PaperBanana: Automating Academic Illustration for AI Scientists

Dawei Zhu, Rui Meng, Yale Song, Xiyu Wei, Sujian Li, Tomas Pfister, Jinsung Yoon

Peking University & Google Cloud AI Research

arXiv: [v2] 24 Mar 2026

Saman Khamesian **03/25/2026**

Introduction

- Autonomous AI Scientists have shown strong capabilities in literature review, idea generation, and experiment iteration — yet visual communication remains a critical gap.

- Despite proficiency in textual analysis and code execution, current systems struggle to generate illustrations that adhere to the rigorous standards of academic manuscripts.

- Generating methodology diagrams is particularly challenging: it demands both content fidelity (accurate representation of technical flows) and visual aesthetics (adherence to scholarly norms).

Introduction

- Code-based methods (TikZ, Python-PPTX, SVG) are effective for structured content but face expressiveness limitations for intricate visual elements — specialized icons, custom shapes.

- Image generation models produce visually sophisticated outputs but consistently fail to meet the strict standards of academic illustrations.

- Professional illustration tools require specialized expertise, forcing researchers to invest substantial manual effort.

- **Our goal:** Automate the production of publication-ready academic illustrations, removing this bottleneck from the research workflow.

Introduction

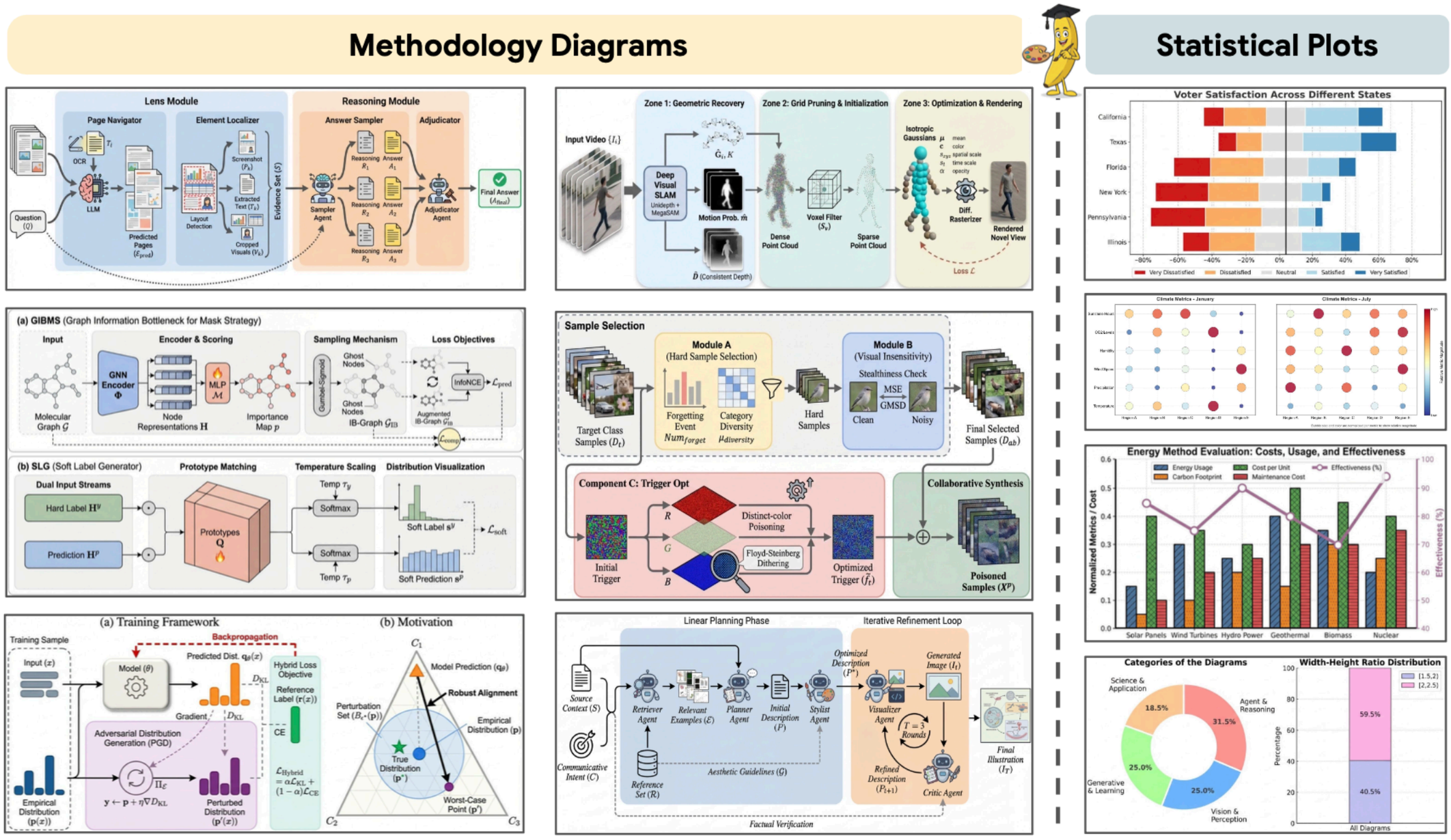


Figure 1 | Examples of methodology diagrams and statistical plots generated by PAPERBANANA, which show the potential of automating the generation of academic illustrations.

Task Formulation

- The task is defined as learning a mapping from a **source context** and a **communicative intent** to a **visual representation.**

- **Let S denote the source context** — the essential information (e.g., the methodology section of a paper).

- **Let C denote the communicative intent** — specifying the scope and focus of the desired illustration (e.g., the figure caption).

- The goal is to **generate an image** I that faithfully visualizes S while fulfilling C:

$$I = f(S, C)$$

Task Formulation

- To further guide generation, the input can be augmented with a set of N reference examples:

$$\mathscr{E} = \{E_n\}_{n=1}^{n}$$

- Each example E_n is a triplet (S_n, C_n, I_n)— a reference illustration I_n paired with its corresponding context S_n and communicative intent C_n. The unified formulation becomes:

$$I_n = f(S, C, \mathscr{E})$$

- Where $\mathscr{E}$ defaults to $\varnothing$ when no examples are used (i.e., zero-shot generation).

Framework Overview

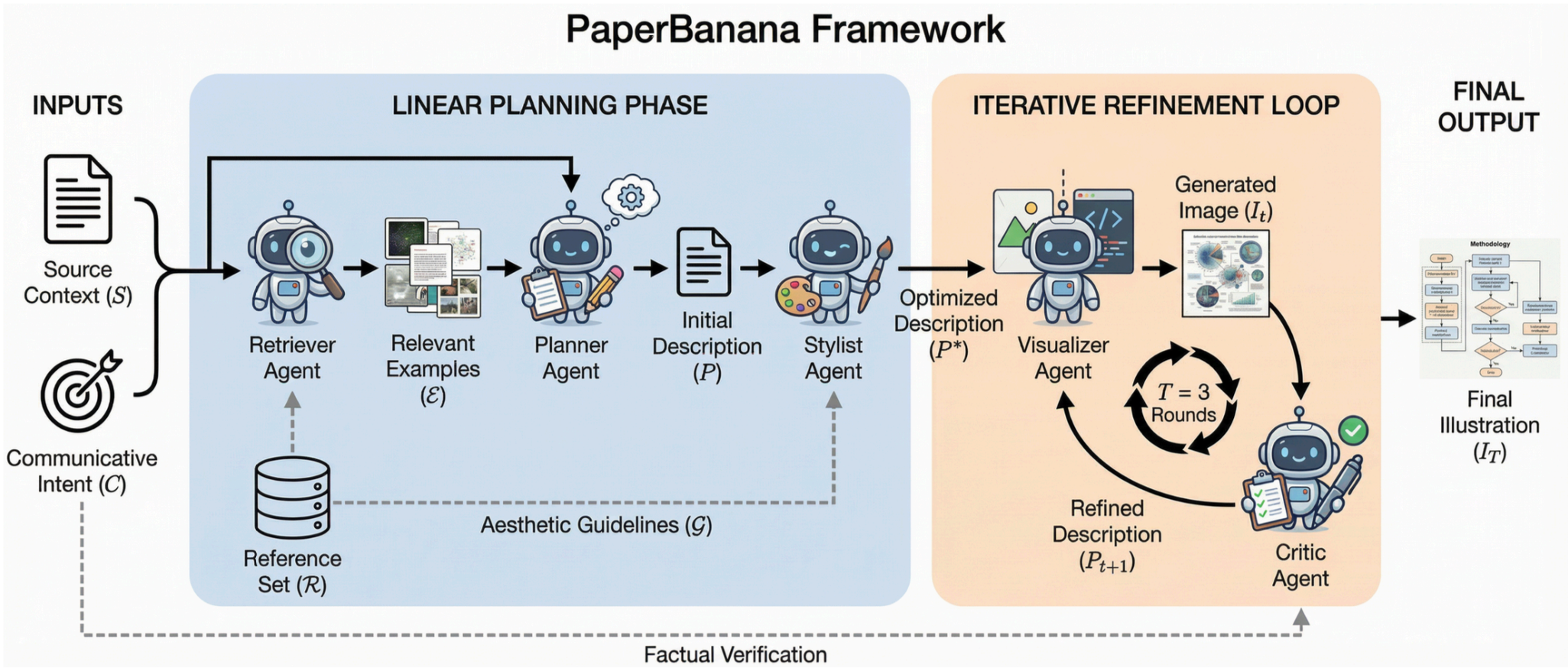


Figure 2 | [Generated by 🍌, textual description to reproduce this diagram is presented in Appendix E.] Overview of our PAPERBANANA framework. Given the source context and communicative intent, we first apply a *Linear Planning Phase* to retrieve relevant reference examples and synthesize a stylistically optimized description. We then use an *Iterative Refinement Loop* (consisting of *Visualizer* and *Critic Agents*) to transform the description into visual output and conduct multi-round refinements to produce the final academic illustration.

Framework Overview

- PaperBanana orchestrates a collaborative team of five specialized agents:

Agent	Role
Retriver	Identifies relevant reference examples
Planner	Translates source content into a detailed illustration description
Stylist	Refines the description to meet academic aesthetic standards
Visualizer	Renders the description into a visual image
Critic	Evaluates the output and provides targeted refinement feedback

- The pipeline consists of two main phases:
  - **Linear Planning Phase:** Retriever → Planner → Stylist
  - **Iterative Refinement Loop:** Visualizer ↔ Critic (3 rounds)

Methodology

Agent 1: Retriever

- **Role:** Identify the most relevant reference examples from the fixed reference set R to guide downstream agents.

- **Mechanism:** A generative retrieval approach — the VLM performs reasoned selection over candidate metadata:

$$\mathscr{E} = VLM_{Ret}(S, C, \{(S_i, C_i)\}_{E_i \in R})$$

- Selection criteria (prioritized order):

  1. Visual structure and diagram type (e.g., pipeline, architecture)

  2. Research domain (e.g., Agent & Reasoning, Vision & Perception)

Methodology

Agent 2: Planner

- **Role:** Transform unstructured or structured source content into a comprehensive, detailed textual description of the target illustration.

- **Inputs:** Source context S, communicative intent C, and retrieved examples E.

- **Mechanism:** In-context learning from the retrieved demonstrations:

$$P = VLM_{plan}(S, C, \{(S_i, C_i, I_i)\}_{E_i \in \mathscr{E}})$$

- **Key contribution:** By observing how reference illustrations I_i correspond to their contexts and captions, the Planner learns the compositional logic required for academic diagrams — what elements to include, how to organize information flow, and what level of detail is appropriate.

Methodology

Agent 3: Stylist

- **Role:** Refine the illustration description to ensure adherence to the aesthetic standards of modern academic manuscripts.

- **The core challenge:** Defining a comprehensive *academic style* manually is incomplete and subjective.

- **Solution:** The Stylist traverses the entire reference collection R to automatically synthesize an Aesthetic Guideline $\mathcal{G}$, covering:
  - Color palette
  - Shapes and containers
  - Lines and arrows
  - Layout and composition
  - Typography and icons

Methodology

Agent 3: Stylist

- **Mechanism:** Armed with $\mathcal{G}$, the Stylist refines the initial description P into a stylistically optimized version P^*:

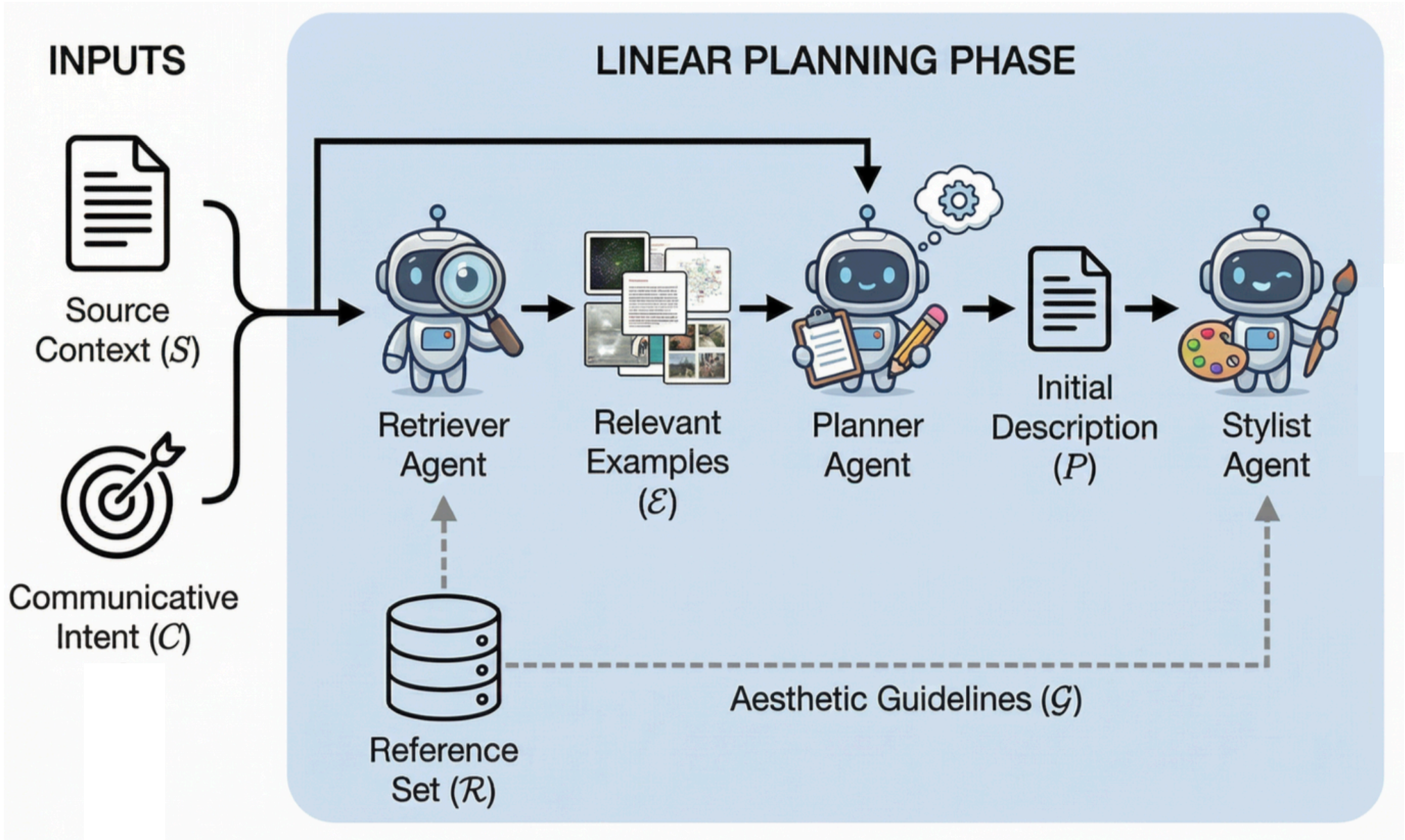
$$P^* = VLM_{style}(P, \mathcal{G})$$

- **Outcome:** The final illustration is not only accurate in content but also visually professional in presentation.

- **Note:** The input P (Output of the Planner agent) is a rich, structured textual description that serves as the backbone for all subsequent rendering steps.

Methodology

Summary (Linear Planning Phase)



Methodology

Agent 4: Visualizer

- **Role:** Leverages an image generation model to transform the optimized description into a visual output.

- **Mechanism:** At each iteration t, given description P_t, the Visualizer generates:

$$I_t = Image - Gen(P_t)$$

- Where the initial description P_0 is set to $P*$.

Methodology

Agent 5: Critic

- **Role:** Forms a closed-loop refinement mechanism with the Visualizer.

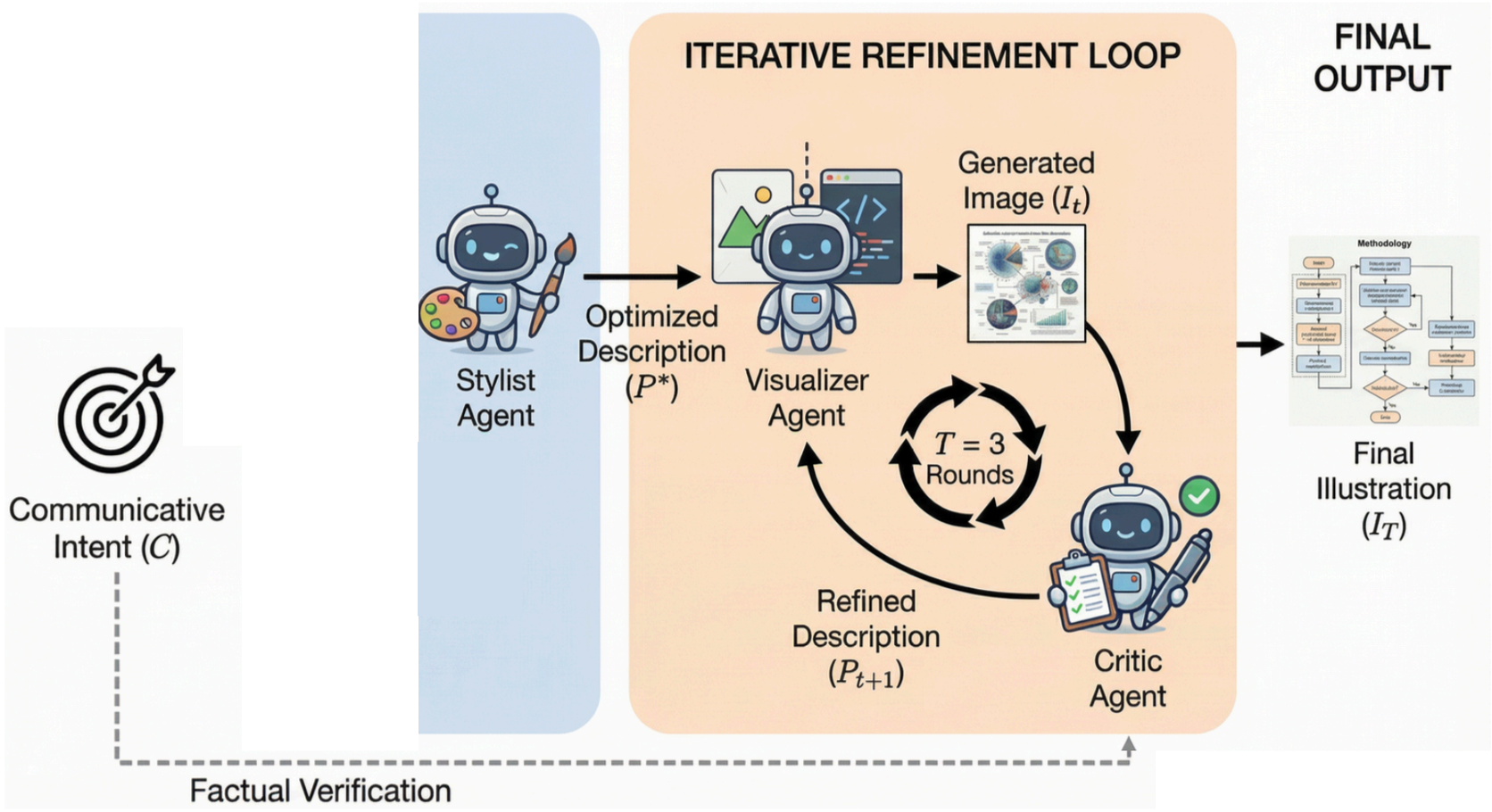
- **Mechanism:** Upon receiving image I_t, the Critic inspects it against the original source context (S, C) to identify:
  - Factual misalignments
  - Visual glitches
  - Areas for improvement

- **Output:** Produces a refined description P_{t+1}:

$$P_{t+1} = VLM_{critic}(I_t, S, C, P_t)$$

Methodology

Summary (Iterative Refinement Loop)



- The loop runs for $T = 3$ rounds, with the final output $I = I_t$. This iterative process ensures a balance between aesthetic polish (Stylist contribution) and technical accuracy (Critic correction) — the Critic effectively recovers faithfulness that visual polishing may sacrifice.

PaperBananaBench

A Dedicated Evaluation Benchmark

- **Motivation:** No dedicated benchmark existed for evaluating automated academic diagram generation.

1. **Collection & Parsing:** Randomly sampled 2,000 papers from 5,275 NeurIPS 2025 publications; extracted methodology sections, diagrams, and captions using the MinerU toolkit.

2. **Filtering:** Retained papers with methodology diagrams and restricted aspect ratio (w:h) to [1.5, 2.5] **(ratios below 1.5 exclude typical landscape layouts; ratios above 2.5 are unsupported by current image generation models).**

PaperBananaBench

A Dedicated Evaluation Benchmark

3. **Categorization:** Diagrams classified into four categories based on visual topology and content:
   A. Agent & Reasoning (31.5%)
   B. Vision & Perception (25.0%)
   C. Generative & Learning (25.0%)
   D. Science & Applications (18.5%)

4. **Human Curation:** Annotators verified descriptions, validated categorizations, and filtered low-quality diagrams.

❖ **Final dataset:** 584 samples, randomly split into:

- **Test set (N = 292)** — for evaluation
- **Reference set (N = 292)** — for retrieval-augmented in-context learning

Evaluation Protocol

VLM-as-a-Judge with Referenced Scoring

- **Approach:** A referenced comparison — the VLM judge compares the model-generated diagram against the human-drawn diagram to determine which better satisfies each criterion.

- **Four evaluation dimensions (Inspired by Quispel et al., 2018) [Appendix H]:**

A. Content:

1. Faithfulness — Alignment with source context and communicative intent
2. Conciseness — Focus on core information without visual clutter

B. Presentation:

3. Readability — Intelligible layout, legible text, no excessive crossing lines
4. Aesthetics — Adherence to stylistic norms of academic manuscripts

Evaluation Protocol

VLM-as-a-Judge with Referenced Scoring

- **Referenced Scoring:** For each dimension, the VLM judge compares the model-generated diagram against the human reference given the context and caption.

- It determines Model wins, Human wins, or Tie based on relative quality, which are then mapped to scores of 100, 0, and 50, respectively.

- **Hierarchical aggregation for Overall score:**
  - **Primary dimensions:** Faithfulness & Readability
  - **Secondary dimensions:** Conciseness & Aesthetics
  - Primary dimensions determine the overall winner.
  - In case of a tie, we apply the same rule to the secondary dimensions.

Experiments

Baseline Methods and Models

- **Vanilla:** Direct prompting of the image generation model
- **Few-shot:** Vanilla + 10 in-context examples
- **Paper2Any:** Closest agentic baseline (focuses on high-level ideas rather than faithful methodological flows)

Table 1 | Main results on PAPERBANANABENCH. Best score in each column is in **bold**.

Method	Faithfulness ↑	Conciseness ↑	Readability ↑	Aesthetic ↑	Overall ↑
Vanilla Settings					
GPT-Image-1.5	4.5	37.5	30.0	37.0	11.5
Nano-Banana-Pro	43.0	43.5	38.5	65.5	43.2
Few-shot Nano-Banana-Pro	41.6	49.6	37.6	60.5	41.8
Agentic Frameworks					
Paper2Any (w/ Nano-Banana-Pro)	6.5	44.0	20.5	40.0	8.5
PAPERBANANA (Ours)					
w/ GPT-Image-1.5	16.0	65.0	33.0	56.0	19.0
w/ Nano-Banana-Pro	45.8	**80.7**	**51.4**	**72.1**	**60.2**
Human	**50.0**	50.0	50.0	50.0	50.0

Experiments

Baseline Methods and Models

- Figure 4 compares PaperBanana with vanilla Gemini-3-Pro on our curated test set.

- Our method consistently outperforms the baseline across all dimensions, achieving gains of +1.4%, +5.0%, +3.1%, and +4.0% in Faithfulness, Conciseness, Readability, and Aesthetics, respectively, resulting in a +4.1% overall improvement.

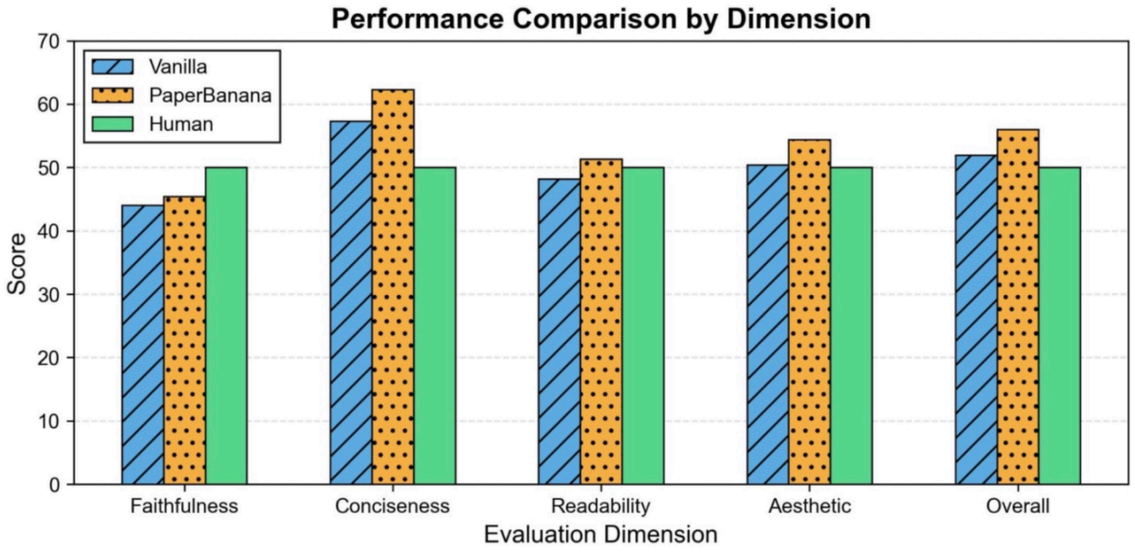


Figure 4 | [Generated by 🍌] Vanilla Gemini-3-Pro vs. PaperBanana for statistical plots generation.

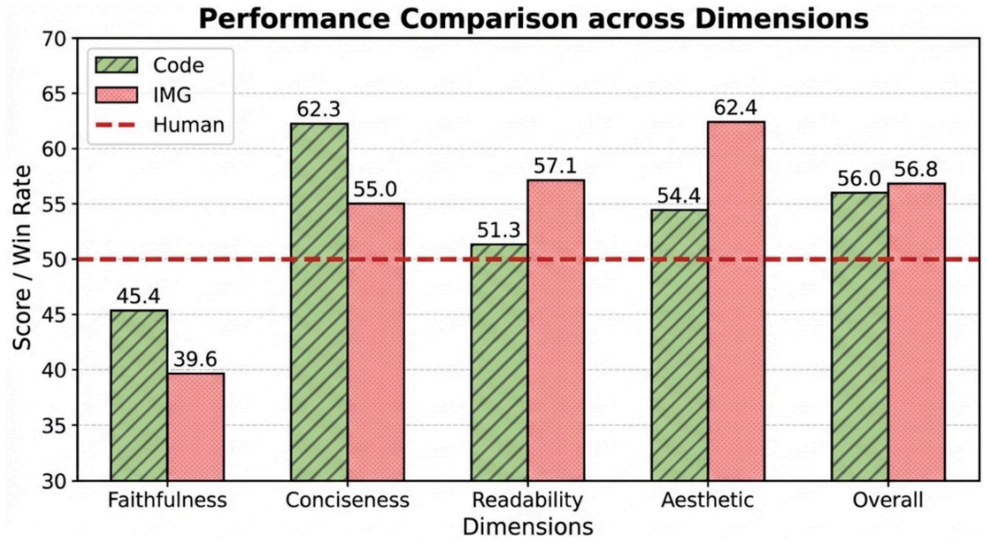


Figure 5 | [Generated by 🍌] Coding vs. Image Generation for visualizing statistical plots.

Experiments

Ablation Study — Contribution of Each Agent

- To understand the contribution of each agent component, we conduct an ablation study, with results presented in Table 2.

Table 2 | Ablation study on PaperBananaBench. The shaded row indicates the default setting of PaperBanana. We systematically ablate each agent component to assess its contribution. The ◯ symbol denotes the Random Retriever which randomly selects 10 examples instead of performing semantic retrieval.

#	Module					Faithfulness ↑	Conciseness ↑	Readability ↑	Aesthetic ↑	Overall ↑
	Retriever	Planner	Stylist	Visualizer	Critic					
①	✓	✓	✓	✓	3 iters	**45.8**	**80.7**	**51.4**	**72.1**	**60.2**
②	✓	✓	✓	✓	1 iter	38.3	75.2	50.6	68.9	51.8
③	✓	✓	✓	✓	-	30.7	79.2	47.0	72.1	45.6
④	✓	✓	-	✓	-	39.2	61.7	47.9	67.4	49.2
⑤	◯	✓	-	✓	-	37.3	62.7	51.1	65.6	48.3
⑥	-	✓	-	✓	-	41.9	58.6	43.1	62.9	44.2

Limitations & Future Directions

1. **Towards Editable Illustrations:** PaperBanana outputs *raster images*, which are difficult to edit post-generation. Future directions include:

   - Image editing models for minor adjustments.
   - Reconstruction pipelines (OCR + segmentation + Python-PPTX) for structural edits.
   - GUI Agents operating professional vector software (e.g., Adobe Illustrator) for fully editable vector outputs.

Limitations & Future Directions

2. **Style Standardization vs. Diversity:** The unified aesthetic guideline ensures professional rigor but reduces stylistic diversity. Future work should explore dynamic style adaptation mechanisms.

3. **Fine-Grained Faithfulness:** The most prevalent errors involve fine-grained connectivity — misaligned start/end points, incorrect arrow directions — which often escape current critic models. Closing this gap requires advancing the fine-grained visual perception capabilities of foundation VLMs.

4. **Advancing Evaluation Paradigms:** Current VLM-as-a-Judge evaluation faces limitations in detecting structural errors and aligning with subjective aesthetic preferences. Future protocols could incorporate structure-based or rubric-based metrics, and train customized reward models.

Thank you for your attention