Stop Mixing Things Up! BISCUIT Teaches Vision-Language Models to Learn New Concepts from Images on the Spot

Jiahua Bao[1,2,3], **Siyao Cheng**[1,2,3*], **Jiaxing Du**[1], **Yuhang Jia**[1], **Boyang Niu**[1], **Zeming Lang**[1]
Changjiang He[1], **Hao Zhang**[1,2,3], **Jie Liu**[1,2,3]

[1]Research Center of Ubiquitous Computing and Intelligent Systems, Harbin Institute of Technology, China
[2]National Key Laboratory of Smart Farming Technology and Systems, China
[3]China Mobile 5G Institute, China
jhbao@stu.hit.edu.cn, csy@hit.edu.cn

Abstract

Vision-Language Models (VLMs) have achieved impressive performance across various tasks, but often struggle to apply newly introduced visual concepts during inference. A common failure pattern is what we call **Mixing Things Up**: VLMs frequently confuse concept names, resulting in vague descriptions and failure to ground the concept correctly. Existing approaches mainly address person-related concepts through text prompts or tokenizer modifications. However, VLMs still miss or misinterpret untrained visual concepts, underscoring the need to learn new concepts directly from visual input, without relying on prior textual injection. To overcome these limitations, we propose **BISCUIT** (**B**asis-aligned **I**nference through **S**tructured **C**oncept **U**nification and **I**dentification-aware **T**uning), a two-step training method. Step I proposes a dual-stream structure-aware vision encoder that fuses RGB and edge-based embeddings within a shared basis space to enhance concept recognition. Step II enhances generation quality through identification-aware tuning, which encourages alignment between the generated text and the newly introduced visual concepts. Existing methods mainly focus on person concepts and lack comprehensive evaluation across diverse visual categories. We further propose a benchmark BiscuitVQA to evaluate VLMs performance on recognizing and applying novel image-introduced concepts across diverse concept types and task types, including real people, cartoons, animals, and symbolic content. We apply BISCUIT to LLaVA-1.5 and Qwen2.5-VL, achieving competitive results among open-source models and narrowing the gap to Gemini-2.5 and GPT-4o. Interestingly, our BISCUIT maintains strong generalization, showing minimal degradation on other downstream tasks.

Details — https://github.com/Samsara-1999/BISCUIT

Introduction

Vision-Language Models (VLMs) have achieved impressive performance on tasks like image captioning and visual question answering, largely due to training on massive image-text datasets(Jin et al. 2024; Laurençon et al. 2024; Bao et al. 2025a). However, no matter how broad the training data is, it is impossible to cover all possible concepts in advance. In real-world scenarios, VLMs are often required to identify new concepts during inference and apply them correctly in output text(Li, Ma, and Peng 2024). However, many concepts in real images such as who is who in a photo or the meaning of a symbol in a chart, which are hard to explain clearly using text alone(Huang et al. 2024a; Bao et al. 2025b). In these cases, introducing the concept directly through the input image is often a more effective choice. Image-introduced concepts are concepts that appear only in the image and require visual understanding without explicit textual description. Existing VLMs often fail to handle image-introduced concepts(Rudman et al. 2025). As shown in Figure 1, when faced with unfamiliar concept names that do not appear during training, the model struggles to associate them with the correct textual descriptions on the spot. We refer to this failure as **Mixing Things Up**. This raises a fundamental question: if we cannot pre-train VLMs on every possible concept or encode all concept names in the tokenizer, **how can we teach VLMs to learn new visual concepts on the spot, directly from images during inference?**

Figure 1: We present a qualitative example comparing the base model and our BISCUIT-trained model. In Question (Q), we randomly rename the character `Caroline` as `Tom` (red box) to ensure the concept is unseen. The base model mismatches the concept, while our model correctly answers by grounding to the provided relationship table. Red denotes errors and green indicates correct concept mapping.

*Corresponding author.

To overcome this limitation, we focus on two essential abilities: (1) **recognizing** when a new concept appears in the image, and (2) correctly **aligning** them with output text. We propose **BISCUIT** (**B**asis-aligned **I**nference through **S**tructured **C**oncept **U**nification and **I**dentification-aware **T**uning), a two-step training method designed to help VLMs recognize and apply novel image-introduced concepts, as shown in Figure 2. Unlike prior approaches that inject concept labels or rely on predefined tokens during fine-tuning(Chen et al. 2025), BISCUIT requires no explicit injection in tokenizer, instead teaching the model to learn from raw visual input alone. To our knowledge, this is the first work that enables VLMs to acquire novel concepts purely through input images, without textual injection in advance.

Specifically, in **Step I**, we introduce a dual-stream encoder to enhance concept recognition. The RGB image is processed by the pre-trained vision encoder, while the edge image generated by the Canny(Ding and Goshtasby 2001) operator highlights object boundaries and concept locations, serving as input to newly added attention modules. These new modules are initialized using decomposition from the original encoder. We fuse both streams through a basis-aligned strategy to form a structure-aware visual concept encoder. In **Step II**, we improve the performance of generated text by fine-tuning with enriched prompts that describe the visual attributes of introduced concepts. To align text with visual input, we propose an identification-aware loss combining a global contrastive term and a token-level continuity penalty, encouraging precise and consistent outputs.

While most existing methods focus on person-related concepts, they often overlook a broader range of image-introduced concepts, such as cartoon characters, animals, and symbolic elements in charts(Zhang et al. 2024; Keraghel, Morbieu, and Nadif 2024). These concepts frequently appear in real-world scenarios but are difficult to collect and organize(Wu et al. 2024b; Yang et al. 2024). To bridge this gap, we construct the BiscuitVQA benchmark to evaluate model performance in recognizing and applying such concepts. The benchmark includes four task types: Brief Answer, Detailed Description, Chain of Thought, and Single Choice, covering diverse visual domains. Experiments show that BISCUIT significantly improves model performance across these tasks. Applied to both LLaVA-1.5 and Qwen2.5-VL, our method outperforms strong open-source baselines and even achieves results competitive with Gemini-2.5 and GPT-4o on several tasks. Moreover, BISCUIT demonstrates strong generalization by preserving performance on unrelated downstream tasks better than other methods.

Our contributions are as follows:

- We propose BISCUIT, a two-step training method that improves the model's ability to recognize image-introduced concepts by integrating structural cues from edge images, and enhances generation quality through identification-aware tuning with a contrastive penalty.
- We construct a benchmark for evaluating image-introduced concept learning, covering four task types: Brief Answer, Detailed Description, Chain of Thought, and Single Choice, across diverse domains including real people, cartoons, animals, and symbolic charts.
- BISCUIT achieves competitive performance on image-introduced concept tasks compared with other methods. It even performs competitively with Gemini-2.5 and GPT-4o on several tasks, and shows strong generalization by preserving performance on unrelated benchmarks better than other methods.

Related Works

Vision-Language Model Architectures

Existing VLMs employ a modular design with a vision encoder, a modality merger (connector), and a large language model (LLM) as text decoder. The vision encoder, usually a vision transformer (ViT), generates image embeddings that are merged with text tokens and passed to the LLM(Zhang, Huang et al. 2024; Li et al. 2025). Qwen-VL(Wang et al. 2024) and LLaVA(Liu et al. 2024b) are representative models based on this design. In most cases, the vision encoder contains significantly fewer parameters than the LLM(Fini et al. 2025). InternVL(Chen et al. 2024) points out that this imbalance may limit the overall performance of the model, especially when dealing with unseen inputs. However, due to the complexity of the VLMs architecture, fine-tuning all three modules for specific tasks often leads to significant drops in generalization ability(Huang et al. 2024b; Ge et al. 2025; Liang et al. 2024). Therefore, a key challenge is how to improve task-specific performance while minimizing the loss of generalization across other diverse tasks(Wu et al. 2024a; Wang et al. 2023b).

Visual Concept Learning

Visual concept learning in VLMs requires models to recognize and apply new concepts, such as unseen characters or symbolic elements, through generated text(Bouritsas et al. 2018; Lee et al. 2023). Recent methods like Yo'LLaVA(Nguyen et al. 2024) and MC-LLaVA(An et al. 2024) improve performance by modifying tokenizers and using paired prompts, but they rely heavily on memorization and struggle to learn from structural cues in the image. These approaches also focus mostly on person concepts, overlooking broader categories like cartoons, animals, and symbolic content. Instead of injecting each concept, a more generalizable solution is to let the model learn new image-introduced concepts on the spot from the image during inference.

Methods

BISCUIT Overview

We propose BISCUIT, a two-step training method designed to help VLMs learn and apply new visual concepts introduced directly through images. The method focuses on two core capabilities: recognizing unfamiliar concepts and expressing them accurately in output text. As shown in Figure 2, Step I enhances the model's ability to recognize new concepts by introducing a structure-aware encoder that incorporates both RGB and edge-based visual cues, while Step

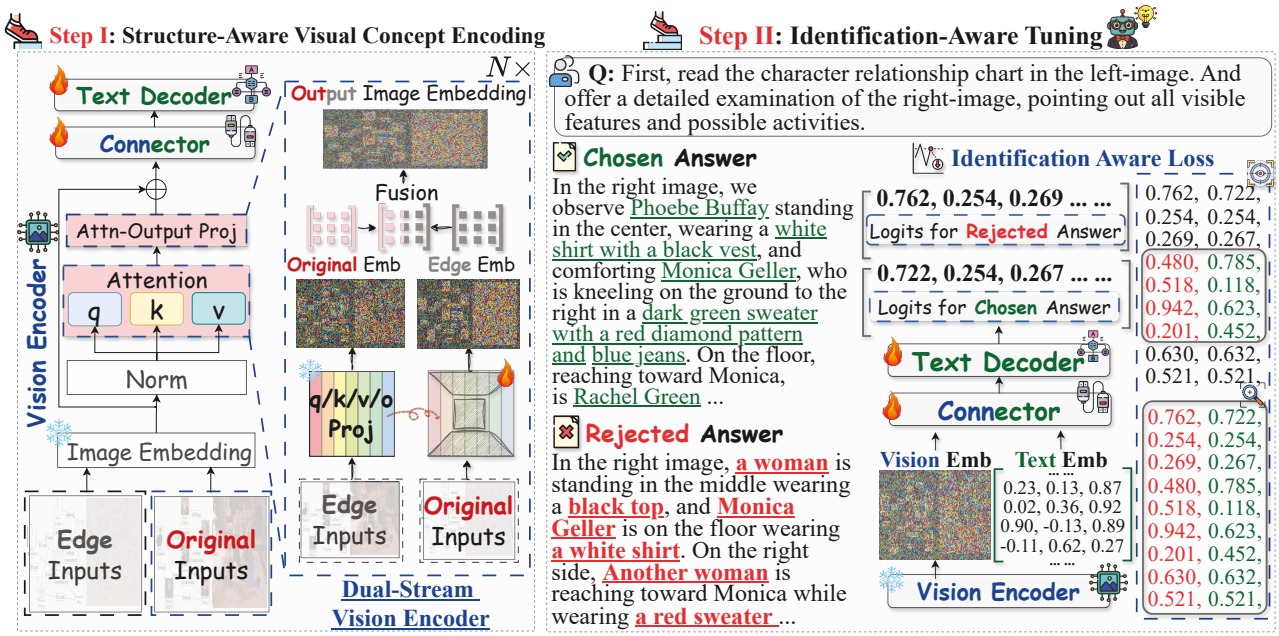


Figure 2: BISCUIT is a two-step training method for teaching VLMs to recognize and apply image-introduced concepts. **Preparation** constructs a multi-types dataset, where each image is paired with QA pairs. Step II are expanded from Step I to emphasize more details of the image. In **Step I**, a structure-aware dual-stream encoder processes both RGB and edge inputs. The edge stream is initialized by decomposing the pre-trained vision encoder's attention. Embeddings from both streams are fused in a shared basis space and passed to the text decoder via a connector. In **Step II**, we optimize the text decoder with identification-aware tuning, which combines a contrastive loss that focuses on global token-differences between candidate outputs, and a continuity loss that captures local token-level inconsistencies, encouraging precise and concept-aligned generation.

II enables the text decoder to generate consistent and precise responses based on the image-introduced concepts.

Step I: Structure-Aware Visual Concept Encoding

In this step, we aim to enable the VLMs to recognize novel visual concepts by capturing input image's structural patterns. Specifically, we design the Structure-Aware Visual Concept Encoder, which consists of two components: a frozen RGB stream based on the original vision encoder and a set of learnable edge stream attention modules that receives input edge images. These edge images generated by the Canny operator, which we find consistently highlights meaningful structural contours such as concept boundaries. These edge stream attention modules are initialized from the original vision encoder using a singular value decomposition(Abdi 2007), a SVD-based initialization strategy. SVD-based initialization naturally aligns with the sparse nature of edge images. It decomposes the original weights into three low-rank weight matrices, enabling efficient transformation by compressing, projecting, and reconstructing salient features.

However, the two streams generate distinct image embeddings: one from the RGB image and another from the edge image. Since existing VLM architectures accept only a single image embedding as input(Bordes et al. 2024), directly concatenating both would increase image token length and training cost. **This raises a key challenge: how to fuse the two embeddings effectively**. To address this, we propose a basis-aligned fusion strategy that projects both embeddings into a shared basis space and combines them adaptively. Specifically, given the generated image embeddings from the RGB-stream $\mathbf{E}_{\text{rgb}} \in \mathbb{R}^{B\times N\times D}$ and the edge-stream $\mathbf{E}_{\text{edge}} \in \mathbb{R}^{B\times N\times D}$, where B is the batch size, N is the number of image tokens, and D is the embedding dimension, we perform orthogonal projection into a shared basis space using QR decomposition(Xu et al. 2024).

We combine both by summing their basis matrices extracted via QR decomposition. This decomposition factorizes a embedding into an orthogonal basis matrix and an upper triangular projection matrix. This operation merges the subspace structures of the RGB and edge features, allowing the fused space to preserve salient directions from both inputs. We then apply another QR decomposition to ensure that the fused basis $\mathbf{Q}_{\text{fused}}$ is orthogonal. This step is essential for stabilizing the subsequent projection process, as orthogonal bases avoid redundant directions and preserve numerical stability. In our experiments, we observe that omitting this step results in significantly higher variance and instability in training loss, confirming the necessity of maintaining a well-conditioned fused subspace. Specifically, let $\mathcal{Q}(\cdot)$ denote the function that extracts the basis matrix from the input via QR decomposition. We define the fused basis $\mathbf{Q}_{\text{fused}}$ as:

$$\mathbf{Q}_{\text{fused}} = \mathcal{Q}\left(\mathcal{Q}(\mathbf{E}_{\text{rgb}}) + \mathcal{Q}(\mathbf{E}_{\text{edge}})\right) \quad (1)$$

We then project $\mathbf{E}_{\text{rgb}}$ and $\mathbf{E}_{\text{edge}}$ into the shared basis space: $\tilde{\mathbf{E}}_{\text{rgb}} = \mathbf{Q}_{\text{fused}} \cdot \left(\mathbf{Q}_{\text{fused}}^{\top} \cdot \mathbf{E}_{\text{rgb}}\right), \tilde{\mathbf{E}}_{\text{edge}} = \mathbf{Q}_{\text{fused}} \cdot \left(\mathbf{Q}_{\text{fused}}^{\top} \cdot \mathbf{E}_{\text{edge}}\right)$. We compute cosine similarity(Xia, Zhang, and Li 2015) between the projected embeddings to measure their informational difference. A high similarity indicates low divergence between the RGB and edge embeddings, allowing the model to preserve the original visual stream's semantics. Conversely, a lower similarity highlights the contribution of complementary structural cues from the edge stream. This adaptive weighting helps balance new structural signals with existing visual features, ensuring effective fusion while maintaining model stability: $w_{\text{rgb}} = \cos\left(\tilde{\mathbf{E}}_{\text{rgb}}, \tilde{\mathbf{E}}_{\text{edge}}\right)$, $w_{\text{edge}} = 1 - w_{\text{rgb}}$. These weights w_{rgb} and w_{edge} are normalized with a softmax: $[w_{\text{rgb}}, w_{\text{edge}}] = \text{softmax}([w_{\text{rgb}}, w_{\text{edge}}])$. And the output fused embedding is a weighted sum of the projected features: $\mathbf{E}_{\text{fused}} = w_{\text{rgb}} \cdot \tilde{\mathbf{E}}_{\text{rgb}} + w_{\text{edge}} \cdot \tilde{\mathbf{E}}_{\text{edge}}$. Finally, the fused embedding is added as a residual to the RGB stream and passed through the original connector before being fed into the text decoder. During Step I, we adopt the Cross-Entropy Loss(Krizhevsky, Sutskever, and Hinton 2012) and freeze all original visual attention layers to preserve pre-trained knowledge. Only the newly introduced edge-stream modules, the connector, and the text decoder are updated.

Step II: Identification-Aware Tuning

While Step I helps the model recognize new visual concepts by capturing structural patterns, Step II focuses on aligning text generation with those concepts accurately. We refer to this stage as identification-aware tuning, which fine-tunes the connector and text decoder using enriched prompts and a novel loss function.

We design structured prompts that link visual concepts with textual descriptions. For example, we use fine-grained attributes like **"Jack is the man in a green jacket"** and questions that require multi-step reasoning (**"Who is Jack, and what is he doing?"**). These prompts guide the model to produce more specific and grounded outputs.

More importantly, we propose the identification-aware loss to further enforce consistency between the model's output and the input image. This loss encourages the model to prefer precise, concept-aligned output while penalizing vague or overgeneralized ones. Formally, inspired by direct preference optimization (DPO)(Rafailov et al. 2023), we define a contrastive loss that compares two candidate output answers for the same input x (image + prompt): the preferred y^+ and the less preferred y^-. The loss encourages higher likelihood for y^+ over y^-:

$$\mathcal{L}_{\text{contrastive}} = -\log \sigma\left(\log \pi(y^+ \mid x) - \log \pi(y^- \mid x)\right) \quad (2)$$

where σ is the sigmoid function, $\pi(y \mid x)$ denotes the model's likelihood of generating text y given input x.

However, overall preference is not sufficient to ensure precise generation(Yan et al. 2024; Wang et al. 2023a). To enforce token-level concept-alignment, we propose a continuity loss that highlights regions where the model hesitates or drifts away from the visual input. For instance, the correct output text might be **"Jack is the man in a green jacket sitting on the left and reading a book"**. However, the model may generate **"Tom is the man in a green jacket sitting on the left and reading a book"**, which superficially aligns with the image but fails to identify the correct concept. Our continuity loss captures such token-level inconsistencies by focusing on words like `Jack` and `Tom` that reflect semantic mismatches. For each token position t in the output, we compute the absolute log-probability difference between the chosen and rejected outputs:

$$d_t = \left|\log \pi(y_t^+ \mid x) - \log \pi(y_t^- \mid x)\right| \quad (3)$$

we then transform this difference into a soft confidence mask: $\delta_t = \sigma \cdot (d_t - \tau)$, where τ is a threshold that determines whether two candidate output answers differ significantly at a given token position. Let $\mathbf{m}_t$ be a binary mask that filters out padding tokens. The final weight for each token becomes: $\tilde{\delta}_t = \delta_t \cdot \mathbf{m}_t$. We define the continuity loss as the average product of adjacent uncertain tokens, which can penalizes long vague spans:

$$\mathcal{L}_{\text{continuity}} = \lambda \cdot \frac{1}{T-1} \sum_{t=0}^{T-1} \tilde{\delta}_t \cdot \tilde{\delta}_{t+1} \quad (4)$$

where T is the output length and λ is trainable. This loss penalizes vague spans in the output and encourages the model to be more decisive on critical tokens based on image-introduced concepts.

In summary, the contrastive loss captures holistic preference between candidate outputs y^- and y^+, while the continuity loss focuses on localized inconsistencies. The final identification-aware tuning objective is defined as: $\mathcal{L} = \mathcal{L}_{\text{contrastive}} + \mathcal{L}_{\text{continuity}}$.

Experiments and Analyses

Experimental Setup

We evaluate BISCUIT on our BiscuitVQA benchmark for image-introduced concept learning, comparing it with both open-source and closed-source models such as GPT-4o(Hurst et al. 2024) and Gemini-2.5(Comanici et al. 2025). We adopt two complementary metrics: S_{ANLS} for surface-level accuracy and S_{GPT} scoring for semantic alignment. To assess generalization, we also test BISCUIT on 11 standard VLM benchmarks and apply it to multiple base models, including LLaVA-1.5(Liu et al. 2024a) and Qwen2.5-VL(Bai et al. 2025) across different scales. All models are trained with three runs and the average result is reported. Experiments are conducted on NVIDIA A800 GPUs.

Dataset Summary: Existing datasets for concept learning have two main limitations. First, they narrowly define concepts as real-person names, focusing on character matching while ignoring broader visual concepts like animals, cartoons, or symbols(Patel et al. 2024). Second, there is no standardized benchmark for evaluating how well VLMs understand and apply image-introduced concepts. To address this, we construct a new dataset that expands beyond human identity and supports consistent evaluation across diverse domains and reasoning types.

Methods	Concept Types								Task Types								Concepts Count x						Overall↑	
	Human↑		Cartoon↑		Animal↑		Symbol↑		Brief↑		Detail↑		Choice↑		COT↑		$x=0$↑		$1 \leq x \leq 3$↑		$x>3$↑			
	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}
Qwen2.5-VL-3B																								
Base Model	28.2	52.1	29.3	53.2	26.4	49.2	21.0	48.0	20.4	56.5	25.0	50.3	47.9	62.6	23.3	54.2	29.3	67.2	23.9	61.3	29.2	59.8	303.9	614.3
MyVLM	30.0	58.3	**34.7**	**64.2**	32.9	55.7	26.3	56.7	**27.3**	57.9	30.4	56.5	59.3	73.3	36.8	65.6	30.9	67.6	28.1	64.7	**41.1**	64.2	377.8	684.6
Yo'LLaVA	29.8	61.7	32.8	61.4	31.8	**63.3**	28.8	52.8	25.0	52.1	27.9	60.0	53.3	**78.3**	40.5	66.2	**31.7**	**72.7**	27.9	61.0	35.6	63.8	364.9	693.2
MC-LLaVA	**32.6**	58.1	34.0	60.6	31.8	61.3	28.1	**59.3**	24.0	**61.4**	**31.8**	62.3	**70.0**	67.8	**44.4**	**71.3**	29.1	70.0	**30.8**	66.2	40.1	**68.5**	**396.8**	**706.8**
Yo'Chameleon	30.1	**63.5**	34.0	56.1	**35.0**	58.8	**29.0**	45.5	23.5	56.9	29.3	**64.9**	67.8	72.4	43.8	67.5	30.3	65.5	28.5	**68.4**	41.0	63.0	392.1	682.4
BISCUIT(ours)	**33.1**	**67.2**	**36.3**	**63.5**	**35.5**	**62.7**	**29.2**	**60.5**	**29.4**	**63.1**	**33.8**	**68.6**	**69.6**	**78.5**	**46.0**	**70.3**	**32.0**	**74.4**	**31.7**	**67.3**	**41.7**	**72.7**	**418.3**	**748.7**
Qwen2.5-VL-7B																								
Base Model	31.5	54.5	32.5	57.0	28.5	55.8	23.1	48.7	19.1	55.0	28.1	54.6	58.7	70.6	26.2	69.3	33.8	72.4	24.6	70.7	33.8	68.2	339.9	676.6
MyVLM	37.6	60.9	38.7	53.1	38.1	**67.7**	30.2	63.8	26.7	50.8	29.1	**65.5**	**72.6**	**84.8**	58.0	75.2	34.6	79.6	30.3	**75.7**	43.7	71.4	439.6	748.3
Yo'LLaVA	34.3	65.0	40.8	57.5	41.5	62.2	28.6	61.5	26.7	55.7	27.1	61.5	64.9	65.3	50.9	70.0	31.3	73.5	31.3	71.4	41.2	**73.7**	418.5	717.5
MC-LLaVA	38.4	**71.7**	41.3	58.8	**43.3**	64.3	30.4	61.3	**30.1**	61.6	**33.8**	63.4	66.3	77.0	57.2	72.3	35.7	**81.4**	36.8	69.0	**46.1**	70.2	459.2	750.7
Yo'Chameleon	**38.6**	67.7	**42.0**	**61.6**	41.2	65.6	**30.8**	**65.8**	28.6	**61.7**	32.0	58.4	69.9	**82.9**	**59.6**	**77.3**	**35.9**	80.3	**38.6**	74.1	45.4	72.7	**462.4**	**768.2**
BISCUIT(ours)	**40.1**	**71.4**	**42.3**	**65.5**	**42.9**	**68.6**	**31.4**	**65.1**	**30.2**	**68.0**	**35.5**	**71.7**	**72.7**	78.7	**61.6**	**76.7**	**37.9**	**82.6**	**40.2**	**76.0**	**49.0**	**75.5**	**483.9**	**799.6**
LLaVA-1.5-7B																								
Base Model	21.9	53.1	23.4	55.5	23.0	54.2	21.6	57.3	20.9	61.1	22.1	64.7	41.7	59.7	23.7	76.2	28.5	72.5	23.1	64.1	24.1	68.1	274.0	686.5
MyVLM	18.0	**63.1**	26.5	62.2	23.2	**64.6**	22.7	61.8	19.9	67.8	23.4	**73.3**	55.5	63.9	**26.2**	**82.8**	27.6	82.8	**23.1**	**69.3**	23.8	73.1	289.9	**764.9**
Yo'LLaVA	20.0	61.1	27.2	62.4	22.8	59.1	21.8	**69.8**	21.0	69.0	**26.5**	68.1	53.2	60.7	23.2	77.7	24.0	**84.0**	19.3	62.9	24.6	76.2	283.4	750.9
MC-LLaVA	**23.5**	58.4	**28.9**	59.7	**26.8**	58.2	**24.4**	**68.8**	**24.6**	62.6	21.4	72.5	**59.3**	**66.9**	25.5	82.8	28.6	78.0	22.2	66.7	25.3	**80.2**	**310.7**	754.6
Yo'Chameleon	22.0	56.4	27.2	**63.9**	25.8	57.1	22.4	63.0	19.7	**70.3**	24.1	71.4	51.1	64.2	25.9	80.9	**30.2**	**84.5**	22.9	64.1	**26.3**	76.5	297.6	752.2
BISCUIT(ours)	**23.9**	**63.9**	**29.6**	**62.4**	**27.8**	**60.0**	**24.0**	68.5	**24.4**	**72.4**	**27.2**	**72.7**	**59.4**	**68.5**	**26.7**	**84.6**	**30.6**	83.6	**25.1**	**68.1**	**26.4**	**78.6**	**325.2**	**783.3**
LLaVA-1.5-13B																								
Base Model	24.7	56.0	25.8	54.4	24.6	59.7	23.5	62.5	21.0	68.8	24.6	70.6	44.7	71.2	21.9	78.3	31.0	80.5	24.0	65.8	27.9	71.7	293.6	739.3
MyVLM	31.8	**64.4**	33.6	56.2	32.6	**64.3**	34.2	65.3	29.5	67.5	30.2	71.9	68.5	**78.9**	21.4	83.2	37.4	82.8	**34.6**	66.6	34.0	75.8	387.8	**776.9**
Yo'LLaVA	33.1	60.1	32.7	**60.4**	**36.7**	**64.6**	32.2	60.3	27.6	70.3	31.1	**73.0**	71.1	78.2	22.1	74.0	35.5	79.1	25.1	57.8	36.5	72.8	383.6	750.6
MC-LLaVA	**39.1**	59.9	**41.0**	48.9	34.1	60.6	**37.9**	**65.3**	26.0	**73.1**	29.1	70.2	**82.3**	70.0	**24.8**	85.5	31.6	**84.2**	32.6	62.9	35.9	74.6	414.3	755.1
Yo'Chameleon	36.5	57.2	38.8	54.8	33.6	58.8	33.3	64.8	**30.0**	69.7	**32.9**	72.7	**81.0**	72.0	21.0	**86.3**	**38.1**	78.4	30.4	**69.0**	**38.9**	**77.3**	**414.4**	760.9
BISCUIT(ours)	**40.0**	**64.4**	**41.8**	**60.8**	**38.4**	62.7	**42.7**	**68.3**	**31.5**	**75.4**	**36.1**	**75.0**	80.5	**81.3**	**26.2**	**88.1**	**41.3**	**85.9**	**36.5**	**71.4**	**40.6**	**80.6**	**455.7**	**813.8**
GPT-4o	51.8	82.6	42.4	78.2	40.1	80.3	54.8	86.6	46.6	91.3	38.7	88.6	96.7	93.7	32.7	90.0	48.5	92.5	47.2	79.6	44.9	94.8	544.3	958.1
Gemini-2.5	48.0	77.8	40.4	66.4	42.2	61.5	46.6	71.9	38.0	86.1	38.7	72.9	88.3	91.1	28.1	88.4	50.0	84.2	43.9	75.0	41.9	98.5	505.9	873.8

Table 1: We evaluate our method and other baselines on BiscuitVQA benchmark. The evaluation covers four visual concept types (Human, Cartoon, Animal, Symbol), four task types (Brief Answer, Detailed Description, Single Choice, Chain-of-Thought), and varying numbers of image-introduced concepts within a single benchmark sample. Performance is assessed using two metrics: S_{ANLS} and S_{GPT}. "Overall" denotes the sum of all scores. **Bold and underlined** indicates the best performance among open-source methods, while **bold only** indicates the second best.

Concept and Task Types: Our dataset covers four concept categories: (1) real-world people (Human), (2) cartoon characters (Cartoon), (3) animals (Animal), and (4) symbolic content (Symbol) such as charts and tables. For each concept, we create four question types: Brief Answer (Brief), Detailed Description (Detail), Chain-of-Thought (CoT), and Single Choice (Choice), targeting different reasoning abilities. We also include multi-concept cases requiring the model to distinguish and reason over several novel concepts in one image. Notably, to prevent models from relying solely on surface-level name matching, we design a subset of samples where the model engages with novel visual concepts without the need to explicitly state their names in the answer, corresponding to a concept count of zero, encouraging deeper understanding of the visual context, as shown in Figure 3. The dataset includes 31,384 samples for Step I (recognition) and 38,954 for Step II (generative reasoning). Additionally, we construct a test benchmark of 5,845 samples for fair evaluation, ensuring they are excluded from training.

Construction Process: To introduce new concepts visually, we construct a concept relationship graph for each image and place it next to the original image, forming a composite input as shown in Figure 2. These composite images present novel names or symbols within visual context, without relying on textual prompts. Initial answers are generated by Claude-3.5(Anthropic 2024) and then refined by human annotators to ensure consistency with the visual concepts.

Benchmark Evaluation

We evaluate BISCUIT on our BiscuitVQA benchmark against both open-source and closed-source models. Open-source baselines include state-of-the-art models from 2024 and 2025: MyVLM(Alaluf et al. 2024), Yo'LLaVA(Nguyen et al. 2024) , MC-LLaVA(An et al. 2024) , and Yo'Chameleon(Nguyen et al. 2025). For fairness, we adopt their standard concept-injection strategies, such as name prompting in Yo'LLaVA. Closed-source models include GPT-4o and Gemini-2.5. To reduce format-related variations, we standardize all model outputs using GPT-4o-mini by removing template artifacts and irrelevant prefixes based on different input types. We further employ two metrics to

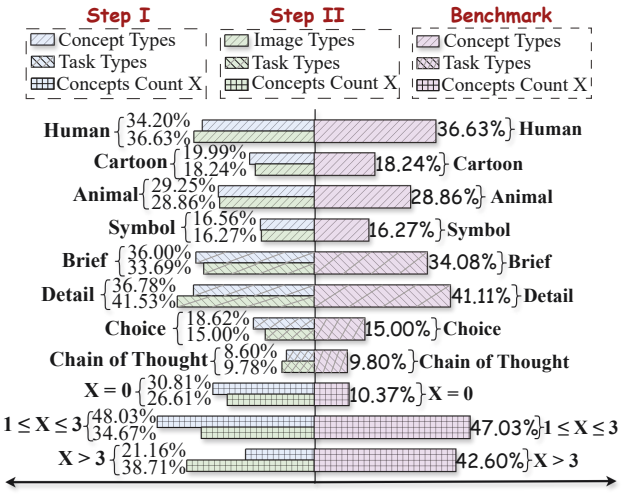


Figure 3: Left bars represent data distributions in Step I and Step II, while the right bars summarize the BiscuitVQA benchmark status, with an average token length of 185. Step I with an average token length of 161. Step II extends Step I with more detailed reasoning, increasing the average token length from 161 to 216.

evaluate each model's performance: (1) Average Normalized Levenshtein Similarity ($S_{\mathbf{ANLS}}$)(Peer et al. 2024), which measures character-level similarity between predictions and references; and (2) GPT-4o Rating ($S_{\mathbf{GPT}}$), which scores semantic alignment between model outputs and ground-truth answers using GPT-4o with a fixed prompt. Each benchmark sample receives a score between 0.1 and 1.0 for both ANLS and GPT-4o rating. We compute the final benchmark scores S_{ANLS} and S_{GPT} by averaging the respective scores across all samples and multiplying the result by 100 to obtain a percentage.

Image-Introduced Concept Learning Test: As shown in Table 1, BISCUIT consistently outperforms all open-source baselines. Specifically, it brings the largest performance gains over base models across all methods. On average, BISCUIT improves the Qwen2.5-VL series by approximately 40% in S_{ANLS} and 20% in S_{GPT}, and boosts LLaVA-1.5 by around 36% and 12% respectively. Moreover, BISCUIT even achieves performance comparable to GPT-4o and Gemini-2.5, especially on Cartoon and Detailed Description subsets under the LLaVA-1.5-13B setting.

Generalization Test: In addition to achieving competitive performance on image-introduced concept benchmark, an interesting observation is that our method retains strong generalization to unrelated tasks. We evaluate on benchmarks across multiple domains (e.g., SQA(Saikh et al. 2022), OCR-VQA(Mishra et al. 2019), HB(Guan et al. 2024) et al.). As shown in Figure 4, our method retains an average of 85.28% and 94.07% of generalization performance on Qwen2.5-VL-3B and Qwen2.5-VL-7B, outperforming the second-best results of 81.21% and 91.09% respectively. On LLaVA-1.5-7B and 13B, we achieve the best retention, with averages of 94.87% and 94.29%.

Methods	Qwen2.5-VL-3B		Qwen2.5-VL-7B		LLaVA-1.5-7B		LLaVA-1.5-13B	
	IICL	Gene	IICL	Gene	IICL	Gene	IICL	Gene
w/o	**544.8**	**85.5%**	608.0	**95.2%**	511.8	**94.2%**	611.1	92.4%
Laplacian	540.8	79.5%	617.8	91.1%	**522.6**	91.5%	602.6	90.7%
Sobel	515.4	77.1%	587.4	84.5%	500.4	86.4%	595.2	87.7%
HED	578.4	81.0%	**622.8**	86.4%	514.0	92.0%	**622.1**	**92.4%**
Canny	**583.5**	**85.3%**	**641.8**	**94.1%**	**554.3**	**94.9%**	**634.8**	**94.3%**

Table 2: Performance comparison of different edge detection methods used in Step I. Image-Introduced Concept Learning (IICL) is computed as the average of the Overall scores in Table 1, while Generalization (Gene) represents the mean performance across different benchmarks shown in Figure 4.

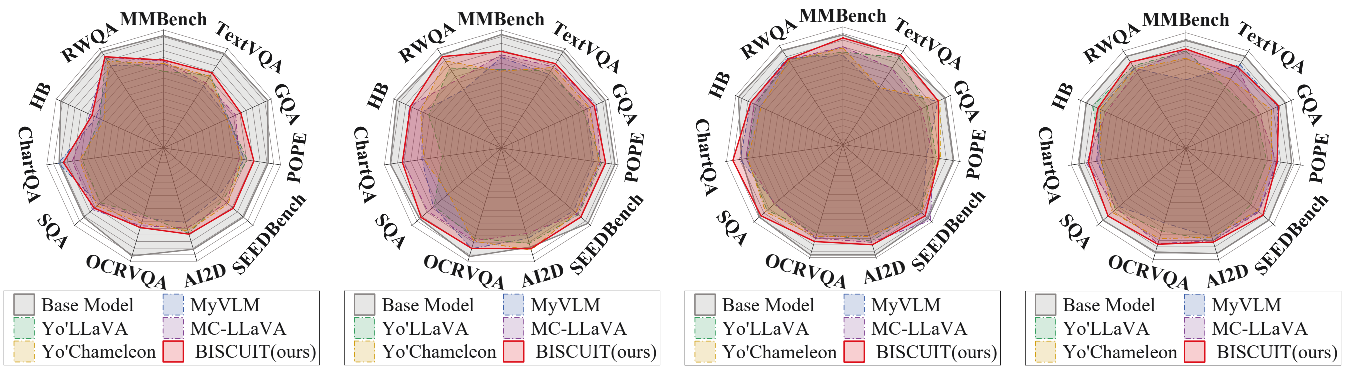
Methods	Qwen2.5-VL-3B		Qwen2.5-VL-7B		LLaVA-1.5-7B		LLaVA-1.5-13B	
	IICL	Gene	IICL	Gene	IICL	Gene	IICL	Gene
Sum	539.8	83.0%	**621.5**	88.9%	528.0	90.0%	609.0	89.2%
SVD	550.1	83.5%	587.1	91.4%	513.4	**92.2%**	602.4	**92.1%**
LU	557.7	**85.3%**	605.5	89.4%	505.2	91.3%	**616.5**	86.5%
Eigen	**561.6**	79.2%	619.2	**91.7%**	**531.5**	86.7%	597.1	89.3%
QR	**583.5**	**85.3%**	**641.8**	**94.1%**	**554.3**	**94.9%**	**634.8**	**94.3%**

Table 3: IICL and Gene performance with different fusion methods of edge and original embeddings in step I.

We attribute this to the BISCUIT architecture. Specifically, in Step I, the original vision encoder is fully frozen, preserving the model's foundational visual representations. In Step II, although we fine-tune the connector and text decoder with identification-aware loss, the fusion of dual-stream embeddings introduces structural cues without disrupting the pre-trained vision-text alignment. This setup prevents catastrophic forgetting and allows the model to incorporate new concepts while leveraging existing knowledge.

Edge Image Variants: In Step I, we employ Canny edge detection to generate edge images. To assess its effectiveness, we compare it with Laplacian(Van Dokkum 2001), Sobel(Gao et al. 2010), and HED(Xie and Tu 2015), as shown in Table 2, "w/o" indicates that no edge detection methods are used. Canny shows the highest overall performance across both concept learning and generalization. We attribute this to its ability to preserve key structural contours (e.g., facial outlines, chart legends) while suppressing background noisy. However, HED tends to over-emphasize background regions, and Laplacian and Sobel often miss fine-grained details crucial for identifying characters and symbolic content.

Fusion Method Comparison: In Step I, to validate our use of QR decomposition for aligning RGB and edge-stream embeddings, we conduct a comparative study with other common matrix decomposition techniques: SVD, LU decomposition(Bartels and Golub 1969), and Eigenvalue decomposition(Hall, Marshall, and Martin 2002), as shown in Table 3, "Sum" denotes the direct addition of the two embeddings. Results indicate that QR-based basis projection yields the best performance. We attribute this to the orthogonality-preserving nature of QR, which enables more stable and consistent projection into the shared structure space. Unlike SVD or LU decomposition, QR avoids over-



(a) Qwen2.5-VL-3B (b) Qwen2.5-VL-7B (c) LLaVA-1.5-7B (d) LLaVA-1.5-13B

Figure 4: To quantify the impact of image-introduced concept learning on generalization, we evaluate four models on diverse downstream benchmarks. BISCUIT achieves the highest average performance retention across all methods.

♣ ♠ ♦	Concept Types								Task Types								Concepts Count x						Overall↑		Gene↑
	Human↑		Cartoon↑		Animal↑		Symbol↑		Brief↑		Detail↑		Choice↑		COT↑		$x=0$↑		$1 \leq x \leq 3$↑		$x>3$↑				
	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	S_{ANLS}	S_{GPT}	
✔ ✔ ✔	**40.1**	**71.4**	**42.3**	**65.5**	**42.9**	**68.6**	**31.4**	**65.1**	**30.2**	**68.0**	**35.5**	**71.7**	**72.7**	**78.7**	**61.6**	**76.7**	**37.9**	**82.6**	**40.2**	**76.0**	**49.0**	**75.5**	**483.9**	**799.6**	**94.1%**
✔ ✔ ✘	35.8	62.5	38.1	55.7	**38.9**	62.6	27.8	60.8	30.7	60.1	32.4	66.4	**69.3**	67.4	**58.4**	**71.2**	33.5	79.0	36.3	70.1	45.0	65.8	**446.3**	721.7	94.7%
✔ ✘ ✔	**36.7**	**63.9**	37.2	57.6	37.8	62.9	27.8	**62.0**	27.0	62.2	**32.6**	**70.8**	65.2	70.2	56.3	70.5	**35.9**	79.6	36.2	**71.1**	**45.0**	67.7	437.6	**738.6**	91.0%
✘ ✔ ✔	36.4	63.5	**38.5**	**62.3**	38.5	60.4	**28.9**	58.0	28.8	61.2	32.2	67.3	68.3	**70.9**	55.4	66.2	34.6	**79.9**	**36.3**	70.5	43.5	**71.2**	441.5	731.4	89.7%
✔ ✘ ✘	29.4	63.8	34.0	53.2	34.9	61.4	26.0	56.8	23.9	**63.1**	29.5	60.6	64.2	68.3	52.4	65.8	29.8	75.8	34.3	66.6	41.9	67.3	400.4	702.6	89.9%
✘ ✔ ✘	33.0	63.0	34.3	57.1	36.3	**63.6**	26.2	58.6	25.1	61.4	30.4	65.7	66.9	66.9	55.0	68.0	34.7	74.5	34.6	68.4	41.1	64.8	417.5	712.2	89.6%
✘ ✘ ✔	30.4	60.8	37.6	54.5	34.3	59.4	28.2	56.1	26.6	60.5	28.8	62.4	64.2	64.2	53.8	70.0	33.0	78.4	34.0	69.5	42.1	66.3	413.0	702.0	87.3%

Table 4: Ablation results on Qwen2.5-VL-7B assessing the impact of each module on benchmark scores and generalization performance. "♣" denotes the dual-stream encoder, "♠" indicates the initialization strategy for edge-stream modules, and "♦" represents the identification-aware loss.

compression or skewed scaling of the feature components from the original and edge image embeddings, which could otherwise distort the alignment of local structural cues.

Loss Parameter Sensitivity Analysis: We study how the identification-aware loss threshold τ affects model performance by varying it from 0.0 to 0.5. As shown in Figure 5, setting $\tau = 0.0$ leads to relatively high image-introduced concept learning benchmark performance but significantly harms generalization, especially on Qwen2.5-VL-3B. This phenomenon is intuitive: when τ is too low, the model focuses excessively on matching the positive target y^+ at the token level. This causes the output style to gradually converge toward y^+, eventually leading to a mismatched or overfitted representation that undermines generalization. Conversely, when τ is too high, the model fails to effectively learn the intended structure of y^+, leading to weaker performance of concept learning. To strike a balance, we set $\tau = 0.1$ as the default, which achieves both effective concept learning and generalization preservation.

Ablation Studies: We further evaluate the contribution of each component in BISCUIT, as shown in Table 4. We consider three variants: (1) replacing the dual-stream encoder with the original single-stream vision encoder, (2) replacing SVD-based initialization with random weights, and (3) substituting the identification-aware loss with standard DPO. Interestingly, removing Step I components leads to a notable decline in generation quality, while removing the identification-aware loss results in a clear drop in S_{GPT}.

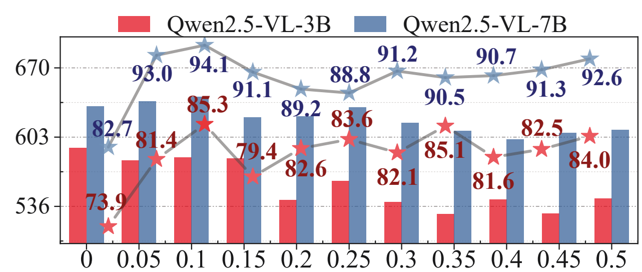


Figure 5: Model performance score (y-axis) trends under different τ values (x-axis) in step II. We evaluate both our benchmark score (bar) and generalization score (%) (line).

Conclusion

We propose BISCUIT, a two-step training method that mitigates VLMs' failure to learn image-introduced concepts: Step I enhances concept recognition via a dual-stream, structure-aware encoder, and Step II improves text generation with a novel identification-aware loss. Furthermore, we construct a BiscuitVQA benchmark covering diverse concept types and task formats. Experiments show that BISCUIT outperforms open-source baselines and better preserves generalization to other tasks.

Acknowledgments

This work is partly supported by the Project granted by the Ministry of Agriculture and Rural Affairs, the Key Research and Development Program of Heilongjiang Province under Grant Nos. 2022ZX01A22, 2021ZXJ05A03, the National Natural Science Foundation of China under Grant Nos. 62350710797, 61972114, 62106061, the National Science and Technology Major Project of China under Grant Nos. 2021ZD0110901, the collaborative innovation and promotion system of the modern agricultural industry technology for watermelon and melon in Heilongjiang Province.

References

Abdi, H. 2007. Singular value decomposition (SVD) and generalized singular value decomposition. *Encyclopedia of measurement and statistics*, 907(912): 44.

Alaluf, Y.; Richardson, E.; Tulyakov, S.; Aberman, K.; and Cohen-Or, D. 2024. Myvlm: Personalizing vlms for user-specific queries. In *European Conference on Computer Vision*, 73–91. Springer.

An, R.; Yang, S.; Lu, M.; Zhang, R.; Zeng, K.; Luo, Y.; Cao, J.; Liang, H.; Chen, Y.; She, Q.; et al. 2024. Mc-llava: Multi-concept personalized vision-language model. *arXiv preprint arXiv:2411.11706*.

Anthropic. 2024. Claude 3.5 Sonnet Model Card: October Addendum. Technical report, Anthropic.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Bao, J.; Cheng, S.; Du, J.; He, C.; Lang, Z.; Zhang, H.; and Liu, J. 2025a. BOLT: Fewer Tokens but More Performance Retention for Efficient Vision-Language Models Inference. In *Proceedings of the 33rd ACM International Conference on Multimedia*, 4058–4067.

Bao, J.; Cheng, S.; Du, J.; Li, Z.; He, C.; and Liu, J. 2025b. History Tracker: Retrieving Historical Image Embeddings for Efficient Fine-Grained Reasoning in Vision-Language Models. In *IEEE International Conference on Multimedia and Expo, ICME 2025, Nantes, France, June 30 - July 4, 2025*, 1–6. IEEE.

Bartels, R. H.; and Golub, G. H. 1969. The simplex method of linear programming using LU decomposition. *Communications of the ACM*, 12(5): 266–268.

Bordes, F.; Pang, R. Y.; Ajay, A.; Li, A. C.; Bardes, A.; Petryk, S.; Mañas, O.; Lin, Z.; Mahmoud, A.; Jayaraman, B.; et al. 2024. An introduction to vision-language modeling. *arXiv preprint arXiv:2405.17247*.

Bouritsas, G.; Koutras, P.; Zlatintsi, A.; and Maragos, P. 2018. Multimodal visual concept learning with weakly supervised techniques. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4914–4923.

Chen, Z.; Huang, X.; Fan, X.; Wang, K.; Zhou, Y.; Guan, Q.; and Lin, L. 2025. Reproducible Vision-Language Models Meet Concepts Out of Pre-Training. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 14701–14711.

Chen, Z.; Wu, J.; Wang, W.; Su, W.; Chen, G.; Xing, S.; Zhong, M.; Zhang, Q.; Zhu, X.; Lu, L.; et al. 2024. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 24185–24198.

Comanici, G.; Bieber, E.; Schaekermann, M.; Pasupat, I.; Sachdeva, N.; Dhillon, I.; Blistein, M.; Ram, O.; Zhang, D.; Rosen, E.; et al. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Ding, L.; and Goshtasby, A. 2001. On the Canny edge detector. *Pattern recognition*, 34(3): 721–725.

Fini, E.; Shukor, M.; Li, X.; Dufter, P.; Klein, M.; Haldimann, D.; Aitharaju, S.; da Costa, V. G. T.; Béthune, L.; Gan, Z.; et al. 2025. Multimodal autoregressive pre-training of large vision encoders. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 9641–9654.

Gao, W.; Zhang, X.; Yang, L.; and Liu, H. 2010. An improved Sobel edge detection. In *2010 3rd International conference on computer science and information technology*, volume 5, 67–71. IEEE.

Ge, C.; Wang, X.; Zhang, Z.; Chen, H.; Fan, J.; Huang, L.; Xue, H.; and Zhu, W. 2025. Dynamic Mixture of Curriculum LoRA Experts for Continual Multimodal Instruction Tuning. *arXiv preprint arXiv:2506.11672*.

Guan, T.; Liu, F.; Wu, X.; Xian, R.; Li, Z.; Liu, X.; Wang, X.; Chen, L.; Huang, F.; Yacoob, Y.; et al. 2024. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14375–14385.

Hall, P.; Marshall, D.; and Martin, R. 2002. Adding and subtracting eigenspaces with eigenvalue decomposition and singular value decomposition. *Image and Vision Computing*, 20(13-14): 1009–1016.

Huang, K.-H.; Chan, H. P.; Fung, Y. R.; Qiu, H.; Zhou, M.; Joty, S.; Chang, S.-F.; and Ji, H. 2024a. From pixels to insights: A survey on automatic chart understanding in the era of large foundation models. *IEEE Transactions on Knowledge and Data Engineering*.

Huang, W.; Liang, J.; Shi, Z.; Zhu, D.; Wan, G.; Li, H.; Du, B.; Tao, D.; and Ye, M. 2024b. Learn from downstream and be yourself in multimodal large language model fine-tuning. *arXiv preprint arXiv:2411.10928*.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Jin, Y.; Li, J.; Liu, Y.; Gu, T.; Wu, K.; Jiang, Z.; He, M.; Zhao, B.; Tan, X.; Gan, Z.; et al. 2024. Efficient multimodal large language models: A survey. *arXiv preprint arXiv:2405.10739*.

Keraghel, I.; Morbieu, S.; and Nadif, M. 2024. A survey on recent advances in named entity recognition. *arXiv preprint arXiv:2401.10825*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.

Laurençon, H.; Marafioti, A.; Sanh, V.; and Tronchon, L. 2024. Building and better understanding vision-language models: insights and future directions. *arXiv preprint arXiv:2408.12637*.

Lee, S.; Zhang, Y.; Wu, S.; and Wu, J. 2023. Language-informed visual concept learning. *arXiv preprint arXiv:2312.03587*.

Li, T.; Ma, M.; and Peng, X. 2024. Deal: Disentangle and localize concept-level explanations for vlms. In *European Conference on Computer Vision*, 383–401. Springer.

Li, Z.; Wu, X.; Du, H.; Liu, F.; Nghiem, H.; and Shi, G. 2025. A survey of state of the art large vision language models: Alignment, benchmark, evaluations and challenges. *arXiv preprint arXiv:2501.02189*.

Liang, C. X.; Tian, P.; Yin, C. H.; Yua, Y.; An-Hou, W.; Ming, L.; Wang, T.; Bi, Z.; and Liu, M. 2024. A comprehensive survey and guide to multimodal large language models in vision-language tasks. *arXiv preprint arXiv:2411.06284*.

Liu, H.; Li, C.; Li, Y.; and Lee, Y. J. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 26296–26306.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2024b. Visual instruction tuning. *Advances in neural information processing systems*, 36.

Mishra, A.; Shekhar, S.; Singh, A. K.; and Chakraborty, A. 2019. Ocr-vqa: Visual question answering by reading text in images. In *2019 international conference on document analysis and recognition (ICDAR)*, 947–952. IEEE.

Nguyen, T.; Liu, H.; Li, Y.; Cai, M.; Ojha, U.; and Lee, Y. J. 2024. Yo'llava: Your personalized language and vision assistant. *Advances in Neural Information Processing Systems*, 37: 40913–40951.

Nguyen, T.; Singh, K. K.; Shi, J.; Bui, T.; Lee, Y. J.; and Li, Y. 2025. Yo'Chameleon: Personalized Vision and Language Generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 14438–14448.

Patel, M.; Gokhale, T.; Baral, C.; and Yang, Y. 2024. Conceptbed: Evaluating concept learning abilities of text-to-image diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 14554–14562.

Peer, D.; Schopf, P.; Nebendahl, V.; Rietzler, A.; and Stabinger, S. 2024. ANLS*–A Universal Document Processing Metric for Generative Large Language Models. *arXiv preprint arXiv:2402.03848*.

Rafailov, R.; Sharma, A.; Mitchell, E.; Manning, C. D.; Ermon, S.; and Finn, C. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36: 53728–53741.

Rudman, W.; Golovanevsky, M.; Bar, A.; Palit, V.; LeCun, Y.; Eickhoff, C.; and Singh, R. 2025. Forgotten polygons: Multimodal large language models are shape-blind. *arXiv preprint arXiv:2502.15969*.

Saikh, T.; Ghosal, T.; Mittal, A.; Ekbal, A.; and Bhattacharyya, P. 2022. Scienceqa: A novel resource for question answering on scholarly articles. *International Journal on Digital Libraries*, 23(3): 289–301.

Van Dokkum, P. G. 2001. Cosmic-ray rejection by laplacian edge detection. *Publications of the Astronomical Society of the Pacific*, 113(789): 1420.

Wang, D.; Li, M.; Liu, X.; Xu, M.; Chen, B.; and Zhang, H. 2023a. Tuning multi-mode token-level prompt alignment across modalities. *Advances in Neural Information Processing Systems*, 36: 52792–52810.

Wang, P.; Bai, S.; Tan, S.; Wang, S.; Fan, Z.; Bai, J.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; et al. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Wang, X.; Chen, G.; Qian, G.; Gao, P.; Wei, X.-Y.; Wang, Y.; Tian, Y.; and Gao, W. 2023b. Large-scale multi-modal pre-trained models: A comprehensive survey. *Machine Intelligence Research*, 20(4): 447–482.

Wu, J.; Lyu, H.; Xia, Y.; Zhang, Z.; Barrow, J.; Kumar, I.; Mirtaheri, M.; Chen, H.; Rossi, R. A.; Dernoncourt, F.; et al. 2024a. Personalized multimodal large language models: A survey. *arXiv preprint arXiv:2412.02142*.

Wu, Y.; Yan, L.; Shen, L.; Wang, Y.; Tang, N.; and Luo, Y. 2024b. Chartinsights: Evaluating multimodal large language models for low-level chart question answering. *arXiv preprint arXiv:2405.07001*.

Xia, P.; Zhang, L.; and Li, F. 2015. Learning similarity with cosine similarity ensemble. *Information sciences*, 307: 39–52.

Xie, S.; and Tu, Z. 2015. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, 1395–1403.

Xu, R.; Wei, T.; Wei, Y.; and Xie, P. 2024. QR decomposition of dual matrices and its application. *Applied Mathematics Letters*, 156: 109144.

Yan, Y.; Miao, Y.; Li, J.; Zhang, Y.; Xie, J.; Deng, Z.; and Yan, D. 2024. 3d-properties: Identifying challenges in dpo and charting a path forward. *arXiv preprint arXiv:2406.07327*.

Yang, Y.; Li, Z.; Dong, Q.; Xia, H.; and Sui, Z. 2024. Can large multimodal models uncover deep semantics behind images? *arXiv preprint arXiv:2402.11281*.

Zhang, J.; Huang, J.; et al. 2024. Vision-language models for vision tasks: A survey. *IEEE TPAMI*.

Zhang, Y.; He, Q.; Wang, X.; Yuan, S.; Liang, J.; and Xiao, Y. 2024. Light Up the Shadows: Enhance Long-Tailed Entity Grounding with Concept-Guided Vision-Language Models. *arXiv preprint arXiv:2406.10902*.