Extended Abstract

Motivation. This project began from a neuroscience question, how people remember what they see and which parts of a scene they actually keep. A bounded memory system cannot retain every feature of an image, so it has to choose what to keep, and the useful choice depends on a question that has not been asked yet. A query about whether a scene contained food needs semantic content, a query about the dominant color needs perceptual content, and a query about which of several images was seen needs distinguishing detail. We study the version of this problem in which an agent commits a small fixed number of feature units to memory before the query is revealed, then answers from the retained units alone. The question is whether learning what to keep beats compressing the image blindly at the same budget.

Method. For each image we extract a fixed menu of at most fifteen typed feature vectors from three frozen families, a language-aligned SigLIP gist, a self-supervised DINOv2 gist, up to twelve per-object DINOv2 slots pooled inside COCO instance masks, and one low-level color and texture summary. A selection policy scores the candidates with a two-layer set-Transformer and samples exactly K of them without replacement using a Plackett-Luce head, which gives a valid size-K action and an exact log-probability. The retained units pass to two frozen probes, an attribute question-answering head and a contrastive recognition head, whose accuracies form a non-differentiable reward. We treat one image as a single-step contextual bandit and train the policy with Group-Relative Policy Optimization (GRPO), using the within-group reward mean as a value-free baseline. No gradient flows through the probes.

Implementation. We use COCO val2017, split 4000/500/500 over image ids, with the same split reused for the selector and both probes. The probes are pretrained on random K-subsets and then frozen, so all reward variation reflects the selection. The reward is an equal mix of attribute accuracy and recognition, where recognition is top-1 retrieval of the retained trace against a shuffled fifty-way lineup of the image's hardest semantic neighbors. For each image GRPO draws eight stochastic selections and forms the advantage from their within-group reward statistics, and we train one policy at each budget $K \in \{1, 2, 4\}$. We compare against three discrete baselines that share the menu interface, random, globals-only, and a mask-area saliency rule, and against three continuous compressors that squeeze the full feature pool into a K-dimensional latent, PCA, an autoencoder, and a variational information bottleneck.

Results. On the held-out test split the learned policy reaches a combined reward of 0.908 at the operating budget $K{=}2$, above random (0.831) and the strongest discrete heuristic (0.864), and it is the only discrete selector strong on both reward axes, scoring 0.886 on attributes and 0.930 on recognition. The saliency rule edges it on attributes (0.899) but trails it by ten points on recognition, and the continuous compressors lead only on a recognition axis confounded by shared query-gallery encoding while falling below the policy on the cleaner attribute comparison. The central result is the budget sweep. The advantage of learned selection over random is largest exactly where memory is scarce, $+0.153$ reward at $K{=}1$, halves to $+0.077$ at $K{=}2$, and closes to within noise by $K{=}4$, where the recognition probe saturates and any reasonable selection succeeds. Learning what to keep matters precisely when the budget binds, and a reward-mix ablation sharpens the picture by showing that training on recognition alone already preserves attribute accuracy.

Conclusion. A value-free policy gradient learns task-conditioned discrete memory selection from a typed multi-family menu under a delayed, non-differentiable reward. It beats random selection and three discrete heuristics at the tight budgets where memory is scarce, is the only discrete method strong on both task axes, and matches the continuous compressors on the one axis that compares them fairly. The project's central finding is that the value of learned forgetting is governed by the memory budget, largest when the budget is tight and fading as the task saturates, which turns "what should a bounded memory keep" into a question with a measurable answer.

Learned Forgetting: Task-Conditioned Visual Memory Selection via Reinforcement Learning

Han (Harrison) Shaun Lee
Department of Computer Science
Stanford University
hanl@stanford.edu

Abstract

We study task-conditioned visual memory selection, where an agent commits a small fixed number of feature units to memory before a query is revealed and then answers from the retained units alone. We cast a single image as a single-step contextual bandit. A set-Transformer policy scores a menu of at most fifteen typed candidate vectors, drawn from frozen SigLIP, DINOv2, and low-level extractors, and a Plackett-Luce head samples exactly K of them without replacement, which gives a valid action and an exact log-probability. Two frozen probes, attribute question answering and contrastive recognition, turn the retained set into a non-differentiable reward, and we train the policy with Group-Relative Policy Optimization. On the COCO val2017 test split the learned policy beats random selection and three discrete heuristics, is the only discrete method strong on both attribute and recognition accuracy at the operating budget $K{=}2$, and outperforms PCA, autoencoder, and variational information bottleneck compressors on attribute accuracy, the axis that compares them fairly because continuous recognition is confounded by shared query-gallery encoding. Our central result is a budget sweep showing that the advantage of learned selection is largest where memory is scarcest, $+0.153$ reward over random at $K{=}1$, and fades to within noise by $K{=}4$ where the task saturates. The value of choosing what to keep is therefore set by the memory budget. We report evaluation-set confidence intervals that separate the policy from random selection on recognition at $K{=}2$.

1 Introduction

The starting point for this project came from neuroscience, from wanting to understand how people remember what they see. Human visual memory is selective and reconstructive rather than photographic. We keep the parts of a scene that later turn out to matter and discard the rest, and different aspects of a scene, the gist of its category, the identity of a particular object, and its low-level appearance, are carried by different parts of the visual system. That selectivity is part of what makes recall efficient, and it is what the multi-family menu below is meant to echo. Turning the question into something trainable led to the narrower computational version studied here.

The computational task is concrete. After seeing an image, an agent keeps only K feature units from a fixed menu, before learning which downstream query will be asked, and the retained subset is then graded by frozen probes for attribute question answering and recognition. The policy receives only the probes' discrete accuracy as reward, so the selector cannot be trained by ordinary backpropagation through the task. The problem is not how to compress an image into one shorter vector. It is which named pieces of the representation to retain when only a few fit.

This is a reinforcement learning problem for three reasons. First, the reward is non-differentiable. The retained units are read by frozen probes whose decisions, an argmax over a retrieval lineup and

a thresholded yes-no answer, give no gradient to the selection and no supervised target for which units are correct. Second, the action space is combinatorial, with $\binom{15}{K}$ admissible selections and a reward that couples the chosen units through cross-attention in the probes. Third, the agent must explore, since it has to find which of several typed families carries the signal for a query it has not seen. Reward design also mattered in practice. An early version of the recognition reward gave empty selections a perfect score until we shuffled the lineup, a bug reported in Section 5.

We make three contributions. We build a typed multi-family selection environment in which a delayed, multi-task reward is produced by frozen probes, so that reward variation is attributable to the selection alone. We train a value-free policy gradient, GRPO, to perform hard, discrete, exactly-size-K selection over this menu, with a Plackett-Luce head that gives a clean exact log-probability. We show, with matched-budget baselines, a reward-mix ablation, and evaluation-set confidence intervals, that learned selection helps most when memory is scarce and fades as the budget grows, which identifies the budgets where selection quality actually changes the reward.

2 Related Work

Reinforcement learning for selective vision. The Recurrent Attention Model Mnih et al. [2014] trains a REINFORCE Williams [1992] agent that decides where to look next in an image, accumulating glimpses toward a classification decision. Our agent instead decides what to keep from an already extracted menu, and the reward is delayed and multi-task rather than a single per-step label. We share the use of a policy gradient over a discrete, non-differentiable decision, and we replace the bootstrapped sequential credit assignment with a single-step bandit so the gradient is exact, which is also why a group-relative baseline is natural here.

Learned discrete token selection. DynamicViT Rao et al. [2021] sparsifies tokens inside a single vision transformer with a Gumbel-Softmax Jang et al. [2017] relaxation, supervised by a classification loss that flows through the kept tokens. This is the closest prior work, and it differs from our setting in three ways. Its candidates come from one backbone, ours span several foundation models with different abstraction levels. Its objective is differentiable through the selection, ours is a delayed reward read by frozen probes, so a relaxation cannot pass gradient to the choice. Its goal is throughput, ours is task-conditioned retention. Gumbel-Softmax is therefore the natural non-reinforcement comparator for our problem rather than a drop-in method.

Continuous compression and the information bottleneck. The information bottleneck Tishby et al. [1999] frames task-relevant compression as a constrained mutual-information objective, and its deep variational form Alemi et al. [2017] is the variational information bottleneck we use as a baseline. PCA and a plain autoencoder are the task-agnostic linear and nonlinear analogues. These methods sit at the opposite end of a design space from our agent, since they compress every input into a fixed continuous code rather than picking an interpretable subset of named units. We do not claim a matched bit budget. Our selector keeps K named 256-dimensional projected tokens, whereas the continuous baselines compress to K scalar dimensions, so we read them as representation-style comparisons rather than storage-equivalent competitors.

Object-centric region features. Our candidate menu connects to region-based visual representations used in detection, captioning, and visual question answering. Rather than representing an image only as a grid of patches, those methods reason over semantically meaningful regions, and our object slots follow the same intuition, since COCO instance masks give spatially grounded regions that make the selector's decisions interpretable. We do not train a question-answering model end to end. We use annotation-derived yes-no questions as a frozen diagnostic probe, which narrows the evaluation but isolates the memory-selection problem from language-model capacity.

Group-relative policy optimization. GRPO Shao et al. [2024] estimates the advantage from the empirical reward statistics of a group of samples drawn for the same prompt, removing the learned value network used in PPO Schulman et al. [2017]. It was developed for language-model fine-tuning, where it sits alongside RLHF-style reward optimization Ouyang et al. [2022]. We borrow its group baseline for a vision selection task, where a single image plays the role of the prompt and a group of stochastic selections supplies the within-group statistics. Because our episode is single-step, the group mean is an exact per-state baseline and the policy gradient has no bootstrapping error.

3 Method

3.1 Problem formulation

We model one image as a single-step contextual bandit. The agent observes the menu as context, commits to a size-K subset as its action, and receives a terminal scalar reward. There is no temporal credit assignment, so the policy gradient is exact. Table 1 summarizes the formulation. The objective is

$$\max_{\theta} \mathbb{E}_s \, \mathbb{E}_{a \sim \pi_\theta(\cdot|s)} \big[R(s,a) \big],$$

where s is the menu for one image, a is a binary mask with $\|a\|_1 = K$, and $R(s,a) \in [0,1]$ is frozen-probe accuracy on the retained units.

Table 1: The single-step contextual bandit for one image.

Element	Definition
State s	Menu of at most 15 typed candidate vectors for one image
Action a	Subset mask $a \in \{0,1\}^{15}$ with $\|a\|_1 = K$
Policy π_θ	Set-Transformer logits, then Plackett-Luce sampling
Reward $R(s,a)$	Frozen-probe accuracy in $[0,1]$, non-differentiable
Objective	$\max_\theta \mathbb{E}_{s,\, a \sim \pi_\theta}[R(s,a)]$

3.2 Feature menu

The menu mixes complementary families so that different queries can draw on different units. For each image we build at most fifteen candidates, listed in Table 2. The SigLIP global token Zhai et al. [2023] carries language-aligned semantic content. The DINOv2 CLS token Oquab et al. [2024] carries a self-supervised perceptual gist. The object slots are per-object DINOv2 features obtained by mean-pooling the patch grid inside each COCO instance mask Lin et al. [2014], stored in descending mask-area order so that slot one is the largest object. We chose this object-as-slot formulation because it makes selection semantically meaningful, since the agent can keep a dog rather than the background instead of an arbitrary patch. A single low-level vector concatenates an HSV color histogram with Gabor texture statistics, capturing raw appearance that the larger backbones tend to abstract away. Object slots beyond the number of instances in an image are zeroed and masked out by a per-image validity mask.

The four families arrive in different native dimensions, 768 for the two gists and the object slots and 108 for the low-level vector, so before any of them can be compared they are mapped into one common space. A per-family linear adapter projects each unit to a shared width of 256, and a learned family-type embedding is added so the token still records which family produced it. The DINOv2 CLS token and the object slots run through the same adapter and are separated only by that family tag. After this step the menu is a single homogeneous set of at most fifteen 256-dimensional tokens, and every later operation, scoring in the policy and attention in the probes, happens inside this shared pool rather than within a family. The policy and the two probes each instantiate the front-end with their own weights, so the shared-pool layout is a common design pattern, not a shared set of parameters.

Table 2: The fixed candidate menu, at most 15 typed units per image.

Family	Content	Dim	Count
SigLIP global	Language-aligned semantic gist	768	1
DINOv2 CLS	Self-supervised perceptual gist	768	1
DINOv2 slots	Per-object mean-pooled patches (COCO masks)	768	≤ 12
Low-level	HSV histogram and Gabor statistics	108	1

3.3 Selection policy

A two-layer set-Transformer Lee et al. [2019] encodes the menu with cross-candidate attention and a small head emits one scalar score per candidate,

$$(s_1, \ldots, s_M) = \mathrm{SetTransformer}\big(\{x_i\}_{i=1}^{M};\, \theta\big).$$

Because the encoder runs over the single shared pool, each score is relative to the other candidates in the same image. A unit is kept because it beats its peers in that pool, not because it clears any per-family threshold, so attending across candidates lets the policy recognize redundancy, for example two near-duplicate object slots, and prefer a more diverse retained set. Padded slots are excluded with a key-padding mask, and invalid positions are set to $-\infty$ so they can never be selected, which lets a single permutation-equivariant network handle the variable-length menu.

We turn scores into a size-K action with Plackett-Luce sampling without replacement Plackett [1975], Luce [1959]. We draw candidates one at a time, each from a softmax over the remaining valid candidates, and remove each drawn item before the next step. The log-probability of the full selection decomposes into K sequential-softmax terms,

$$\log \pi_\theta(a \mid s) = \sum_{k=1}^{K} \log \frac{\exp(s_{i_k})}{\sum_{j \notin \{i_1,\dots,i_{k-1}\}} \exp(s_j)}.$$

This guarantees exactly K distinct valid picks, or all valid candidates when fewer than K exist, and gives an exact, low-variance log-probability for the policy gradient. At evaluation we replace sampling with greedy top-K for a deterministic decoding.

3.4 Frozen probe environment

The reward is produced by two probes that are pretrained and then frozen, so that all reward variation during policy training reflects the selection rather than probe drift. The attribute probe takes the retained units and a question-template embedding, runs a two-layer set-Transformer over the question token concatenated with the candidates, and reads a binary yes-no answer from the question token's output. It is trained with binary cross-entropy on questions drawn from a bank of 87 templates, namely 80 object-presence questions over the COCO categories, four spatial-half questions (left, right, top, bottom), and three count questions with thresholds $N \in \{1, 3, 5\}$, sampled with a bias toward caption-relevant categories. The caption-relevance weighting applies only to the object-presence questions. The recognition probe pools the retained units with a single learned attention query and projects to a 32-dimensional L2-normalized embedding. It is trained contrastively with InfoNCE van den Oord et al. [2018] and in-batch negatives at temperature 0.07, with the anchor a random K-subset of the menu and the reference the full menu, so that a sparse trace embeds close to its own image and far from others. The embedding is kept small on purpose. A wider embedding has enough capacity to identify all 500 evaluation images trivially, which would remove the signal the selector is supposed to learn.

Training both probes on random K-subsets matters. An earlier recognition probe used independent Bernoulli dropout over candidates while protecting the two global tokens from being dropped, and it learned a degenerate solution that identified images almost entirely from the always-present globals and treated the object slots as noise. Switching to exact K-sampling with no protected positions removed that behavior and produced a probe whose embedding depends on the contents of the selected subset.

Given a selection mask, the reward combines the two probes,

$$R(s, a) = \alpha\, R_{\text{attr}}(s, a) + \beta\, R_{\text{recog}}(s, a), \qquad \alpha = \beta = 0.5.$$

R_{attr} is the mean correctness over four caption-weighted questions, and R_{recog} is the indicator that the trace embedding retrieves the true image as top-1 in a shuffled 50-way lineup whose distractors are the image's 49 semantic nearest neighbors in SigLIP space. The lineup is shuffled and the true position is tracked at every reward call. This removes a tie-breaking artifact, discussed in Section 5, in which a degenerate selection scored a perfect recognition.

3.5 Training with GRPO

We optimize the policy with Group-Relative Policy Optimization Shao et al. [2024]. For each image we draw $G{=}8$ stochastic selections and standardize their rewards within the group,

$$\hat{A}_j = \frac{R_j - \mu_G}{\sigma_G + \varepsilon}, \qquad \mu_G = \frac{1}{G} \sum_j R_j, \qquad \sigma_G = \operatorname{std}_j R_j.$$

The loss is the policy gradient with an entropy bonus on the first-pick distribution,

$$\mathcal{L}(\theta) = -\,\mathbb{E}_j\big[\hat{A}_j \log \pi_\theta(a_j \mid s)\big] - \lambda\, H(\pi_\theta), \qquad \lambda = 0.01.$$

Because the reward is bounded in $[0, 1]$ the group standardization is stable, and the group mean is a free, per-state baseline with no critic to train as in PPO and no separately tuned running mean as in plain REINFORCE, which is exactly what a reward this sparse and binary calls for. The advantage is detached, so the gradient flows only through $\log \pi_\theta$ and never through the frozen probes. The entropy term keeps early exploration broad so the policy does not collapse onto a single family before it has seen enough queries. We use Adam at 3×10^{-4}, a batch of 32 images with $G{=}8$ samples each for an effective batch of 256, 10 epochs, and one policy trained per budget K. We keep the checkpoint with the best greedy validation reward.

4 Experimental Setup

Data and splits. We use COCO val2017 Lin et al. [2014], which has 5000 images, 80 categories, and instance masks, the last being essential since the masks define the regions for object-slot pooling. We split the image ids 4000/500/500 into train, validation, and test with seed 231, and the same split is reused for the selector and both probes so that no validation or test image is seen during any training stage. Features for all three families are extracted once and cached. Hard recognition lineups use each image's 50 nearest semantic and perceptual neighbors, computed over the full 5000-image set so a distractor gallery is available for every split. The neighbor computation uses frozen features only and serves lineup construction, not the training of the selector or the probes, so it does not leak held-out labels into learning.

Metrics and reporting. For the probes we report balanced binary accuracy for the attribute task and top-1 retrieval for recognition. For selectors and baselines we report the combined reward and its two components on the 500-image test split, averaged over images. The validation split was used only for model selection during development, and the operating budget, the recognition embedding dimension, and the choice of K-sampling over dropout were all fixed on validation. We report balanced attribute accuracy rather than raw accuracy because object-presence questions are skewed, since most categories are absent from most images. Without the correction, a probe could earn credit from answer-frequency bias instead of from retained visual information.

Uncertainty. Recognition reward is a Bernoulli retrieval outcome over $n{=}500$ test images, and attribute reward averages over $4{\times}500$ binary question outcomes, so we attach Wald 95% confidence intervals, $\hat{p} \pm 1.96\sqrt{\hat{p}(1-\hat{p})/n}$, to both accuracies. These quantify sampling over the held-out test set and separate the learned policy from random selection on recognition at the operating budget.

Baselines. Three discrete baselines share the menu-mask interface with the learned policy and are scored by the same reward. **Random** samples K valid candidates uniformly. **Globals-only** always commits to the SigLIP and DINOv2 gists, then low-level, a committed-summary floor. **Saliency** keeps the top-K object slots by mask area, with globals as fallback when an image has fewer than K slots. Mask area is a proxy for the per-patch attention norm named in the original plan, since caching per-patch attention was deferred, and larger masks aggregate more high-attention patches. Three continuous baselines compress a 2412-dimensional pooled feature vector, the concatenation of SigLIP, DINOv2 CLS, the mean of valid slots, and the low-level vector, into a K-dimensional latent. **PCA** takes the top-K principal directions of the train pool. The **autoencoder** learns a nonlinear bottleneck-K code on reconstruction with no task signal. The **variational information bottleneck** Alemi et al. [2017] learns a K-dimensional Gaussian latent jointly with its probes under a task loss plus a KL penalty to a standard normal. Each continuous method consumes its own small attribute and recognition probe pair trained on the latent directly, since the token-shaped probes expect a menu of typed tokens.

5 Results

5.1 Quantitative Evaluation

5.1.1 Main comparison at the operating budget

Table 3 reports every method at $K{=}2$ on the test split, with confidence intervals on the two task axes. The learned policy reaches a combined reward of 0.908, above random, globals-only, and saliency, and among the discrete methods it is the only selector that does not trade one axis for the other, with attribute 0.886 and recognition 0.930. The heuristics each win one axis and lose the other. Saliency edges the policy on attributes, 0.899 against 0.886, because the largest object usually answers a question of the form "was there an X," yet being task-agnostic it trails by ten points on identity, 0.828 against 0.930, with non-overlapping intervals. The continuous compressors reach a perfect 1.000 on recognition, but this is an artifact of how their gallery is built. Each continuous method encodes both the query and the gallery with the same compressor, so the retrieval target sits on top of the query and the lineup is trivial. The cleaner cross-method comparison is therefore attribute accuracy, where the order is saliency $0.899 >$ policy $0.886 >$ random $0.856 >$ autoencoder $0.848 >$ PCA $0.823 >$ globals $0.812 >$ VIB 0.807. On that axis GRPO sits behind saliency but ahead of every learned baseline, and unlike saliency it also keeps high recognition accuracy. The four reference rows form a strict ordering on combined reward, $0.286 < 0.735 < 0.831 < 0.957$, which shows the reward discriminates selection quality across the full range.

Table 3: Main comparison at $K{=}2$ on the 500-image test split, semantic-NN distractors. Confidence intervals are Wald 95% over the evaluation set ($n{=}500$ for recognition, $n{=}2000$ for attribute). †Continuous recognition is 1.000 because query and gallery share a compressor, so the fair cross-method axis is Attribute.

Selector	Reward	Attribute (95% CI)	Recognition (95% CI)
Empty (floor)	0.286	0.559	0.014
Globals-only	0.735	0.812 (0.795, 0.829)	0.658 (0.616, 0.700)
Random $K{=}2$	0.831	0.856 (0.841, 0.871)	0.806 (0.771, 0.841)
Saliency (mask area)	0.864	**0.899** (0.886, 0.912)	0.828 (0.795, 0.861)
PCA (continuous)	0.912	0.823 (0.806, 0.840)	1.000†
Autoencoder (continuous)	0.924	0.848 (0.832, 0.864)	1.000†
VIB (continuous)	0.903	0.807 (0.790, 0.824)	1.000†
GRPO (ours)	**0.908**	0.886 (0.872, 0.900)	**0.930** (0.908, 0.952)
Full menu (ceiling)	0.957	0.915	1.000

5.1.2 Effect of the memory budget

The budget sweep is the most reliable result, since it shows both when the method helps and when it stops. Table 4 reports test combined reward at $K \in \{1, 2, 4\}$ for the policy and the three discrete baselines, and Figure 2 in the appendix plots the same data. The policy's advantage over random is largest at the tightest budget, $+0.153$ at $K{=}1$, halves to $+0.077$ at $K{=}2$, and turns into a small loss within noise at $K{=}4$. At $K{=}1$ it also clears the strong saliency heuristic by a clear margin, 0.816 against 0.760.

Table 4: Combined reward on the test split across budgets K, and the learned policy's lift over random. Learned selection helps most at the tightest budget and fades as the task saturates.

K	Random	Globals	Saliency	**GRPO (ours)**	Lift vs. random
1	0.663	0.285	0.760	**0.816**	$+0.153$
2	0.831	0.735	0.864	**0.908**	$+0.077$
4	0.952	0.687	0.949	0.942	-0.010

The two ends of the sweep are the same mechanism seen from opposite sides. At $K{=}4$ the recognition probe is saturated, as Table 5 shows, so almost any selection answers the queries and the methods converge. At $K{=}1$ the bottleneck is at its tightest, and the gap between a learned pick (0.816) and a

committed global gist (0.285, below random's 0.663) shows that the single global token carries far too little fingerprint detail to identify an image against hard distractors. Section 5.2 returns to why the convergence at $K{=}4$ is an optimization fact about GRPO and not only a property of the probe.

Table 5: Recognition probe top-1 accuracy by number of retained units, 50-way validation lineup, chance $= 0.020$. The probe saturates by $K{=}4$ even against hard distractors, which motivates $K{=}2$ as the operating budget.

Units K	Random distractors	Semantic NN	Perceptual NN
1	0.700	0.602	0.608
2	0.924	0.858	0.858
4	1.000	0.988	0.988

5.1.3 Reward-mix ablation

To check whether the policy simply optimizes one reward component, we trained two additional $K{=}2$ policies with extreme weights, attribute-only ($\alpha{=}1, \beta{=}0$) and recognition-only ($\alpha{=}0, \beta{=}1$), and scored all three under the default balanced reward so the numbers are comparable. Table 6 reports the result. The recognition-only specialist slightly exceeds the balanced default on combined reward and on recognition, and essentially matches it on attribute, 0.881 against 0.886. The asymmetry runs the other way for attribute-only training, which gains 1.7 points on attribute, 0.903 against 0.886, at a cost of 10.6 points on recognition, 0.824 against 0.930, for a lower combined reward. The two specialists are not mirror images, then, and Section 5.2 reads the gap between them as a statement about what each objective forces the policy to keep. The numbers here also bound the headroom. Even the attribute-specialized policy beats the saliency heuristic on attribute by only 0.004, within noise, so on the current question bank the attribute ceiling is set by the questions rather than by the selector.

Table 6: Reward-mix ablation at $K{=}2$ on the test split. All three policies are scored under the default $\alpha{=}\beta{=}0.5$ reward. Recognition-only essentially matches the balanced default, while attribute-only trades recognition for a marginal attribute gain.

Policy variant ($K{=}2$)	Reward	Attribute	Recognition
Random (reference)	0.831	0.856	0.806
Saliency (reference)	0.864	0.899	0.828
Attribute-only ($\alpha{=}1, \beta{=}0$)	0.864	**0.903**	0.824
Default ($\alpha{=}\beta{=}0.5$)	0.908	0.886	0.930
Recognition-only ($\alpha{=}0, \beta{=}1$)	**0.913**	0.881	**0.946**

5.2 Qualitative Analysis

5.2.1 The tradeoff between attribute and recognition accuracy

On the two axes the reward sums, attribute and recognition (Figure 3, appendix), the learned policy is the only method sharing the menu interface that is not dominated. Saliency reaches comparable attribute accuracy but trails on recognition, globals-only is low on both, and GRPO sits closest to the full-menu ceiling in the upper right.

The interesting part is why GRPO and saliency separate, and the menu plus the lineup make the answer concrete. Saliency is a fixed rule that keeps the two largest-area object slots, the first two positions in the area-sorted layout. The recognition distractors, however, are each image's semantic nearest neighbors, so the two largest objects are exactly the wrong thing to keep against them. A big, prototypical object is the part of the scene a semantic neighbor is most likely to share, which makes it a weak discriminator. GRPO matches saliency on attribute, so it keeps object information too, yet it wins recognition by ten points, which at a budget of two is only possible if it spends a pick differently. Where saliency commits both picks to the two largest objects, the units that separate an image from a semantic neighbor are the ones mask area is blind to, the DINOv2 gist and a more distinctive, often smaller object slot chosen by content rather than by area, and these are what the recognition reward pushes the policy toward. The cross-candidate attention makes the move reachable. Scoring each

unit against the others in the shared pool, the encoder can see that a second near-prototypical slot is redundant with the first and prefer a complementary unit. The logged selections in Section 5.2.3 bear this out directly, the policy keeps the DINOv2 gist in most images and spends its remaining pick on a single content-bearing slot rather than the two largest objects.

The three continuous compressors form their own cluster at recognition 1.000, far to the left on attribute, which turns the gallery artifact into a picture rather than a caveat. They are pinned to the ceiling on the confounded axis and sit at or below random on the honest one.

5.2.2 What the two reward specialists keep

The reward-mix specialists in Table 6 act as a behavioral probe. Because both were trained with the same menu and architecture and differ only in the reward weights, the gap between them measures what each objective forces the policy to keep. Attribute-only training trades most of its recognition for a small attribute gain, the signature of a policy that keeps whatever answers an object-presence question and discards the detail that separates an image from its neighbors. Recognition-only training shows the more informative direction. With no attribute reward at all it still matches the balanced policy on attribute, so the units needed to make an image identifiable already carry most of what the attribute probe asks for. Identity-preserving selections are attribute-sufficient, but attribute-sufficient selections are not identity-preserving, which is the same ordering the frontier shows and the same reason recognition, not attribute, is the axis a tight budget makes hard.

5.2.3 What the policy keeps, read from logged selections

The reading above is behavioral, inferred from rewards rather than from the picks themselves. To make it a direct measurement we ran each trained selector in greedy mode on the 500 test images, logged every retained menu position, and aggregated the picks by feature family, by menu position, and by the COCO category cached with each object slot. Across all three $K{=}2$ variants the policy converges on one pattern, the DINOv2 gist token plus a single object slot, which together account for at least 99.5% of all picks. The SigLIP global is kept in under 0.5% of cases and the low-level color and texture summary in under 0.2%, so both are effectively forgotten. The policy was never told to do this. It discovered on its own that, at a budget of two, the best use of memory is one token for what kind of scene this is and one for the object that anchors it, that the SigLIP gist is redundant with the DINOv2 gist, and that the color histogram rarely earns a slot.

The reward weights then move this allocation in interpretable directions (Table 7). The attribute-only specialist hardens its reliance on the gist, keeping the DINOv2 CLS token in 99.2% of test images and spending its remaining pick on an object slot. The recognition-only specialist does the opposite, keeping the gist in only 77.8% of images and placing more of its budget on object slots (60.7% of picks). That is a 21-point swing in gist retention produced by nothing but the reward mix, and it matches the structure of the two tasks. The attribute probe asks mostly binary, category-level questions that the gist answers cheaply, whereas the recognition probe must single an image out from 49 near neighbors, which forces the policy to keep object-level detail. This is the direct form of the asymmetry in the previous section, identity-preserving selection demands per-image object detail while attribute-sufficient selection can lean on the category gist. The budget rows of Table 7 echo the sweep, at $K{=}1$ the policy collapses almost entirely onto object slots, and only at $K{=}4$ are all four families used non-trivially.

Table 7: Share of greedy selections by feature family on the 500-image test split, so each $K{=}2$ variant contributes 1000 picks. Values are percentages within a row. The default $K{=}2$ policy splits between the DINOv2 gist and an object slot, attribute-only leans toward the gist, and recognition-only toward object slots.

Selector	SigLIP	DINOv2 CLS	Object slot	Low-level
Default $K{=}2$	0.4	40.5	59.1	0.0
Attribute-only $K{=}2$	0.4	49.6	50.0	0.0
Recognition-only $K{=}2$	0.2	38.9	60.7	0.2
Default $K{=}1$	0.0	24.2	75.8	0.0
Default $K{=}4$	10.2	16.9	69.6	3.3

Figure 1: COCO val2017 image 14831 as a worked example. Each $K{=}2$ variant keeps the DINOv2 gist plus one object slot, and the highlighted region is the slot each one chose. The default and recognition-only policies keep the bed, the scene-defining object that distinguishes this bedroom from its semantic neighbors, while the attribute-only policy keeps the cat, a rarer category that more cheaply answers an object-presence question. The reward mix changes which object the agent remembers from the same image.

Recovering the COCO category at each retained slot tells the same story in object space. "Person" is the most-kept category for every variant, unsurprising given how often people appear in COCO, but the tails diverge. The default and recognition-only selectors then favor scene-defining objects such as car, chair, and bed, while the attribute-only specialist more often keeps single-instance objects such as toilet, tennis racket, and bus, each of which cleanly answers one object-presence question. Figure 1 makes the contrast concrete on a single image. On a photograph of a cat on a bed, the default and recognition-only policies keep the bed, the scene-defining object that, with the gist, separates this bedroom from its semantic neighbors, while the attribute-only policy keeps the cat, the rarer category that more cheaply answers a yes-no question. Same image, same menu, different reward, different memory. The logged selections therefore confirm that the reward-mix variants are specialized policies rather than noisy copies of one strategy, and that the learned policy implements a readable rule, keep the gist, then keep the object this particular image is about.

That asymmetry also explains why the attribute axis is hard to win by learning at all, and the reason is in the evaluation rather than the policy. Object-presence templates are 80 of the 87 questions, and the reward samples a caption-relevant one with probability 0.7, so most graded questions ask about a prominent, described object, which is usually the largest mask in the image. The attribute reward is therefore correlated with mask area, the exact quantity the saliency heuristic already maximizes. A learned policy can match saliency on this axis but has little room to pass it, which is why even the attribute-only specialist clears saliency by only 0.004. The attribute ceiling is a property of how the questions are sampled, not of the selector.

5.2.4 Why the budget controls the gradient, not just the ceiling

The collapse of the learned advantage at $K{=}4$ is easy to read as "any selection works," but the more precise statement is about the optimizer. GRPO forms its advantage by standardizing within a group of $G{=}8$ selections drawn for the same image, $\hat{A}_j = (R_j - \mu_G)/(\sigma_G + \varepsilon)$ with $\varepsilon = 10^{-6}$, and the advantage is detached. When the probes are saturated, the eight selections for an image all earn nearly the same reward, $\sigma_G \to 0$, and the standardized advantage collapses toward zero against the ε floor. That image then contributes essentially no gradient, no matter how combinatorial its action space still is. The $K{=}4$ convergence is thus not only a fact about evaluation. It is that at train time the group baseline cancels, and there is nothing left to push the policy off random. The tight budget is the same statement inverted. At $K{=}1$ the eight selections spread across the full reward range, σ_G is large, and every image delivers a strong learning signal, which is the regime where the $+0.153$ lift appears. What the budget really controls is the within-group reward variance, and that variance is the quantity GRPO converts into gradient.

The same lens covers the two places the method does not help. It ties random at $K{=}4$ because the advantage vanishes once selections stop separating in reward, and it does not robustly pass saliency on attribute because, as the previous section showed, good and bad selections earn nearly the same attribute reward there. Learned selection pays off exactly where selections of different quality are scored differently.

5.2.5 Reward sanity, a bug, and measurement robustness

A policy-gradient result is only as trustworthy as the reward that produced it, so we checked that the reward orders selection quality before reading anything into the learned numbers. The reference rows in Table 3 are strictly monotone from empty to full, and an empty trace lands at 0.014 on recognition, indistinguishable from the $1/50 = 0.020$ chance level. Reaching that point took a real fix. An earlier recognition reward scored the empty selection at a perfect 1.000. An empty trace produces an all-zero anchor embedding, the cosine similarities against the lineup are then all equal, $\arg\max$ on a uniform vector returns index zero, and the lineup happened to place the true image at index zero, so every degenerate selection scored perfectly. The fix has two parts that both survive in the code. The encoder zeroes the embedding of a genuinely empty selection so it cannot spuriously align with any reference, and the lineup is shuffled at every reward call with the true position tracked, which removes the index-zero tie-break. After the fix the empty trace returns to chance. A reward bug of this kind would have made the entire learning result meaningless, which is why we report the reference rows rather than assume them.

Two robustness checks close out the analysis. Replacing the semantic nearest-neighbor distractors with perceptual ones shifts absolute rewards by a point or two but leaves the method ranking unchanged at every budget, with the $K{=}2$ policy at 0.908 and 0.907 under the two distractor families. And every model-selection choice was made on validation alone, yet the held-out test split reproduces each ordering, so those choices transfer rather than overfit the data they were tuned on.

6 Discussion

The main claim is intentionally narrow. It is not that GRPO beats compression or heuristics in general, but that when the selector can keep only one or two units and the probes still separate good selections from bad, policy learning finds better subsets than random or a fixed rule, and the lift fades as the budget grows because that within-group reward variance shrinks. We read the continuous compressors as the instructive contrast rather than a storage-matched contest, since they top the combined reward only through the gallery artifact and rank at or below random on the honest attribute axis.

Reward design as the central lever. The action space was not the main difficulty. Plackett-Luce sampling without replacement made exact-K exploration straightforward and gave a clean log-probability, and the single-step bandit removed temporal credit assignment. The harder part was building a reward that kept headroom. Once recognition saturated at $K{=}4$, the policy had little useful signal even though the action space stayed combinatorial, so reward design, not policy expressivity, set the difficulty of the problem. The policy-gradient machinery worked once the reward carried real headroom, since headroom is what gives the group baseline a non-zero advantage to follow. The binding constraint was shaping that signal, not the optimizer that consumed it.

Scope and natural extensions. The findings are scoped to the setting they were tested in, and that setting points cleanly at how to grow the work. The budget-sweep effects are large relative to the reported intervals on a fixed split; a multi-seed repeat would tighten them further. Two properties of the current design are the most inviting to extend. The episode is single-step, so the agent commits once rather than revising a memory over a stream of images, and a sequential formulation would turn the bandit into a genuine memory-management policy. The recognition gallery holds 500 images, where a few rich units already make scenes non-confusable; scaling it, for instance to a COCO train2017 subset of roughly $118{,}000$ images, would push the saturation point past $K{=}4$ and widen the regime where selection pays off. On the attribute side the analysis already names the ceiling, so a bank with finer-grained color and spatial-relation questions would let learned selection separate from the largest-object heuristic.

7 Conclusion

We framed visual memory as a problem of choosing what to keep before the question is known, and we cast a single image as a single-step contextual bandit over a typed multi-family menu. A set-Transformer policy with a Plackett-Luce head, trained by value-free GRPO against a frozen multi-task probe reward, learns hard discrete selection under a delayed, non-differentiable signal. At matched budgets it outperforms random selection and three discrete heuristics, is the only discrete method strong on both task axes, and beats three continuous compressors on the attribute axis that is free of gallery leakage. The budget sweep gives the project its sharpest statement. The payoff of learned forgetting is set by how tight the memory is, largest when a single unit must stand in for the whole image and tapering only once the task itself runs out of headroom.

AI Tools Disclosure

I wrote the reinforcement learning components myself. These are the selection policy and its Plackett-Luce sampler (`src/agent/selector.py`), the GRPO training loop with the within-group advantage and entropy bonus (`src/agent/train_grpo.py`), the reward function over the two frozen probes (`src/agent/reward.py`), both probes, and the discrete and continuous baselines. I used Claude Code as an aid for minor debugging, including the diagnostic that surfaced the empty-selection reward artifact in Section 5, for routine data-pipeline fixes, for expanding terse code comments, and for polishing the prose here. The method design and the RL algorithms are my own.

Team Contributions

This was an individual project. Han (Harrison) Shaun Lee carried out all of it, including the problem framing, the feature pipeline, the two frozen probes, the GRPO selector and reward, the discrete and continuous baselines, the experiments, and the writing.

References

Volodymyr Mnih, Nicolas Heess, Alex Graves, and Koray Kavukcuoglu. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2014.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8:229–256, 1992.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations (ICLR)*, 2017.

Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Proceedings of the 37th Allerton Conference on Communication, Control and Computing*, 1999.

Alexander A. Alemi, Ian Fischer, Joshua V. Dillon, and Kevin Murphy. Deep variational information bottleneck. In *International Conference on Learning Representations (ICLR)*, 2017.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research (TMLR)*, 2024.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam R. Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *International Conference on Machine Learning (ICML)*, 2019.

R. L. Plackett. The analysis of permutations. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 24(2):193–202, 1975.

R. Duncan Luce. *Individual Choice Behavior: A Theoretical Analysis*. Wiley, 1959.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding, 2018.

A Implementation Details

Architectures. The policy, the attribute probe, and the recognition probe share the same front end, a per-family linear adapter to a width of 256 plus a learned family-type embedding over four families. The policy and the attribute probe use a two-layer set-Transformer encoder with four heads and a feed-forward width of 512, batch-first, with dropout 0.1. The policy head is a two-layer MLP that emits one score per candidate, with invalid positions set to $-\infty$. The attribute probe prepends a question-template embedding token and reads the yes-no logit from that token's output. The recognition probe pools with a single learned attention query and projects to a 32-dimensional L2-normalized embedding, trained with InfoNCE at temperature 0.07 using in-batch negatives. Empty selections are guarded so attention does not produce NaN and are zeroed so they cannot spuriously match in the lineup.

Hyperparameters. GRPO uses a group size of 8, a per-step batch of 32 images for an effective batch of 256, Adam at 3×10^{-4}, an entropy coefficient of 0.01, advantage standardization with $\varepsilon = 10^{-6}$, and 10 epochs, with one policy trained per budget. The attribute probe trains for 8 epochs at batch size 64, the recognition probe for 15 epochs at batch size 128. The reward uses $\alpha = \beta = 0.5$, four caption-weighted questions per image with a 0.7 probability of sampling a caption-relevant category, and a 50-way recognition lineup with semantic nearest-neighbor distractors. The global seed is 231 throughout, including the data split.

Feature pool for the continuous baselines. The continuous compressors operate on a fixed 2412-dimensional pool, the concatenation of the 768-dimensional SigLIP token, the 768-dimensional DINOv2 CLS token, the 768-dimensional mean of valid object slots, and the 108-dimensional low-level vector. Collapsing the slots to a single mean is deliberate, since preserving per-slot identity would be discrete selection in disguise, which is the contrast the continuous baselines are meant to provide. PCA is fit by truncated SVD on the train pool, the autoencoder and VIB use a 512-wide two-hidden-layer encoder, and at evaluation the VIB uses its posterior mean so retrieval is deterministic.

B Supplementary Figures

This appendix collects the two plots that visualize numbers already tabulated in the main text, the budget sweep of Table 4 (Figure 2) and the attribute–recognition frontier of Table 3 (Figure 3).

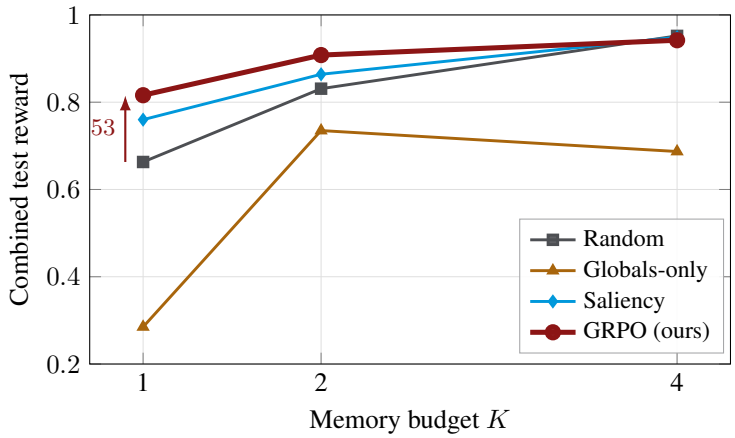

Figure 2: Combined test reward across memory budgets K. The learned policy's advantage over random selection is largest at the tightest budget, $+0.153$ at $K{=}1$, and disappears by $K{=}4$, where the recognition probe saturates and random, saliency, and GRPO converge. Globals-only stays well below the rest, since the gist tokens alone cannot identify a specific image. Same data as Table 4.

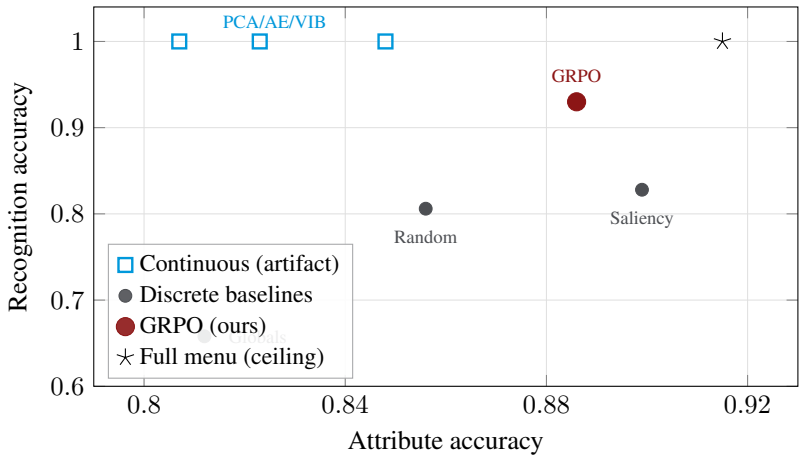

Figure 3: The attribute and recognition axes at $K{=}2$ on the test split, from the same numbers as Table 3. Among methods that share the menu interface, GRPO is the only one not dominated, sitting closest to the full-menu ceiling. Saliency reaches comparable attribute accuracy but trails on recognition, and the continuous compressors are pinned to recognition 1.000 by the shared-compressor gallery artifact while falling to the left on the honest attribute axis.