM, together with all used data resources we have curated in this paper to the supplementary material and the public.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: See Appendix A.9 for the implementation details to reproduce the results in this paper. We also provide the detailed creation of training data used in this paper in Section 3.2, and experiment settings in Section 4 for ensuring the reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We suitably provide the significant test for the applicable experiments such as human evaluation in Table 5, as well as the standard deviation of results in Figure 6 for the test-time verification.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

Justification: We provide the description of compute resources needed to conduct the experiments in Appendix A.9.4.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We will make sure to follow the NeurIPS code of ethics and the policy that preserves anonymity.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Yes, we provide the description of broader impact of our paper in Appendix A.2.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [Yes]

Justification: Check the safeguards statement in the Appendix A.11.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We provide the description of licenses for the used assets in Appendix A.9.5.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: We will properly provide the documentation for the releasing code and our trained models used in this paper, together with the necessary license.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.

- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: See Appendix A.10 for the details of human evaluation, and Figure 12 for a screenshot of the platform developed for the evaluation.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [Yes]

    Justification: The human evaluation conducted in this project has been reviewed and approved by the ethical panel of `[anonymised]` with the case ID: 912488.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.