Extended Abstract

Motivation Today's vision-language models (VLMs) reason and answer primarily in text tokens, even for questions about motion, forces, depth, and spatial relations – e.g., "is the red cup or blue ball further from the camera?" (depth) or "how did the blue object move?" (optical flow). Reasoning about the physical world purely in language is ungrounded and a known source of geometric and dynamic hallucinations. Crucially, *which* visual information helps is task-dependent, which raises our central question: **what** visual abstractions should a reasoner invoke, and **when**?

Method We build an 8B-parameter generative VLM that integrates *visual abstractions as streams of thought* – camera pose, depth, optical flow, and text – in its reasoning trace, rather than text alone. We cast abstraction selection as a learned policy and train it with reinforcement learning over a verifiable reward: the model's own probability of the correct answer. Our system is self-improving; a single network serves as both the policy selecting the abstractions and the VQA-capable VLM utilizing them (and the same network is moreover an action-conditioned world model, a capability we reserve for counterfactual planning in future work).

Implementation The policy is initialized from the PSI visual world model and shares one token vocabulary across all modalities. The abstraction values (depth, optical flow, camera pose) are tokenized into a shared vocabulary; the model selects *which* to condition on and answers from them. We build an offline dataset of $\sim$220k video-questions, each storing 64 candidate notations with precomputed VQA losses, and optimize with advantage-weighted behavior cloning: among the lowest-loss traces that beat an RGB-only baseline, we softmax-sample a target by loss (with a small length penalty) and behavior clone it.

Results These visual abstractions improve VLM visually-grounded reasoning: most sampled notations beat RGB-only, and on EgoDex the 90th-percentile notation cuts loss by 13% – which in turn confirms exploitable signal for our RL. Building on this, across four held-out QA test sets (OpenVid, SSv2, EgoDex, PE) the learned policy lowers VQA loss over the RGB-only baseline on every dataset, and the best stream-of-thought is *input-dependent* – no fixed pipeline matches a learned, per-input policy.

Discussion Because one network is both the policy and the VQA-answerer – and also a world model that could later be used for counterfactual planning – improvements share weights and compound. This is what makes the loop *self-improving* rather than mere tool-selection, and it targets a crucial weakness of current VLMs – physical understanding and hallucination. Two limitations of our current projects are that the learned policy currently produces notations with limited diversity, and our training pipeline uses ground-truth VQA answers to compute reward. Future work includes exploring more diverse sampling strategies and self-generated reward mechanisms.

Conclusion Our offline RL is an *entry* into a cycle of recursive self-improvement. Crucially, this cycle bootstraps on *visual* abstractions – grounded, low-dimensional, disentangled properties of the physical world (depth, optical flow, camera pose) – rather than text, making it a *principled* mechanism for self-improvement: unlike existing text-based chain-of-thought methods, the rationales are tied to the geometry and dynamics that physical understanding actually depends on, precisely where text-only reasoning is ungrounded and hallucinates. For future work, we are excited to close the online RL loop, replace supervised reward with self-generated rewards, and use our model's world modeling capability to solve tasks that require counterfactual planning. These point toward VLMs that improve their own physical reasoning and tackle complex multi-step tasks (e.g., assembling an IKEA table).

Self-improving Vision-Language Models: Reinforcement Learning over Visual Abstractions

Khai Loong Aw*
Department of Computer Science
Stanford University
khaiaw@stanford.edu

Baihan Zhang*
Department of Computer Science
Stanford University
baihan@stanford.edu

Abstract

Modern vision-language models (VLMs) reason in text tokens even for questions about motion, depth, and spatial relations, leaving their physical reasoning ungrounded and prone to hallucination. We build an 8B-parameter generative VLM that instead emits and integrates *visual abstractions as streams of thought* – camera pose, depth, optical flow, and text – for reasoning. Crucially, which abstraction helps is video-, timepoint-, and task-dependent (depth for "is the red cup or blue ball further from the camera?", optical flow for "what action is the man doing?"), so we use reinforcement learning to learn *what* to invoke and *when*. We first show that visual abstractions broadly improve VLM visually-grounded reasoning – most sampled notations beat an RGB-only baseline, and the most helpful one is input-dependent – motivating a learned policy and scoping its design space. We then train the VLM as a policy with RL over verifiable rewards (RLVR) on Video Question Answering (VQA) over ∼220k video-questions, each annotated with 64 candidate notations and their VQA losses; the learned policy beats RGB-only and uniform policies across four held-out test sets. By learning to integrate geometric inductive biases on demand, we bootstrap an entry into a cycle of recursive self-improvement.

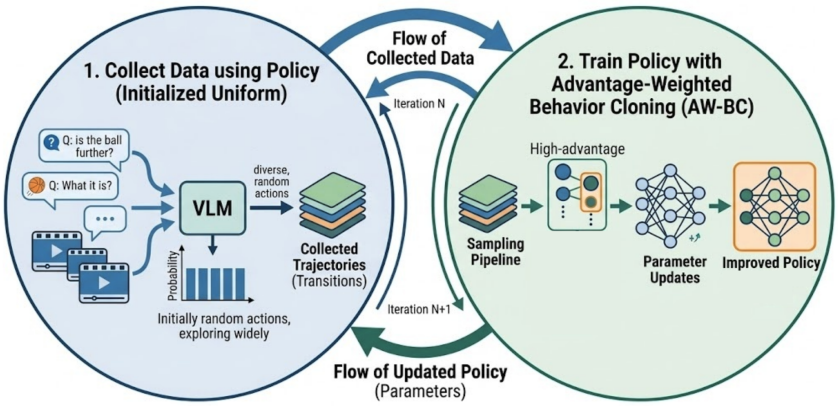

Figure 1: The self-improvement loop. The same network is both the *policy* that selects which visual abstractions to invoke and the *answerer* that conditions on them, so the reward is the model's own answer likelihood – self-supplied, no separate critic.

*Equal contribution

1 Introduction

Visual understanding is as fundamental as language for reasoning about the physical world; when people predict how objects will move, they simulate visually rather than verbally (Shepard and Metzler, 1971; Battaglia et al., 2013). Yet modern vision-language models (VLMs) reason and answer primarily by producing language tokens, even for questions about motion, forces, depth, and spatial relations (Liu et al., 2024; Alayrac et al., 2022). Reasoning about geometry and dynamics purely in text is ungrounded, and is a documented source of hallucination that object-level mitigation methods do not address (Li et al., 2023; Guan et al., 2023).

Crucially, which visual information helps is *video-, timepoint-, and task-dependent*. "Is the red cup or blue ball further from the camera?" calls for depth; "what action is the person performing?" calls for optical flow. This raises the central question of this work: **what** visual abstractions should a reasoner invoke, and **when**? Hand-designing a fixed pipeline that always invokes the same abstractions in the same order cannot answer this, because the right abstraction changes from input to input.

We therefore cast abstraction selection as a *learned policy*. We build an 8B-parameter generative VLM that emits its own visual stream of thought – selecting which abstractions (depth, flow, camera pose) to invoke for a given task – and train this policy with reinforcement learning over a verifiable reward: the model's probability of the correct answer. A key property of our setup is that a single network is both the policy that selects abstractions and the VLM that answers from them, so the reward is the model evaluating itself, with no separate reward model or human labels at scoring time. The model's own capability to *generate* these abstractions is reserved for the future closed-loop setting, and its action-conditioned world-modeling for later counterfactual planning.

Our contributions are: (1) *evidence that visual abstractions improve VLM visually-grounded reasoning* – across our QA sets most sampled notations beat an RGB-only baseline (on EgoDex the 90th-percentile notation cuts answer loss by 13%) – with the secondary, project-specific benefit of scoping the RL design space and showing the optimal abstraction is input-dependent; (2) an *offline RL policy*, trained by advantage-weighted behavior cloning over $\sim$220k video-questions, that lowers VQA loss over RGB-only across four held-out QA sets; and (3) a concrete framework for *self-improving VLMs* – one network serving as both policy and answerer gives a self-supplied, verifiable reward over visual-abstraction rationales – bringing to vision-language models the self-improvement previously shown for LLMs (e.g., STaR (Zelikman et al., 2022)), but grounded in visual abstractions that target physical understanding and hallucination.

Our work demonstrates a single, *offline* round of RL and evaluates on held-out QA from the training distributions; online RL and external spatial benchmarks (our original proposal) are framed as future work. We view this single round as an *entry* into a recursive self-improvement cycle rather than a demonstration of the full cycle.

2 Related Work

Visual world models. We build on Probabilistic Structure Integration (PSI) (Kotar et al., 2025), a token-based autoregressive model that jointly predicts RGB frames, camera pose, depth, and optical flow from video by treating each spatial patch as an independent token and learning distributions over their joint ordering. PSI shows a single model can serve as a richly promptable world model across visual modalities, but it has no text and no learned policy for selecting *which* modality to predict next. We extend PSI with text (and point-track) tokens so language can interleave with visual abstractions, and add a learned modality-selection policy trained against task reward.

Visual chain-of-thought. A growing body of work augments VLM reasoning with intermediate visual steps: Visual Sketchpad (Hu et al., 2024) draws annotations onto the image, V* (Wu and Xie, 2023) iteratively crops regions of interest, CogCoM (Qi et al., 2025) chains perceptual manipulations, Whiteboard-of-Thought (Menon et al., 2024) renders diagrams via code, and Mirage (Yang et al., 2025) interleaves latent visual tokens with text. These demonstrate that externalized visual intermediates help, but the schema of operations is hand-designed or selected ad-hoc by an LLM prompt – no prior system learns a modality-selection policy from task reward – and the intermediates are discrete operations over the input image rather than generated abstractions of scene geometry and motion. Our policy is learned, and our intermediates are full visual modalities (depth, flow, camera pose) rather than operations on the input image.

VLM hallucination and grounding. VLMs hallucinate objects, attributes, and relations not supported by the image (Li et al., 2023; Guan et al., 2023). Mitigations span training-time preference alignment (RLHF-V (Yu et al., 2024)) and decoding-time correction (OPERA (Huang et al., 2024), Visual Contrastive Decoding (Leng et al., 2023)). These largely target *object-level* hallucinations grounded in the input image, leaving the *geometric and dynamic* failures ("is the cup further than the ball?", "did it move left?") that arise when text-only reasoning has no explicit 3D or motion representation. By grounding reasoning in generated depth, flow, and camera pose, our approach is complementary and targets these failure modes.

RL for reasoning. RL with verifiable rewards (RLVR) has driven recent advances in language-model reasoning, optimizing chains of thought against automatically-checkable correctness signals. Our reward is likewise verifiable – the model's likelihood of the ground-truth answer – but the "thoughts" are visual abstractions rather than text. Because constructing the policy target requires only ranking precomputed candidate traces by reward, we adopt an *offline* update in the family of advantage-weighted regression and reward-weighted behavior cloning (Peng et al., 2019), which is stable and avoids online rollouts; we discuss the move to online RL in future work.

Self-improving frameworks, in which a model bootstraps its own training signal from its successes, have been demonstrated for *language* models – most notably STaR (Zelikman et al., 2022), which iteratively fine-tunes an LLM on the reasoning traces that lead it to correct answers. Our framework is the analogue for *vision*-language models: the bootstrapped rationales are streams of visual abstractions (depth, optical flow, camera pose) rather than text. We argue this distinction matters, because these abstractions are exactly what grounds reasoning about geometry and motion – the physical-understanding and hallucination failures that text-only self-improvement cannot reach.

3 Method

3.1 One Network as Policy and Answerer: Toward a Self-Improving Loop

The core of our method is that *a single network* is reused across the RL loop (Figure 1). In this project it plays two roles: the *policy*, which proposes a notation z – a stream of visual abstractions to invoke – for a given input, and the *answerer*, which conditions on those abstractions to answer the question. The abstraction *values* (depth, optical flow, camera pose) are tokenized and conditioned on; we do not use the model's own generative capability here. Reusing one network has two consequences that motivate the rest of the design. First, the reward is *self-supplied and verifiable*: it is the same model's probability of the correct answer, so "which abstraction helps" is measured in exactly the currency the answerer uses, with no separately-trained reward model to miscalibrate or exploit. Second, because the policy and answerer *share parameters*, learning to select better abstractions and learning to answer from them reinforce one another, rather than being a policy over frozen external tools. The backbone is itself generative (the PSI world model), so a natural third role – having the model *generate* its own abstractions instead of extracting them – would close a fully self-improving loop; we leave this to future work. Its action-conditioned world-modeling is a distinct capability, reserved for counterfactual planning rather than for the abstractions VQA needs.

Concretely, one turn of the loop (Figure 1) proceeds as follows. (1) Given a video-question, the *policy* proposes a notation – which abstractions to invoke, and in what order. (2) The selected abstractions are obtained – in this work, from off-the-shelf extractors – and tokenized. (3) The *answerer* conditions on them and answers, and we read off how likely it makes the correct answer. (4) Comparing that against an RGB-only baseline gives the reward: how much the chosen abstractions helped (the *advantage*, formalized in Section 3.3). (5) The policy is updated toward high-advantage notations. Iterating (1)–(5) with the improved policy is what makes the system *self-improving* rather than a static selector; in this work we run a single offline pass over a precomputed dataset (Sections 3.3–3.4) and leave online iteration to future work.

3.2 Architecture

The policy is a single 8B-parameter autoregressive transformer, initialized from a pretrained PSI checkpoint (Kotar et al., 2025) (Figure 2). It has 32 layers with model dimension 4096 and 32 attention heads of head dimension 128, using grouped-query attention with 8 key/value heads. The

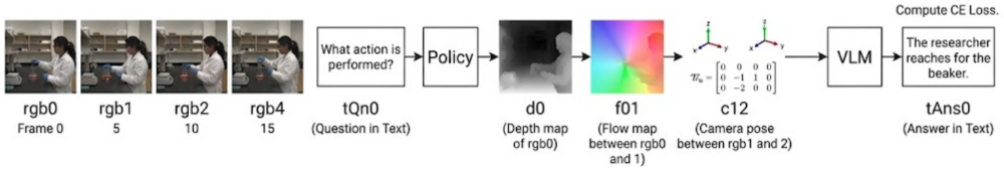

Figure 2: Model architecture. A single 8B-parameter autoregressive transformer over a unified token vocabulary (RGB, camera pose, depth, optical flow, text) emits visual abstractions as streams of thought. The same network acts as the policy that selects abstractions and the VLM that answers from them.

feed-forward blocks use a SwiGLU activation; we apply pre-norm RMSNorm, 3D rotary position embeddings, and fully causal attention, and we tie the input and output embeddings.

All modalities share *one* token vocabulary of 65,536 entries spanning RGB, camera pose (C), depth (D), optical flow (F), and text (T). Following PSI, the visual modalities use *local patch quantization*: each spatial patch is encoded independently and tagged with a *pointer token* specifying its location, so the model can generate patches and modalities in an arbitrary order rather than a fixed raster scan. Because modalities reuse overlapping token-id ranges, each token embedding is summed with a learned *channel embedding* that marks which modality it belongs to, giving every modality a distinct representation without changing the backbone. Abstraction selection is implemented in-stream: the model emits a notation – a comma-separated list of abstraction units – bracketed by `<notation>` ... `</notation>` within its reasoning trace, which triggers generation in the corresponding modality subspaces.

By construction the model is generative in two distinct senses: it can produce per-frame abstractions (depth, optical flow, camera pose), and – as the PSI world model – it can predict future frames conditioned on actions. We exercise neither in the present work. Having the model generate its own abstractions would close the self-improving VQA loop, whereas its action-conditioned world-modeling is what future work would use for counterfactual planning.

3.3 Offline RL via Advantage-Weighted Behavior Cloning

For a candidate notation z on a video-question (v, q, a), define the per-trace loss as the answer's negative log-likelihood, $\ell(z) = -\log p(a \mid v, q, z)$, and let $\ell_{\text{rgb}} = -\log p(a \mid v, q)$ be the RGB-only baseline that conditions on no extra abstractions. The *advantage* of a notation is the loss it removes,

$$A(z) \;=\; \ell_{\text{rgb}} - \ell(z), \tag{1}$$

which is positive exactly when the abstraction makes the correct answer more likely. Among the top-K=10 lowest-loss notations that beat the baseline ($A(z) > 0$), we softmax-sample a behavior-cloning target by loss, with a small penalty on notation length $|z|$ to favor efficient traces,

$$p(z) \;\propto\; \exp\!\Big(-\tfrac{\ell(z)-\ell_{\min}}{T}\Big)\, e^{-\lambda|z|}, \qquad T = 0.5,\ \lambda = 0.01, \tag{2}$$

where $\ell_{\min}$ is the lowest loss among the candidates (a numerical-stability shift). If no candidate beats the baseline, we fall back to the single lowest-loss trace. We then behavior-clone the policy to emit the sampled notation,

$$\max_{\theta}\ \mathbb{E}_{z\sim p}\big[\log \pi_\theta(z \mid v, q)\big]. \tag{3}$$

This is an offline, advantage-weighted update: it requires only ranking precomputed candidates by reward, never online rollouts, which makes training stable and cheap.

3.4 Data Generation and Scoring Pipeline

We construct the offline dataset with a two-stage pipeline. In stage one, for each video-question we sample 64 candidate notations – each 0–6 abstraction units drawn from per-frame depth, and camera pose and optical flow between frames; the value of each abstraction is produced by an off-the-shelf extractor and tokenized into the shared vocabulary. In stage two, we score every candidate by computing $\ell(z)$, the VLM's loss on the ground-truth answer when conditioned on that notation,

together with the RGB-only baseline ℓ_{rgb}. The result is $\sim$220k video-questions, each storing its 64 notations and their losses – a fixed offline dataset over which the advantage-weighted update above is applied.

3.5 Why Visual Abstractions Should Help

Three hypotheses motivate the approach. (i) *Visual chain-of-thought*: just as text chain-of-thought helps language models, externalizing intermediate *visual* steps helps VLMs – cropping and zooming into regions of interest (V* (Wu and Xie, 2023)), drawing or pointing annotations onto the image (Visual Sketchpad (Hu et al., 2024)), and chaining perceptual manipulations (CogCoM (Qi et al., 2025)) all improve grounded reasoning. Our intermediates are likewise "words" of thought, just not in text form; but where those systems operate *on the input image*, ours are generated full modalities of scene geometry and motion (depth, flow, camera pose). (ii) *Test-time decompression*: emitting intermediates unpacks a hard one-shot prediction into easier conditioned sub-steps, spending test-time compute where it helps. (iii) *Disentanglement*: visual abstractions such as depth and flow are low-dimensional and disentangled from appearance, exposing implicit scene properties in an explicit, accessible format.

4 Experimental Setup

Data. The offline dataset comprises $\sim$220k video-questions drawn from OpenVid (160k), EgoDex (28k), SSv2 (20k), and PE (12k) – a mix of action-rich egocentric and Internet videos – with captions rephrased into question–answer pairs. Each video-question carries 64 scored candidate notations as described above.

Evaluation. We evaluate on held-out test splits ($\sim$1,000 videos each) from the four datasets, using the same two-stage protocol (generate the policy's notation, then score the answer loss). Our primary metric is the VQA loss (answer negative log-likelihood); lower is better. We also report the fraction of test samples on which a method is $\geq 10\%$ better than a baseline.

Baselines. We compare the learned policy against two fixed strategies on the same backbone: *RGB-only*, which conditions on no extra abstractions, and *random*, which samples a notation uniformly rather than via the policy. These isolate, respectively, the value of using any abstraction and the value of *learning which* abstraction.

Scoping study. Before training the policy, we ran a design-space study (Figure 3) to test whether visual abstractions help VLM reasoning – and, as a secondary benefit, to confirm exploitable RL signal exists and to choose datasets and the notation search space. For each video we evaluated many sampled notations against the RGB-only baseline.

5 Results

5.1 Visual Abstractions Improve Grounded Reasoning (and Scope the RL Design Space)

First and most broadly, these results show that *visual abstractions improve VLM visually-grounded reasoning*: most sampled notations beat the RGB-only baseline, and on EgoDex the 90th-percentile notation reduces answer loss by 13%. As a secondary, project-specific benefit, this scopes the RL design space and answers our enabling question – is there signal for RL to exploit? – affirmatively. Longer notations tend to help more, which motivates the length penalty in our objective to avoid degenerate always-maximal traces. Most importantly for the policy, the best notation is *input-dependent*: it differs from video to video, so no single fixed pipeline can be optimal and a learned, per-input policy is warranted.

5.2 Quantitative Evaluation

Table 1 and Figure 4 report held-out VQA loss for the learned policy against the RGB-only and random baselines. The policy lowers answer loss over RGB-only on *every* dataset, and on a large fraction of individual samples improves by $\geq 10\%$. Relative to the stronger random-notation baseline, the policy is significantly better on SSv2, EgoDex, and PE, while on OpenVid the gap over random is

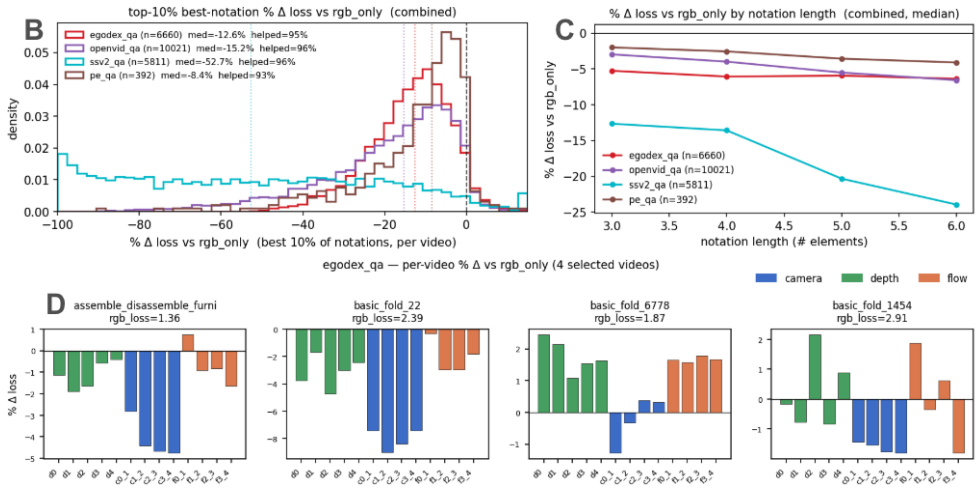

Figure 3: Visual abstractions improve grounded reasoning, and scope the RL design space. Most sampled notations beat the RGB-only baseline; longer notations tend to help more (motivating a length penalty); and the best stream-of-thought differs from video to video – i.e., the optimal abstraction is input-dependent.

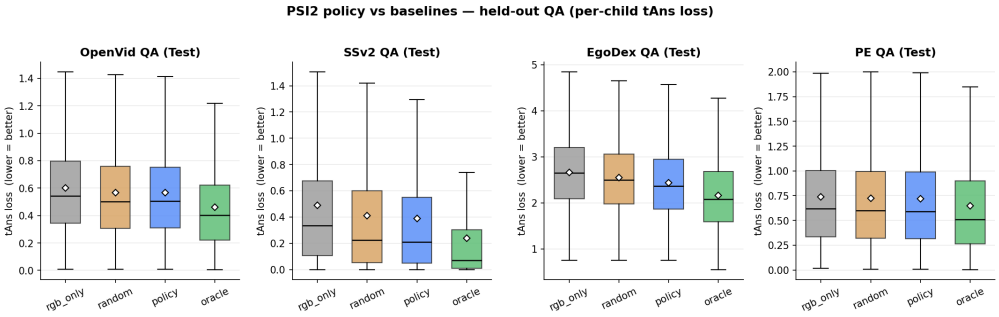

Figure 4: Held-out evaluation of the learned policy against the RGB-only and random-notation baselines across the four QA test sets.

not significant – consistent with OpenVid being the most diverse, in-the-wild split where notation choice matters least. Together these show that (i) invoking visual abstractions helps over text/RGB-only reasoning, and (ii) *learning which* abstraction to invoke adds further gains beyond random selection on most datasets.

6 Discussion

The central implication of our results is methodological: reusing a single network as both policy and answerer is what licenses the *self-improving* framing. The reward – the model's own answer likelihood – is verifiable without a separate critic, and the policy and answerer share parameters so they reinforce each other. Because the backbone is itself generative, having it generate its own abstractions would close a fully self-improving loop – the main direction we leave to future work. This targets a crucial weakness of current VLMs, namely ungrounded physical reasoning and the geometric/dynamic hallucinations it causes.

Two limitations temper the claim. First, the learned policy currently emits a *lack of diversity* in its notations; the training procedure needs to be adjusted to preserve exploration so the policy does not collapse onto a few notations. Second, the reward still relies on *external supervision* (ground-truth answers), so the present system is a single offline step toward self-improvement rather than a closed, label-free loop.

Table 1: PSI2 policy vs. baselines on held-out video QA (n=1000 per set). **Mean TANS loss** (↓): next-token cross-entropy under rgb-only, a random notation, and the policy's notation. **Policy gain**: relative loss reduction of policy over each baseline. **Win rate**: % of examples where policy loss < baseline. **#≥10% better**: examples where policy loss ≤ 0.90× baseline. Significance: paired t-test policy vs. random ($^{***}p<10^{-3}$, $^{**}p<10^{-2}$, n.s. otherwise).

	Mean TANS loss ↓			Policy gain (%)		Win rate (%)		#≥10% better	
Benchmark	rgb	rnd	**policy**	vs rgb	vs rnd	vs rgb	vs rnd	vs rgb	vs rnd
OpenVid QA (Test)	0.600	0.566	**0.565**	5.7	0.1 $^{\text{n.s.}}$	68.9	51.0	284	169
SSv2 QA (Test)	0.489	0.409	**0.390**	20.4	4.8 ***	73.4	53.5	627	413
EgoDex QA (Test)	2.655	2.552	**2.438**	8.2	4.5 ***	80.8	67.1	372	209
PE QA (Test)	0.738	0.721	**0.717**	2.8	0.7 **	66.4	51.6	178	109

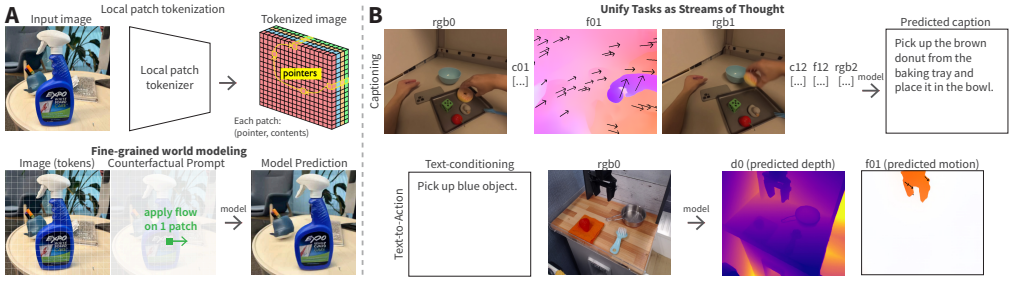

Figure 5: Two capabilities of the underlying model beyond standard VLMs. **(a)** Richly controllable world modeling: conditioning generation on any combination of modalities (optical flow, depth, text, point tracks), with patch-level independence enabling targeted object manipulations; perturbing a few points on an object yields a consistent predicted rollout of the scene (e.g., the induced optical flow), which future work would exploit for "what happens if" counterfactual planning over candidate actions in multi-step tasks. **(b)** Unifying diverse visual-linguistic tasks in a single model (e.g., captioning, text-to-action generation) by expressing each task as a stream of thought.

7 Conclusion

We presented an RL framework for self-improving VLMs that learn *what* visual abstraction to invoke and *when*, trained by advantage-weighted behavior cloning over a verifiable, self-supplied reward. A single network serving as both policy and answerer yields a self-supplied, verifiable reward, and our learned policy beats RGB-only across four held-out QA sets, with the best abstraction being input-dependent. We view this single offline round as an *entry* into a cycle of recursive self-improvement. Future work closes the loop *online* – regenerate notations with the improved policy, re-score, and retrain – replaces supervised reward with self-generated reward by placing the VLM in an environment where it explores, and exploits the world model for *counterfactual rollouts* to plan complex multi-step tasks such as assembling an IKEA table, where each step is simulated before it is committed.

The counterfactual capability we would rely on already exists in the underlying world model (Figure 5): conditioning generation on a hypothetical change – e.g., a small perturbation to a few points on an object – produces a consistent predicted rollout of the scene's response (such as the resulting optical flow). Turning this "what happens if" rendering into a planning signal – scoring candidate actions by the rollouts they induce and selecting the one that reaches the goal – is the bridge from single-step VQA to the multi-step reasoning above, and is the central direction we intend to pursue.

8 Team Contributions

- **Baihan Zhang:** data curation and the notation generation/scoring pipeline, baseline implementations, and held-out benchmark evaluation.

- **Khai Loong Aw:** model architecture extensions and backbone training, Google TPU setup, the offline advantage-weighted training, and held-out benchmark evaluation.

Changes from Proposal Our proposal planned online RL (PPO warm-started from imitation learning) evaluated against external spatial benchmarks (e.g., VSI-Bench, DSI-Bench, SAT). In this work we deliberately scoped to a single *offline* round of RL – advantage-weighted behavior cloning over precomputed, reward-scored notations – which is more stable and let us first establish that visual abstractions help and that exploitable signal exists (the design-space study). We correspondingly evaluate on held-out QA splits of our training distributions (OpenVid, SSv2, EgoDex, PE) rather than external benchmarks. Online RL and external-benchmark transfer remain the primary future work.

AI Tools Disclosure We used AI coding assistants to write boilerplate code and help debug our implementation, and AI writing assistants to improve the clarity of this report's writing. Research ideas, experimental design, and conclusions are our own.

References

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, et al. 2022. Flamingo: a Visual Language Model for Few-Shot Learning. arXiv:2204.14198 [cs.CV]

Peter W. Battaglia, Jessica B. Hamrick, and Joshua B. Tenenbaum. 2013. Simulation as an engine of physical scene understanding. *Proceedings of the National Academy of Sciences* 110, 45 (2013), 18327–18332.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, Dinesh Manocha, and Tianyi Zhou. 2023. HallusionBench: An Advanced Diagnostic Suite for Entangled Language Hallucination and Visual Illusion in Large Vision-Language Models. arXiv:2310.14566 [cs.CV] `https://arxiv.org/abs/2310.14566`

Yushi Hu, Weijia Shi, Xingyu Fu, Dan Roth, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and Ranjay Krishna. 2024. Visual Sketchpad: Sketching as a Visual Chain of Thought for Multimodal Language Models. arXiv:2406.09403 [cs.CV] `https://arxiv.org/abs/2406.09403`

Qidong Huang, Xiaoyi Dong, Pan Zhang, Bin Wang, Conghui He, Jiaqi Wang, Dahua Lin, Weiming Zhang, and Nenghai Yu. 2024. OPERA: Alleviating Hallucination in Multi-Modal Large Language Models via Over-Trust Penalty and Retrospection-Allocation. arXiv:2311.17911 [cs.CV] `https://arxiv.org/abs/2311.17911`

Klemen Kotar, Wanhee Lee, Rahul Venkatesh, Honglin Chen, Daniel Bear, Jared Watrous, Simon Kim, Khai Loong Aw, Lilian Naing Chen, Stefan Stojanov, Kevin Feigelis, Imran Thobani, Alex Durango, Khaled Jedoui, Atlas Kazemian, and Dan Yamins. 2025. World Modeling with Probabilistic Structure Integration. arXiv:2509.09737 [cs.CV] `https://arxiv.org/abs/2509.09737`

Sicong Leng, Hang Zhang, Guanzheng Chen, Xin Li, Shijian Lu, Chunyan Miao, and Lidong Bing. 2023. Mitigating Object Hallucinations in Large Vision-Language Models via Visual Contrastive Decoding. arXiv:2311.16922 [cs.CV] `https://arxiv.org/abs/2311.16922`

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Evaluating Object Hallucination in Large Vision-Language Models. arXiv:2305.10355 [cs.CV] `https://arxiv.org/abs/2305.10355`

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024. Improved Baselines with Visual Instruction Tuning. arXiv:2310.03744 [cs.CV]

Sachit Menon, Richard Zemel, and Carl Vondrick. 2024. Whiteboard-of-Thought: Thinking Step-by-Step Across Modalities. arXiv:2406.14562 [cs.CL] `https://arxiv.org/abs/2406.14562`

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. 2019. Advantage-Weighted Regression: Simple and Scalable Off-Policy Reinforcement Learning. arXiv:1910.00177 [cs.LG]

Ji Qi, Ming Ding, Weihan Wang, Yushi Bai, Qingsong Lv, Wenyi Hong, Bin Xu, Lei Hou, Juanzi Li, Yuxiao Dong, and Jie Tang. 2025. CogCoM: A Visual Language Model with Chain-of-Manipulations Reasoning. arXiv:2402.04236 [cs.CV] https://arxiv.org/abs/2402.04236

Roger N. Shepard and Jacqueline Metzler. 1971. Mental Rotation of Three-Dimensional Objects. *Science* 171, 3972 (1971), 701–703.

Penghao Wu and Saining Xie. 2023. V*: Guided Visual Search as a Core Mechanism in Multimodal LLMs. arXiv:2312.14135 [cs.CV] https://arxiv.org/abs/2312.14135

Zeyuan Yang, Xueyang Yu, Delin Chen, Maohao Shen, and Chuang Gan. 2025. Machine Mental Imagery: Empower Multimodal Reasoning with Latent Visual Tokens. arXiv:2506.17218 [cs.CV] https://arxiv.org/abs/2506.17218

Tianyu Yu, Yuan Yao, Haoye Zhang, Taiwen He, Yifeng Han, Ganqu Cui, Jinyi Hu, Zhiyuan Liu, Hai-Tao Zheng, Maosong Sun, and Tat-Seng Chua. 2024. RLHF-V: Towards Trustworthy MLLMs via Behavior Alignment from Fine-grained Correctional Human Feedback. arXiv:2312.00849 [cs.CV] https://arxiv.org/abs/2312.00849

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. STaR: Bootstrapping Reasoning With Reasoning. arXiv:2203.14465 [cs.LG] https://arxiv.org/abs/2203.14465

A Implementation Details

Table 2 summarizes the model architecture and the offline-RL training hyperparameters.

Table 2: Architecture and training hyperparameters.

Architecture	
Parameters	$\sim$8B
Layers	32
Model dimension	4096
Attention heads	32
Key/value heads (GQA)	8
Head dimension	128
FFN activation	SwiGLU
Normalization	pre-norm RMSNorm
Position encoding	3D rotary (RoPE)
Attention	causal
Vocabulary size	65,536
Embeddings	input/output tied
Offline RL training	
Candidate notations / sample (K)	10
Softmax temperature (T)	0.5
Length penalty (λ)	0.01
Notation markers	<notation> / </notation>