Seeing without Looking: Do Vision-Language Benchmarks Really Test Vision?

Supplementary Material

A. Implementation and Experimental Details

A.1. Benchmarks and Data Splits

We conduct our analysis on four widely used vision-language benchmarks: POPE, A-OKVQA, MME, and AMBER. In the main paper, POPE serves as the primary benchmark for controlled analysis, while A-OKVQA and MME are used to examine whether the observed phenomena extend beyond a single benchmark setting. AMBER is further included as a complementary benchmark for open-generation evaluation.

For POPE, we use a 300-example evaluation set from the official benchmark setting adopted in our study. For A-OKVQA, we evaluate on the validation split and use approximately 1.15K examples. For MME, we follow the full benchmark setting and evaluate on all examples, totaling approximately 2.37K samples. For AMBER, we follow the official evaluation protocol.

Although our study covers multiple benchmarks, the most fine-grained interventions are conducted primarily on POPE. This design is mainly motivated by the substantially higher cost of entity-level semantic manipulation, especially when image editing, quality screening, and manual verification are required. As a result, POPE serves as the main testbed for our most controlled intervention analyses.

A.2. Model Suite and Inference Setup

We evaluate a diverse set of open-source vision-language models, including LLaVA-1.5-7B, Qwen3-VL-4B, Qwen3-VL-8B, Qwen3-VL-32B, Gemma-3-12B, InternVL3-8B, and Molmo-7B-D-0924, as listed in the main paper.

Unless otherwise noted, all experiments are conducted under a unified greedy decoding setup. We use a consistent prompt format within each benchmark and evaluation condition so that the observed differences are less likely to be caused by prompt variation. Experiments are run on NVIDIA RTX 4090 GPUs using bfloat16 precision.

Our implementation is primarily based on the Hugging Face `transformers` library. For Molmo, we use `transformers==4.44.2` due to compatibility issues with newer versions in our inference pipeline.

A.3. Selection of the POPE Positive Subset

Several analyses in this work focus specifically on the positive subset of POPE, namely, examples whose ground-truth answer is *Yes*. Starting from the 300-example POPE evaluation set, we identify 149 examples with ground-truth label *Yes*.

This subset is used in analyses where affirmative support is the quantity of interest, such as examining how strongly the model continues to support the correct positive answer under different visual interventions. Restricting the analysis to ground-truth positive examples allows us to track changes in positive evidence more directly, without conflating them with shifts caused by class balance or answer priors.

For the more demanding semantic-manipulation experiments, the usable subset becomes smaller after additional filtering. In particular, we retain only those examples for which the queried entity can be reliably localized, segmented, semantically modified, and manually verified. This results in a final entity-swap subset of 117 examples.

A.4. Motivation Example Setup

Models. We conduct the experiment on multiple vision–language models, including Qwen3-VL-4B/8B/32B, LLaVA-1.5-7B, and Gemma-3-12B.

Image token dropping. For all models, we remove a subset of image tokens at the *embedding level* before they are fed into the LLM.

Concretely, given the input sequence composed of text embeddings and image embeddings,

$$[\mathbf{E}_{\text{text}}, \mathbf{E}_{\text{img}}],$$

we randomly drop a portion of image embeddings $\mathbf{E}_{\text{img}}$ with drop ratio $r \in \{0, 0.25, 0.5, 0.75\}$, and construct a new input sequence:

$$[\mathbf{E}_{\text{text}}, \mathbf{E}_{\text{img}}^{\text{kept}}].$$

Where the embeddings come from. Image embeddings are taken *after* the visual encoder and projection/merger module, i.e., the final image-token representations that are normally concatenated with text tokens before entering the LLM.

For Qwen3-VL, we explicitly extract these embeddings from the visual merger module. For other models (e.g., LLaVA-1.5, Gemma-3), we apply the same operation at the corresponding image-token embedding stage.

Implementation. We identify the positions of image tokens in the input sequence, remove a random subset of them, and directly feed the modified embeddings into the LLM using the `inputs_embeds` interface (or an equivalent mechanism).

Inference. We use greedy decoding with a maximum of 6 generated tokens. Each sample is processed independently due to variable numbers of image tokens after dropping.

A.5. Global Visual Degradation Settings

We evaluate several global visual degradation settings applied directly to the input images before feeding them into the model.

No-image. We replace the input image with a dummy black image of fixed size. The model still receives an image input, but it contains no valid visual information. Since some models do not accept inputs lacking an image, we treat the absence of visual information as "no images" to ensure alignment during testing.

Black occlusion. We apply structured black occlusion to the image. Specifically, given an image of size $H \times W$:

- $p = 0.5$: the top half of the image is set to black;
- $p = 0.75$: the top half and the bottom-left quarter are set to black;
- $p = 1.0$: the entire image is set to black.

Noise corruption. We mix the original image with random noise. Given an image $\mathbf{I}$ and random noise $\mathbf{N}$, the corrupted image is:

$$\mathbf{I}' = (1 - p)\mathbf{I} + p\mathbf{N},$$

where $p \in \{0.5, 0.75, 1.0\}$ controls the corruption strength.

Implementation. All perturbations are applied in pixel space before the standard image preprocessing pipeline. The same transformations are used across all models.

A.6. Construction of the Entity-Level Evaluation

We construct an entity-level evaluation subset to support localized visual interventions, including *BlackMask*, *BlackBox*, and *Entity Swap*.

Target extraction and localization. For each sample, we first extract the queried entity from the question using a language model. We then localize the entity in the image using Grounding DINO, followed by SAM2 to obtain a pixel-level segmentation mask.

Only samples with a single, clearly identifiable target entity are retained.

Entity-level interventions. Given the localized entity, we construct three types of interventions:

- **BlackMask.** We mask only the pixels corresponding to the segmented entity region. Specifically, pixels inside the SAM2 mask are set to black, while all other pixels remain unchanged. This removes direct visual evidence of the entity while preserving surrounding context.
- **BlackBox.** We occlude the entire bounding box of the entity. Concretely, all pixels inside the Grounding DINO bounding box are set to black. Compared to BlackMask, this removes both the entity and its immediate local context.
- **Entity Swap.** We replace the target entity with a different object using an image editing model (Gemini). The replacement is conditioned on the original image and the extracted entity, while keeping the rest of the scene unchanged. The goal is to introduce semantically incorrect but visually plausible content.

Filtering. Not all samples can be reliably processed through the full pipeline. We apply the following filtering steps:

- Remove samples with multiple queried entities;
- Remove cases where Grounding DINO fails to localize the target;
- Remove cases where SAM2 produces invalid or empty masks;
- For Entity Swap, remove cases where the editing model fails to generate a clear and consistent replacement.

Final subset. Due to failures in detection, segmentation, and image editing, the usable subset is reduced. After automatic filtering, repeated generation, and manual verification, the final entity-swap subset contains 117 samples. The same subset is used across all entity-level interventions.

A.7. Decision Margin Analysis

We compute decision margins based on the model's token-level probabilities at the first answer token.

Definition. Given the logits at the first generated token, we compute:

$$\Delta = \log P(\text{yes}) - \log P(\text{no}),$$

where $P(\text{yes})$ and $P(\text{no})$ are obtained by applying a softmax over the vocabulary.

Implementation. For each input, we construct a standard chat-style prompt and perform a forward pass without decoding. We extract the logits corresponding to the next token (i.e., the first answer token position), and compute log-probabilities via `log_softmax`.

The token IDs for `yes` and `no` are obtained from the tokenizer (including leading whitespace when required), and

the margin is computed as the difference between their log-probabilities.

Evaluation protocol. Margins are computed independently for each input under different visual conditions (e.g., original, black entity, black box, entity swap, no-image). All samples are processed without sampling, and no generation is performed beyond the first token.

Subset. For analysis, we report margin distributions on the GT=Yes subset, following the same data filtering as in the entity-level evaluation.

Visualization. We plot the distribution of margins across samples using violin and box plots, with $\Delta = 0$ as the decision boundary.

A.8. Task-Formulation Interventions

We introduce two task-level interventions by modifying the answer space and generation format.

Multiple-choice with unknown option. In the global visual degradation setting, we extend the binary answer space (yes/no) by adding an explicit `unknown` option. The prompt is kept unchanged except for the instruction to select from `yes`, `no`, or `unknown`.

Open generation. For entity-level analysis, we reformulate the task as open-ended generation. Instead of binary questions, we prompt the model to produce likely objects in the image (e.g., "List the most likely objects in the image").

Logit-based entity ranking. We evaluate whether the target entity is captured in the model's output distribution using a logit-based ranking.

For each input, we extract the logits at the first generated token and compute the probability distribution over the vocabulary. Let $p(v)$ denote the probability of token v. Given a target entity, we obtain its first token ID via the tokenizer, and compute its rank among all tokens sorted by $p(v)$.

The reciprocal rank is defined as:

$$\mathrm{RR} = \frac{1}{\mathrm{rank}},$$

and the final metric is the mean reciprocal rank (MRR) over all samples.

Implementation. We use greedy decoding with `max_new_tokens=1` and extract the first-step logits via `output_scores=True`. Ranking is performed directly on the softmax-normalized logits without generating full text sequences.

A.9. Spatial Structure and Representation Analysis Setup

We analyze spatial structure and representation properties of visual tokens extracted from the vision encoder.

Visual Tokens. For all models, we operate on patch-level visual tokens produced by the vision backbone. For ViT-based encoders, this corresponds to a fixed spatial grid (e.g., $24 \times 24 = 576$ tokens per image). The `[CLS]` token is excluded. Hidden states from all layers are collected for analysis.

Block Partition. For block-based analysis, the spatial grid is evenly divided into 4×4 non-overlapping regions. Each block contains an equal number of tokens (e.g., 6×6 tokens per block for a 24×24 grid). Token indices are mapped to 2D coordinates based on row-major ordering.

Similarity Computation. All similarity measurements are based on cosine similarity. Token features are ℓ_2-normalized before computing pairwise similarities. Intra-block similarity is computed over all token pairs within each block. Inter-block similarity is computed between block-level centroids, obtained by averaging token features within each block.

K-Means Clustering. For data-driven spatial analysis, we perform K-means clustering on token features at each layer. Tokens are clustered into $K = 16$ groups using Euclidean distance. Clustering is performed independently for each image and each layer, with a fixed number of iterations.

Effective Rank. To measure representational diversity, we compute the effective rank of the token feature matrix at each layer. Token features are first mean-centered. Singular values are obtained via SVD, and effective rank is computed as the exponential of the entropy of the normalized squared singular values.

Aggregation. All metrics are computed per image and per layer, and then averaged across the dataset. Confidence intervals are estimated using standard error across samples.

B. More Additional Results

B.1. Detailed Results on AMBER

We first report detailed results on AMBER, an open-generation benchmark that complements the closed-form evaluations in the main paper. Because the full AMBER

Table B.1. **AMBER (Generative) results.** Lower is better for CHAIR/Hal/Cog; higher is better for Cover.

Model	Condition	CHAIR ↓	Cover ↑	Hal ↓	Cog ↓
Qwen3-8B	Original	6.8	47.6	32.8	1.3
	Black (p=0.5)	11.5	42.4	45.7	1.1
	Black (p=0.75)	9.8	36.9	33.8	1.3
	Blur (p=0.5)	6.8	45.3	28.2	1.1
	Blur (p=0.75)	7.6	42.6	26.7	1.2
	No Image	76.6	13.4	95.5	14.8
Qwen3-4B	Original	7.2	52.3	35.6	0.9
	Black (p=0.5)	11.7	45.6	54.4	1.3
	Black (p=0.75)	9.7	39.3	38.1	1.3
	Blur (p=0.5)	6.3	56.1	32.0	1.5
	Blur (p=0.75)	8.4	52.8	34.5	1.9
	No Image	68.9	7.9	78.8	10.1
Qwen3-32B	Original	8.1	37.6	26.9	0.8
	Black (p=0.5)	10.8	33.5	36.7	0.9
	Black (p=0.75)	12.1	26.6	30.1	1.1
	Blur (p=0.5)	7.9	36.3	25.5	0.9
	Blur (p=0.75)	9.1	33.2	26.6	1.1
	No Image	71.1	0.4	5.1	0.8
LLaVA-1.5-7B	Original	7.4	49.4	31.7	3.7
	Black (p=0.5)	8.9	42.7	31.7	3.4
	Black (p=0.75)	11.3	36.5	33.5	3.2
	Blur (p=0.5)	8.5	46.1	30.5	2.9
	Blur (p=0.75)	10.2	44.0	31.4	3.1
	No Image	48.3	6.4	64.3	11.3
Gemma3-12B	Original	5.5	46.6	26.7	0.8
	Black (p=0.5)	8.7	38.7	34.2	0.7
	Black (p=0.75)	8.5	29.6	22.1	0.5
	Blur (p=0.5)	7.1	45.1	28.2	1.1
	Blur (p=0.75)	10.4	41.0	33.6	1.6
	No Image	66.5	0.0	99.9	0.0
InternVL3-8B	Original	5.1	52.8	31.7	1.4
	Black (p=0.5)	6.3	45.7	31.5	1.8
	Black (p=0.75)	9.1	40.5	36.1	1.8
	Blur (p=0.5)	6.3	51.7	32.0	2.1
	Blur (p=0.75)	7.1	46.8	33.7	2.0
	No Image	0.0	0.0	0.0	0.0
Molmo-7B	Original	8.4	65.4	47.0	2.6
	Black (p=0.5)	10.6	53.7	51.3	2.5
	Black (p=0.75)	11.5	43.0	44.6	2.2
	Blur (p=0.5)	8.2	63.6	41.4	2.6
	Blur (p=0.75)	10.6	59.9	48.2	3.4
	No Image	0.0	0.0	0.0	0.0

results table is relatively large and involves multiple generative evaluation metrics, we place it in the appendix for completeness.

Table B.1 presents the full AMBER results across all models and visual conditions. Overall, the results show a pattern broadly consistent with the findings in the main paper. Under moderate global degradations such as *Black* and *Blur*, most models exhibit only limited or gradual performance changes relative to the original-image condition, especially when the degradation level is not too severe. By contrast, the *No Image* condition often leads to much larger deterioration, indicating that completely removing visual input remains substantially more disruptive than weakening fine-grained visual evidence.

At the metric level, *Cover* generally decreases under stronger perturbations, while *CHAIR* and *Hal* often increase, suggesting a reduction in grounded content coverage together with a higher tendency toward hallucinated genera-

Table B.2. Detailed POPE results with expanded answer options.

Model	Condition	Acc ↑	Prec ↑	Recall ↑	F1 ↑	Neg Acc ↑	FP Rate ↓	FN Rate ↓	Unknown Rate ↓	n
LLaVA	Normal	0.9400	0.9580	0.9195	0.9384	0.9603	0.0397	0.0805	0.0000	300
	No Image	0.5033	0.0000	0.0000	0.0000	1.0000	0.0000	1.0000	0.0000	300
	Black ($p = 0.5$)	0.8833	0.9831	0.7785	0.8689	0.9868	0.0132	0.2215	0.0000	300
	Black ($p = 0.75$)	0.7833	0.9286	0.6107	0.7368	0.9536	0.0464	0.3893	0.0000	300
	Noise ($p = 0.5$)	0.9100	0.9357	0.8792	0.9066	0.9404	0.0596	0.1208	0.0000	300
	Noise ($p = 0.75$)	0.7367	0.8977	0.5302	0.6667	0.9404	0.0596	0.4698	0.0000	300
Gemma3	Normal	0.8900	0.8924	0.9592	0.9246	0.8811	0.1189	0.0408	0.0333	300
	No Image	0.5000	0.4954	0.3624	0.4186	0.6358	0.3642	0.6376	0.0000	300
	Black ($p = 0.5$)	0.7967	0.8493	0.8552	0.8522	0.8394	0.1606	0.1448	0.0600	300
	Black ($p = 0.75$)	0.7400	0.8201	0.8201	0.8201	0.8120	0.1880	0.1799	0.0933	300
	Noise ($p = 0.5$)	0.5133	0.7677	0.9835	0.8623	0.4930	0.5070	0.0165	0.3600	300
	Noise ($p = 0.75$)	0.1300	0.6000	1.0000	0.7500	0.0000	1.0000	0.0000	0.7833	300
Qwen-32B	Normal	0.9433	0.9722	0.9396	0.9556	0.9728	0.0272	0.0604	0.0133	300
	No Image	0.5033	0.0000	0.0000	0.0000	1.0000	0.0000	1.0000	0.0000	300
	Black ($p = 0.5$)	0.8533	0.9912	0.7943	0.8819	0.9931	0.0069	0.2057	0.0467	300
	Black ($p = 0.75$)	0.7967	0.9789	0.6739	0.7983	0.9865	0.0135	0.3261	0.0467	300
	Noise ($p = 0.5$)	0.8333	0.9173	0.8841	0.9004	0.9209	0.0791	0.1159	0.0767	300
	Noise ($p = 0.75$)	0.1500	0.9583	0.6765	0.7931	0.9565	0.0435	0.3235	0.8100	300
Qwen3-8B	Normal	0.9700	0.9861	0.9530	0.9693	0.9868	0.0132	0.0470	0.0000	300
	No Image	0.5033	0.0000	0.0000	0.0000	1.0000	0.0000	1.0000	0.0000	300
	Black ($p = 0.5$)	0.8633	0.9821	0.7383	0.8429	0.9868	0.0132	0.2617	0.0000	300
	Black ($p = 0.75$)	0.7700	0.9348	0.5772	0.7137	0.9603	0.0397	0.4228	0.0000	300
	Noise ($p = 0.5$)	0.9067	0.9690	0.8389	0.8993	0.9735	0.0265	0.1611	0.0000	300
	Noise ($p = 0.75$)	0.6967	0.8625	0.4631	0.6026	0.9272	0.0728	0.5369	0.0000	300
Qwen3-4B	Normal	0.9567	0.9857	0.9262	0.9550	0.9868	0.0132	0.0738	0.0000	300
	No Image	0.5033	0.0000	0.0000	0.0000	1.0000	0.0000	1.0000	0.0000	300
	Black ($p = 0.5$)	0.8667	1.0000	0.7365	0.8482	1.0000	0.0000	0.2635	0.0033	300
	Black ($p = 0.75$)	0.7800	0.9560	0.5878	0.7280	0.9735	0.0265	0.4122	0.0033	300
	Noise ($p = 0.5$)	0.8933	0.9535	0.8255	0.8849	0.9603	0.0397	0.1745	0.0000	300
	Noise ($p = 0.75$)	0.6167	0.8305	0.3379	0.4804	0.9315	0.0685	0.6621	0.0300	300
InternVL	Normal	0.9767	0.9797	0.9732	0.9764	0.9801	0.0199	0.0268	0.0000	300
	No Image	0.4667	0.0000	0.0000	0.0000	1.0000	0.0000	1.0000	0.1067	300
	Black ($p = 0.5$)	0.8633	0.9821	0.7432	0.8462	0.9868	0.0132	0.2568	0.0033	300
	Black ($p = 0.75$)	0.8100	0.9694	0.6419	0.7724	0.9801	0.0199	0.3581	0.0033	300
	Noise ($p = 0.5$)	0.9333	0.9574	0.9060	0.9310	0.9603	0.0397	0.0940	0.0000	300
	Noise ($p = 0.75$)	0.7200	0.7982	0.6107	0.6920	0.8446	0.1554	0.3893	0.0100	300
Molmo	Normal	0.9400	0.9524	0.9396	0.9459	0.9530	0.0470	0.0604	0.0067	300
	No Image	0.5067	1.0000	0.0067	0.0133	1.0000	0.0000	0.9933	0.0000	300
	Black ($p = 0.5$)	0.8567	0.9573	0.7568	0.8453	0.9667	0.0333	0.2432	0.0067	300
	Black ($p = 0.75$)	0.7667	0.9100	0.6190	0.7368	0.9392	0.0608	0.3810	0.0167	300
	Noise ($p = 0.5$)	0.9067	0.9362	0.8919	0.9135	0.9396	0.0604	0.1081	0.0100	300
	Noise ($p = 0.75$)	0.6933	0.8991	0.7538	0.8201	0.9091	0.0909	0.2462	0.1633	300

tion. The overall pattern therefore supports the main observation of this work: partial degradation of visual evidence does not always cause catastrophic behavioral collapse, but complete removal of the image typically does.

We also note that model behavior on AMBER is more heterogeneous than on closed-form benchmarks. In particular, some models show extreme outputs under the *No Image* condition, which likely reflects model-specific generation behavior or evaluator interactions in open-ended settings. For this reason, we treat AMBER primarily as complementary evidence rather than the main basis for our controlled analysis.

B.2. Detailed Results with Expanded Answer Options

In the main paper, for the setting with expanded answer options, we report only the *Unknown Rate* as the primary quantity of interest. Here, we provide the full results for completeness, including accuracy, precision, recall, F1, negative accuracy, false-positive rate, and false-negative

Table B.3. **Decision-level support for affirmative predictions under visual interventions** (ground truth = *Yes*). We report mean and median logit margins, mean and median affirmative probabilities (p_{yes}), and confidence-shift statistics relative to the *Original* condition. Here, $\mathbb{E}[\delta]$ denotes the expected reduction in affirmative support.

Condition	Margin ↑		Probability (p_{yes}) ↑		Confidence Shifts		
	Mean	Median	Mean	Median	$\mathbb{E}[\delta]$ ↑	$\Pr(\delta > 0)$ ↑	$\Pr(\delta > 1)$ ↑
Original	4.5625	5.1250	0.9258	0.9922	–	–	–
Black Mask	2.2031	2.4375	0.7695	0.9180	2.3594	0.8591	0.6242
Black Box	1.1406	0.8125	0.6055	0.6914	3.4219	0.8792	0.6711
No Image	-0.1436	0.0000	0.4727	0.5000	4.7188	0.9195	0.8926
Entity Swap	1.0469	0.9375	0.5977	0.7188	3.5312	0.8658	0.6779

rate.

Table B.2 shows the detailed results across all models and visual conditions on POPE when the model is allowed to answer *Yes*, *No*, or *Unknown*. Overall, the results suggest that introducing an *Unknown* option does not fundamentally change the main behavioral pattern observed in the paper. For most models, performance under moderate visual degradation remains relatively stable, whereas the *No Image* condition leads to a much larger drop.

The detailed metrics also help clarify the role of the *Unknown* option. For several models, the increase in *Unknown Rate* under stronger perturbations is accompanied by changes in both recall and false-negative rate, indicating that the model becomes more likely to abstain or avoid committing to positive predictions when visual evidence is severely weakened. At the same time, this behavior is not uniform across models: some models rarely choose *Unknown*, while others use it much more aggressively under severe degradation.

These results therefore support the observation in the main text that the availability of an explicit *Unknown* option does not by itself resolve the benchmark–grounding mismatch. Instead, it mainly reveals model-specific differences in uncertainty expression under degraded visual input.

B.3. Detailed Statistics for Decision-Level Support

In the main paper, we visualize decision-level support for affirmative predictions using box plots. To complement that distributional view, we report here the corresponding summary statistics on the POPE positive subset ($n = 149$, ground truth = *Yes*).

Table B.3 summarizes the model's support for the affirmative answer under different visual interventions in terms of both logit margin and affirmative probability p_{yes}. In addition to the mean and median values, we also report confidence-shift statistics relative to the *Original* condition. Specifically, $\mathbb{E}[\delta]$ measures the expected reduction in affirmative support, while $\Pr(\delta > 0)$ and $\Pr(\delta > 1)$ quantify how often the intervention lowers support, and how often the drop exceeds a larger threshold.

The results are consistent with the distributional patterns shown in the main text. Relative to the *Original* condition, all interventions reduce affirmative support on the positive subset. The effect is strongest under *No Image*, where both the mean margin and mean affirmative probability fall close to the decision boundary, indicating a near-complete loss of positive support. Localized interventions such as *Black Mask*, *Black Box*, and *Entity Swap* also substantially reduce support, even though they do not remove the entire image.

These results further support our main claim that benchmark predictions can remain behaviorally stable even when the internal support for the correct affirmative answer has already weakened considerably. In other words, decision correctness alone may understate the extent to which fine-grained visual evidence has been lost or degraded.

B.4. Qualitative Case Study of Spatial Representation Evolution

To provide an intuitive illustration of the representational trends discussed in the main paper, we present a qualitative case study of spatial token clustering across layers.

Figure B.1 visualizes the k-means clustering assignments of spatial tokens at different layers of the visual encoder. Each colored cell corresponds to one spatial token, and colors indicate cluster membership ($k = 16$). The spatial layout of tokens follows the original patch grid of the image.

In early layers (e.g., Layers 1 and 4), the clusters exhibit relatively coherent spatial regions, with contiguous patches often belonging to the same cluster. This suggests that local visual structure and spatial organization are still preserved in the representation.

As depth increases, however, the clustering structure becomes progressively more fragmented. In later layers (e.g., Layers 17, 21, and 23), cluster assignments appear increasingly mixed and spatially irregular. Neighboring patches are less likely to share the same cluster, indicating that the representations of different spatial regions become more intertwined.

This qualitative pattern aligns with the quantitative results reported in the main paper, where measures such as inter-block similarity and effective rank indicate a gradual

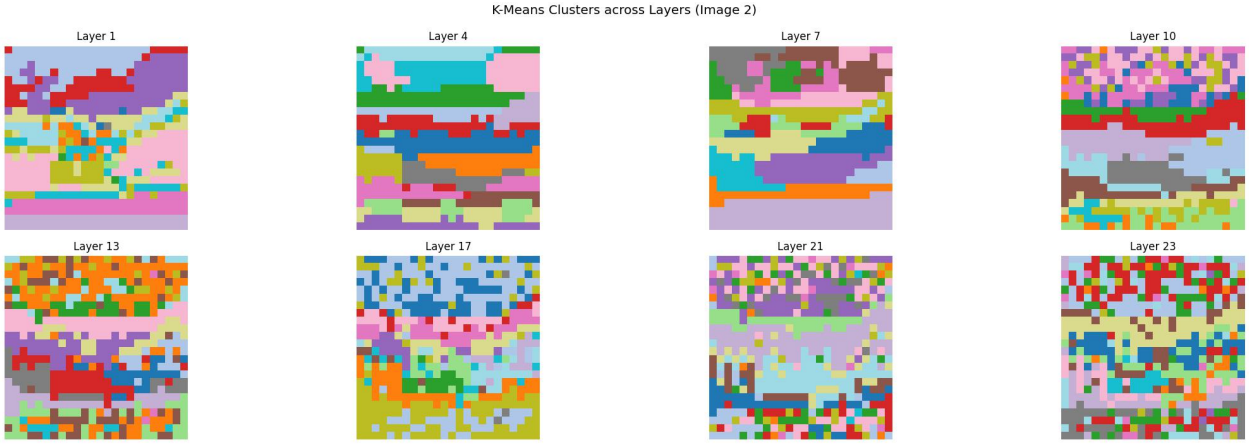

Figure B.1. Qualitative visualization of region-level clustering structures and spatial token evolution.

loss of spatial distinctiveness across layers. The visualization therefore provides an intuitive illustration of how spatial token structure evolves during visual encoding.