Benchmarking and Enhancing Text-to-Image Models for Generating Visual Representations in Early Arithmetic Education

Anonymous ACL submission

Abstract

Visual representations are highly effective in early arithmetic education, as they make abstract mathematical symbols more concrete and support the development of numeracy and reasoning skills. However, creating such visuals is labor-intensive for teachers. Thus, we introduce the equation-to-visual generation task and E2V-Bench, a benchmark for generating pedagogically meaningful visuals from arithmetic equations. Developed with insights from primary school math teachers and informed by visual patterns extracted from six educational resources, E2V-Bench comprises 1.5K arithmetic problems spanning four visual types. We also propose new automatic metrics for evaluating generated visuals. A systematic evaluation on E2V-Bench reveals that open-source text-to-image models perform substantially worse than the strongest closed-source models. Building on these findings, we curate a high-quality training dataset and demonstrate that our model adaptation strategies, including rejection sampling fine-tuning, prompt refinement, and regeneration, significantly improve model performance. This work establishes a foundation for studying equation-to-visual generation (a novel reasoning task) and can support teachers in creating visuals for arithmetic education.

1 Introduction

Visuals are essential tools for teaching foundational arithmetic concepts (e.g., addition, subtraction, multiplication and division) in early primary education (Grades 1–3). They help transform abstract mathematical symbols into intuitive representations (Cooper et al., 2018), supporting student understanding and problem solving (Mayer, 2002). For instance, rather than memorizing that "7 + 2 = 9", young learners may benefit from a visual showing seven objects (e.g., apples) combined with two more to make nine, which helps them grasp the meaning of the "+" operation (Mayer, 2002), as illustrated in the bottom right panel of Figure 2.

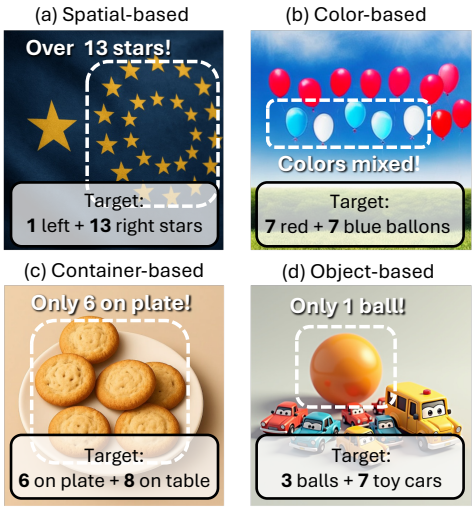

Figure 1: Failure cases of text-to-image models: (a) GPT-Image-1, (b) LMD, (c) Bagel, (d) Show-o2.

However, despite their pedagogical value, producing high-quality instructional visuals remains largely manual and time-consuming (Kaitera and Harmoinen, 2022; Boonen et al., 2016; Xu et al., 2021). Recent work has begun to explore automated generation or retrieval of instructional visuals. Math2Visual (Wang et al., 2025) generates visuals from math word problems, but relies on fixed icon sets and predefined templates, limiting its flexibility in representing standalone arithmetic concepts. Other approaches, such as text-image matching for retrieving visuals for textbook creation (Singh et al., 2023) and text-driven visual question generation (Singh et al., 2019; Luo et al., 2024), focused on retrieval or question construction, rather than generating instructional visuals.

In this work, we focus on primary school level arithmetic education, which deals with learning mathematical operations (addition, subtraction, division, and multiplication) via equations. We formalize the task of equation-to-visual (E2V) and introduce Equation-to-Visual Benchmark (E2V-Bench), a novel benchmark for gener-

ating pedagogically meaningful images from arithmetic equations, together with automatic evaluation metrics for assessing the quality of generated visuals. We manually collected 189 instructional visuals from six educational resources and distilled four recurring visual types that guide our benchmark design. We further conducted interviews with ten primary-school mathematics teachers to refine and validate these design choices.

Using E2V-Bench, we evaluate nine state-of-the-art Text-to-Image (T2I) models across five model families. We show that GPT-Image-1 (OpenAI, 2025), LMD (Lian et al., 2024), Flux.1-dev (Blackforest, 2024), and Bagel (Deng et al., 2025) are top-performing baselines on this task. However, as illustrated in Figure 1, even these strong models exhibit systematic failures in generating pedagogically faithful visuals. Building on these findings, we curate 4,072 high-quality training examples through a generation-and-filtering pipeline from GPT Image-1 and a template-based icon-to-image generation pipeline using the Bagel model. Leveraging this data, we improve the Bagel model through rejection-sampling fine-tuning. In parallel, we adapt prompt-refinement and regeneration techniques for the layout-to-image model LMD, substantially improving its performance. Our contributions are summarized as follows:

- We introduce E2V-Bench, the first exploration on generating pedagogical visuals for arithmetic education, propose automatic metrics and systematically evaluate nine T2I models.
- We release a high-quality training dataset to provide reliable supervision for the E2V task.
- We show that T2I enhancement strategies (i.e., rejection sampling fine-tuning, prompt refinement, and regeneration) yield substantial performance gains on E2V-Bench.

2 Related Work

Visuals in Teaching Primary-School Arithmetic Visuals are widely recognized as beneficial for learning arithmetic skills (Kaitera and Harmoinen, 2022; Boonen et al., 2016). Well-designed visuals make mathematical ideas more accessible (Mayer, 2002; Evagorou et al., 2015; Small and Lin, 2025), foster student engagement (Cooper et al., 2018), and support more efficient learning (Arcavi, 2003). Various pedagogical designs have been proposed to structure arithmetic relationships. For instance, the bar model (Hoven and Garelick, 2007) expresses numerical relations through proportional bars, and has been shown to enhance children's problem-solving ability (Osman et al., 2018) and their use of effective cognitive strategies (Morin et al., 2017). More recently, the Noyon framework depicts mathematical problems with modular iconic components, offering a flexible yet structured approach to visualize mathematical reasoning (Saquib et al., 2021). Collectively, these approaches underscore the importance of carefully designed visuals in arithmetic education.

Automated Visual Generation and Retrieval Despite their benefits, producing instructional visuals still requires considerable effort from teachers (Xu et al., 2021), motivating automation of visual generation and retrieval for educational use. Prior work has explored aligning images with textbook text (Singh et al., 2023), generating image-based multiple-choice questions from semantic representations (Singh et al., 2019), and combining multimodal information to support question generation (Luo et al., 2024). However, these methods generally focus on broad educational applications rather than making the mathematical concepts explicit. Early efforts such as VisualMath (Dwivedi et al., 2017) rendered simple word problems with pre-existing images, but addressed only basic operations (+, -) and lacked pedagogical validation. More recently, Math2Visual (Wang et al., 2025) generates visuals from math word problems with stronger educational grounding, though its reliance on fixed templates limits flexibility in representing standalone arithmetic concepts within equations. These efforts highlight progress toward automation but underscore the absence of benchmarks for generating pedagogically grounded visuals directly from equations.

Text-to-Image Benchmarks Benchmarks for T2I models are typically built from curated prompt sets targeting specific capabilities, with general-purpose benchmarks such as DALL-E 3 Eval (Betker et al., 2023), DrawBench (Saharia et al., 2022), PartiPrompts (Yu et al., 2022), and HEIM (Lee et al., 2023) covering challenges such as multi-object generation, spatial and attribute control, and factual consistency. Other datasets focus on narrower skills, such as alignment (e.g., TIFA (Hu et al., 2023), Gecko (Wiles et al., 2025)) or compositional reasoning (e.g., T2I-CompBench (Huang et al., 2023)). More recently,

Kajic et al. (2024) introduced the GECKONUM to study numerical reasoning in T2I models, focusing on counting-based prompts while omitting broader arithmetic operations and educational grounding. To date, no benchmarks systematically evaluate whether T2I models can generate pedagogically meaningful visuals that faithfully represent arithmetic operations for early arithmetic education.

3 Foundations of Equation-to-Visual Generation

3.1 Task Definition

We define the task of E2V generation as producing visuals that faithfully represent the arithmetic concepts encoded in an equation. Rather than reproducing equations as text, generated visuals should depict quantities through concrete objects and spatial arrangements that support learners' intuitive understanding of arithmetic relationships: a form of visual scaffolding commonly used in early mathematics instruction (Kaitera and Harmoinen, 2022; Boonen et al., 2016).

To ground this task in authentic educational practice, we examined how arithmetic equations are visualized in existing instructional materials. Two researchers manually collected 189 visuals from six widely used educational resources, including three textbooks (Ministry of General Education and Instruction, Republic of South Sudan, 2018; Cotton et al., 2021; Moseley and Rees, 2021) and three online platforms (Accessim, 2025; mathematics monster, 2025; fun2dolabs, 2025). After excluding seven decorative visuals that did not correspond to the equations, we conducted a two-stage thematic analysis on the remaining 182 visuals: an exploratory review of 50 visuals to identify recurring patterns, followed by coding the remaining 132 visuals using the identified categories. This analysis revealed four recurring visual types for representing arithmetic equations:

(1) Color-based: Operands are distinguished by assigning distinct colors to otherwise identical objects (e.g., red vs. green apples). Color serves as the primary grouping cue, a perceptual feature that young children readily use for grouping (Harris et al., 1970).

(2) Object-based: Operands are distinguished by using different object types, with object category serving as the primary grouping cue. Prior work shows that young children can readily organize objects by type (Bornstein and Arterberry, 2010).

(3) Spatial-based: Operands are distinguished by placing object groups in different spatial positions (e.g., left vs. right). Spatial separation serves as the primary grouping cue, which children naturally use to organize visual information (Quinn et al., 2008).

(4) Container-based: Operands are distinguished by placing objects into distinct containers, with each container corresponding to one operand. Prior work shows that children readily reason about containment relations (Casasola et al., 2003), making containers a natural basis for grouping.

Based on this analysis, we define the E2V task as generating accurate visuals in each of the four visual types given a single arithmetic equation. Additional details of the thematic analysis are provided in Appendix A.

3.2 Evaluative Study with Teachers

To assess the pedagogical value of the proposed visual types, we conducted an evaluative study with ten experienced primary school mathematics teachers (Grades 1-3; demographics in Table 8). Teachers reviewed visuals following our four visual types, covering both cartoon style and realistic style designs across the four basic arithmetic operations, and rated their pedagogical usefulness on a 7-point Likert scale. Study details are provided in Appendix D.

The four visual types align with current classroom practice. Teachers consistently reported that the four visual types reflect visuals commonly used in Grades 1-3 classrooms (Table 13). All participants indicated prior experience with similar visuals, suggesting that our designs align well with existing instructional practices. Across evaluation criteria, teachers gave uniformly high ratings to visuals generated using the proposed visual types (Table 9), indicating teachers find visuals following our proposed visual types pedagogically valuable and effective for supporting arithmetic learning.

Visual style preferences. Both realistic and cartoon style visuals were rated as pedagogically useful (Table 11), with realistic scoring slightly higher than cartoon. Teachers noted complementary advantages: realistic visuals support real-world grounding, while cartoon visuals reduce visual distraction. These results suggest that supporting both visual styles is beneficial, allowing teachers to select styles based on instructional goals.

Usefulness of Visual Types Across Arithmetic Operations. Teachers' ratings of each visual type's usefulness in teaching vary by operation (Table 12).

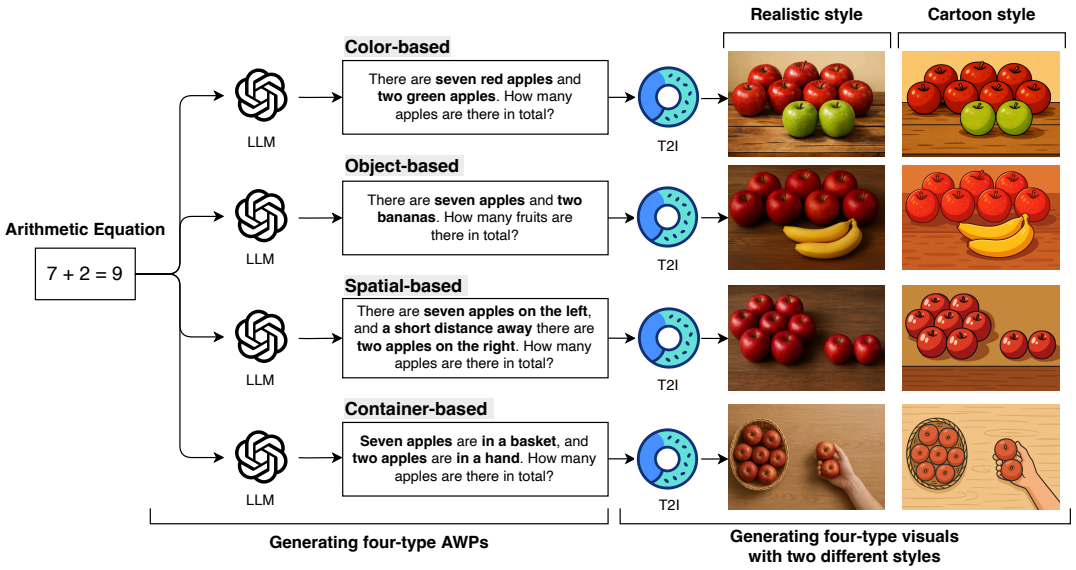

Figure 2: E2V-Bench visual generation pipeline. Equations are systematically generated from four basic arithmetic operations. For each equation, an LLM generates four arithmetic word problems (AWPs), one for each visual type. These AWPs are then used as prompts for T2I models to produce visuals in both realistic and cartoon styles.

Container-based visuals were preferred for multiplication and division, where explicit grouping supports understanding of the underlying operations. Color-based visuals were most often associated with addition, as color cues help distinguish addends while conveying their combination. Spatial-based visuals were frequently cited as helpful for subtraction, where spatial separation supports the notion of removing quantities, and were also considered useful for grouping in multiplication and division. Object-based visuals were viewed as a more cognitively demanding option, better suited for challenging students who have already developed a basic understanding of operations.

4 E2V-Bench: Dataset and Metrics

Although the E2V task is defined over mathematical equations, directly prompting existing T2I models with symbolic expressions is ineffective, as models tend to generate stylized renderings of equations rather than object-based visuals that convey quantity and structure (Appendix C.1). This limitation arises because current T2I models are designed for natural language inputs rather than abstract symbolic expressions. To address this modality gap, we adopt a two-step formulation: we first verbalize each equation into an arithmetic word problem (AWP), mirroring how teachers contextualize equations in practice, and then generate visuals from the resulting AWP. This approach enables the generation of pedagogically meaningful visuals from equations while remaining compatible with standard T2I pipelines.

4.1 Benchmark Construction

We construct E2V-Bench in two phases: equation generation and corresponding AWP generation (Figure 2). We enumerate arithmetic equations over the four basic operations (+, –, ×, ÷), restricting quantities to be below 20 for primary-school suitability, yielding 371 unique equations. Using in-context learning with Gemini-2.5-Flash (Google, 2025), we generate four AWPs per equation, one for each visual type (Container, Object, Color, and Spatial-based), guided by eight manually crafted in-context examples per type.

To validate correctness, two researchers reviewed 120 sampled AWPs (30 per visual type) and confirmed full alignment with the intended equations and visual types. Pedagogical value was further validated by ten primary-school teachers, who evaluated 32 generated AWPs and rated them as suitable for classroom use (Appendix D.3). The final dataset contains 1,484 AWPs, with 1,184 for training and 300 for testing (statistics in Table 5).

4.2 Evaluation Metrics

We evaluate model outputs using two criteria informed by prior work and interviews with ten primary-school mathematics teachers.

Quantity Accuracy measures whether the number of objects in the generated visual matches the quan-

Category	Model	Quantity Acc. (%)		Overall Acc. (%)		CLIPScore	
		Realistic	Cartoon	Realistic	Cartoon	Realistic	Cartoon
Closed-source	Recraft-v3	6.00	5.67	4.67	2.67	0.81	0.78
	GPT-Image-1	**27.00**	**22.33**	**21.33**	**17.00**	0.80	0.81
Diffusion-based	SD-3.5-large	6.67	8.33	3.33	5.00	0.80	0.80
	Flux.1-dev	8.67	9.33	6.33	5.67	0.77	0.75
Layout-to-image	Blueprint	2.67	1.33	1.33	1.00	0.74	0.73
	LMD	22.33	15.33	12.00	9.67	0.75	0.71
Transformer-based	Show-o2	4.33	4.33	2.33	3.00	**0.82**	**0.82**
	Bagel	8.00	11.00	5.67	8.00	0.77	0.79
Prompt refinement	PAE	1.00	3.00	1.00	1.00	0.70	0.69

Table 1: Evaluation of T2I models on E2V-Bench. We report quantity accuracy, overall accuracy, and CLIPScore for both realistic-style and cartoon-style outputs. SD-3.5-large denotes the Stable Diffusion-3.5-large.

tities specified in the AWP. In practice, we adapt CountGD (Amini-Naieni et al., 2024) to automatically count object instances in generated visuals.

Overall Accuracy evaluates whether a visual aligns with the input AWP. Although recent benchmarks use vision-language models as judges, we explicitly avoid them due to their unreliability in exact object counting (Choudhury et al., 2025). To guarantee precise and reproducible evaluation in early arithmetic settings, we adopt a deterministic, rule-based pipeline tailored to each visual type using OpenCV and Scikit-Learn (Appendix C.2).

Both criteria are critical in educational settings, where learning depends on the correctness of instructional information (Metzger et al., 2003; Goldin and Shteingold, 2001). To validate alignment with human judgment, we randomly sampled 216 visuals generated by nine models (Table 1) from the E2V-Bench test set and had two researchers independently annotate each visual. They disagreed on only two cases, resolved through discussion. Cohen's kappa between human annotations and the automatic metrics was 0.96 for both criteria, indicating substantial agreement. To further assess pedagogical validity, we collected expert feedback from ten primary-school teachers during the user study; all teachers agreed that the two criteria adequately capture visual quality for teaching arithmetic skills (details in Appendix D). Following common practice in visual evaluation, we also report CLIPScore (Hessel et al., 2021).

5 Benchmarking T2I Models on E2V-Bench

We evaluate nine T2I models on E2V-Bench, spanning closed-source (GPT-Image-1 (OpenAI, 2025), Recraft-v3 (Recraft, 2025)), diffusion-based (Stable Diffusion-3.5-large (Esser et al., 2024), Flux.1-dev (Blackforest, 2024)), layout-to-image (LMD (Lian et al., 2024), Blueprint (Gani et al., 2024)), transformer-based (Show-o2 (Xie et al., 2025), Bagel (Deng et al., 2025)), and prompt-refinement (PAE (Mo et al., 2024)) models. For each AWP, we generate both cartoon and realistic style visuals. Results are reported in Table 1, with example visuals shown in Figure 13. All results are reported from a single run per model.

Performance Across Model Families. The closed-source GPT-Image-1 model achieves the best performance among the nine models, with an overall accuracy of 21.33% (realistic) and 17.00% (cartoon). The other closed-source model, Recraft-v3, performs substantially worse, with overall accuracy of 4.67% (realistic) and 2.67% (cartoon).

Open-source models exhibit substantial variation across families. Diffusion-based models perform poorly overall: Stable Diffusion-3.5-large achieves low overall accuracy (3.33% realistic, 5.00% cartoon), while Flux.1-dev shows modest improvement, reaching 6.33% (realistic) and 5.67% (cartoon), but still struggles to reliably control object counts. Layout-to-image models show mixed performance. Blueprint performs worst across most metrics, whereas LMD substantially outperforms other open-source models, achieving 12.00% overall accuracy in the realistic setting and 9.67% in the cartoon setting, alongside relatively strong quantity accuracy. These results indicate that explicit layout conditioning provides benefits for numerical grounding. Transformer-based models exhibit variable behavior: Bagel surpasses Show-o2 in both quantity and overall accuracy, achieving 5.67% overall accuracy in the realistic setting and 8.00% in the cartoon setting. Finally, PAE records near-zero overall accuracy, confirming that prompt refinement alone is ineffective at generating images

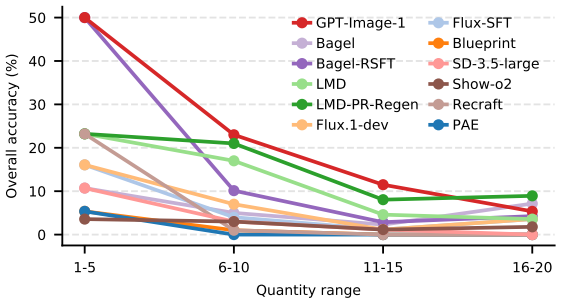

Figure 3: Overall accuracy of realistic-style visuals evaluated across different quantity ranges. Cartoon-style visual results are in Figure 10.

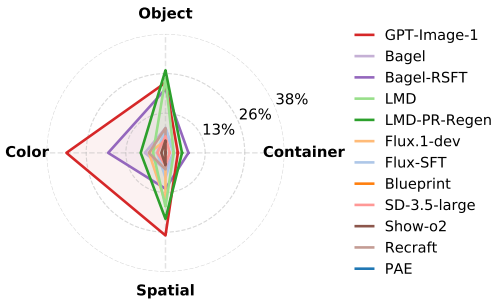

Figure 4: Overall accuracy of realistic-style visuals evaluated across different visual types. Cartoon-style visual results are in Figure 11.

with correct object counts, consistent with prior findings (Cao et al., 2025b). CLIPScores remain comparable across models, indicating similar levels of overall semantic alignment (Hessel et al., 2021).

Variation Across Quantities and Visual Types. Across models, overall accuracy declines sharply as object counts increase (Figure 3), dropping from around 50% for 1–5 objects to below 15% beyond 11 objects. Performance also varies by visual type (Figure 4): GPT-Image-1 shows the most balanced profile, particularly outperforming other models in the color visual type. LMD performs comparably to GPT-Image-1 in the object, spatial, and container visual types. The Bagel emerges as the second-best performer on container and color-based visuals, indicating its potential for further improvement. The LMD-PR-Regen and Bagel-RSFT are enhanced versions of LMD and Bagel; experimental details and analysis are provided in Section 6.

Key Insights and Open-source Potential. Overall, the closed-source GPT-Image-1 delivers the highest-quality outputs but cannot serve as a reproducible research baseline due to its proprietary nature. Among open-source models, LMD, Flux.1-dev, and Bagel stand out as the strongest representatives and demonstrate potential for further improvement. The next section shows how targeted strategies, including rejection-sampling fine-tuning, prompt refinement, and regeneration, can further enhance model performance.

6 Enhancing Equation-to-Visual Generation

As a foundational study of E2V generation, we explore how existing T2I models, such as LMD, Bagel and Flux.1-dev, can be improved through targeted strategies in the task. Considering costs, we report results for the realistic style; however, the same protocols can be applied to cartoon style.

Method	Quant Err. (%)	Bound Err. (%)	Format Err. (%)
LMD	41.67	18.67	2.00
LMD+PR	27.00	22.33	**0.00**
LMD+PR+Regen	**21.00**	**6.67**	**0.00**

Table 2: Error rates for layout generation.

6.1 Layout-to-Image Model

LMD is a representative layout-to-image framework that first generates bounding box layouts in JSON format from text input, and then conditions a diffusion model to produce the final visual (Lian et al., 2024).[1] Given that accurate object counts largely rely on precise layouts (Binyamin et al., 2025), improving the quality of layout generation is essential.

To identify common failure modes in layout generation, two researchers manually reviewed 50 generated layouts and categorized the errors into three recurring types: (1) Quantity error: the number of generated objects does not match the input AWP; (2) Boundary error: bounding boxes extend beyond the image boundary; and (3) Format error: invalid JSON output that cannot be parsed. Grounded on these findings, we developed rule-based scripts to automatically detect each error type. Table 2 reports the frequencies of these errors across 300 samples from the E2V-Bench test set. To address the observed issues of layout generation, we explore two strategies to improve layout generation: prompt refinement and regeneration.

Prompt refinement. We redesigned the prompt instruction to better align the task of generating accurate bounding boxes. Specifically, the revised

[1]All LMD experiments use GPT-4o for layout generation, the state-of-the-art multimodal model at the time (Cao et al., 2025a; Ramachandran et al., 2025).

Method	Quant. Acc. (%)	Overall Acc. (%)	CLIPScore
LMD	22.33	12.00	0.75
LMD-PR	**26.67**	14.67	0.76
LMD-PR-Regen	**26.67**	**15.33**	**0.76**
Flux.1-dev	**8.67**	**6.33**	0.77
Flux.1-dev (SFT)	8.00	5.33	**0.78**
Bagel	8.00	5.67	0.77
Bagel (SFT)	9.00	6.00	**0.78**

Table 3: Evaluation of refined models on realistic-style visual generation. PR means with prompt refinement and Regen means with regeneration.

prompt moves beyond mechanical box placement and instead frames the model as an expert responsible for designing a coherent scene. The prompt guides the model to: (1) imagine a natural background that accommodates all specified objects, (2) prioritize visibility by making the target objects larger and spatially sensible, (3) group similar objects together where appropriate, and (4) ensure realistic placement (e.g., avoiding floating objects, supporting top-down views for crowded settings). In addition, we manually created 15 high-quality in-context examples tailored to E2V, which served as better guidance during layout generation. Original and enhanced prompts are shown in Appendix C.6.

Regeneration. We further introduce an automatic regeneration loop that detects layout errors and re-prompts the model for correction, improving robustness without architectural modifications. We allow up to two regeneration attempts per layout.

6.1.1 Results

Prompt refinement and regeneration lead to substantial improvements in both layout quality and downstream visual accuracy. As shown in Table 2, prompt refinement reduces major layout errors, and adding regeneration further enhances reliability across all error types. These gains are reflected in model performance on the E2V-Bench test set (Table 3), where overall accuracy increases from 12.00% to 14.67% with prompt refinement, and further to 15.33% with the full regeneration pipeline.

6.2 Diffusion and Transformer-based Models

To quantify practical adaptation gains, we evaluate whether supervised fine-tuning on a curated ground-truth set improves a diffusion model (Flux.1-dev) and a transformer-based model (Bagel) without architectural changes.

Training Dataset Construction. We apply a generation-and-filtering pipeline to curate the ground-truth training dataset. Specifically, we generate 5,920 AWPs from equations in the E2V-Bench training split and generate the corresponding visuals using GPT-Image-1, which achieves the highest accuracy in visual generation (Table 1). We then filter the generated images using our automatic metrics and retain only those passing the overall accuracy check, resulting in 1,055 high-quality image–AWP pairs (Table 6).

Supervised Fine-Tuning. We fine-tuned both Flux.1-dev (12B) and BAGEL-MoT (7B) for 10k optimizer steps using supervised fine-tuning. For Flux.1-dev, we employed LoRA adapters for parameter-efficient adaptation, while for Bagel we enabled only the visual generation pathway and froze the VAE using the official training script (Bagel, 2025). Images were generated at 1024×1024 resolution and optimized with AdamW in `bf16` precision. Additional implementation details are provided in Appendix C.7 and C.8.

6.2.1 Results

Table 3 reveals divergent trends: fine-tuning improves the overall accuracy of the transformer-based model Bagel (SFT) from 5.67% to 6.00%, but degrades the overall accuracy of the diffusion-based model Flux.1-dev from 6.33% to 5.33%. We hypothesize that Bagel, an LLM-initialized unified multimodal model trained on interleaved data, retains stronger numerical reasoning capabilities than Flux's purely generative diffusion architecture, consistent with prior findings on the limitations of diffusion models in numerical reasoning (Kajic et al., 2024). Consequently, Bagel appears more robust for visual reasoning.

6.2.2 Further Enhancements to Bagel

To further improve Bagel's performance, we conducted follow-up fine-tuning with synthetic data augmentation and iterative refinement (Figure 9). We first constructed a synthetic training dataset using template-based scripts and icon assets, drawing equations exclusively from the E2V-Bench training split to avoid data leakage. Because these synthetic images differ substantially from Bagel's native generation distribution, directly training on them risks emphasizing stylistic features rather than reasoning. We therefore applied Bagel's image style conversion module to align the synthetic images with Bagel's generation style and, after filtering with our metrics, retained 3,017 images as an extended training set (details in Appendix C.5). Following prior work (Adarsh et al., 2025), we mixed the synthetic

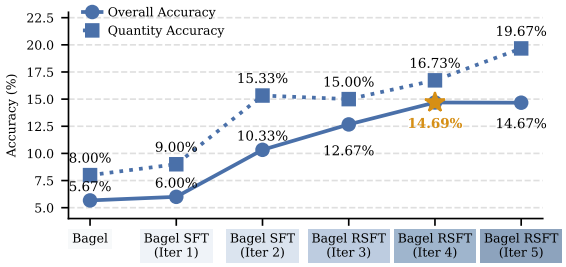

Figure 5: Overall accuracy and quantity accuracy across different training stages of the Bagel model.

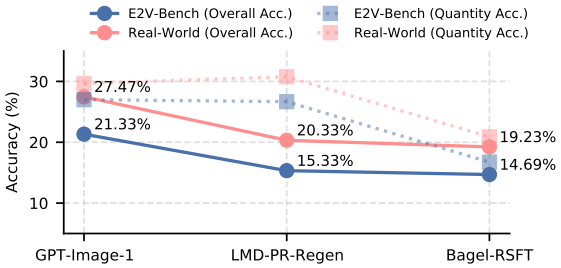

Figure 6: Human evaluation on the real-world educational visual dataset, compared with benchmark results.

and original training data and performed supervised fine-tuning from the previously fine-tuned checkpoint. As shown in Figure 5, this second round of supervised fine-tuning (SFT Iter2) improved overall accuracy from 6.00% to 10.33%.

We further applied rejection-sampling supervised fine-tuning (RSFT), generating ten candidate outputs per training AWP and retaining only samples judged correct by overall accuracy, with at most three visuals preserved per AWP to avoid over-representing easy cases. Each RSFT round was trained for 10k steps, and training stopped once performance no longer improved. As shown in Figure 5, RSFT improves overall accuracy to 12.67% after the first round and peaks at 14.69% after the second. Further RSFT iterations exhibit diminishing returns, potentially due to reduced sample diversity and increased focus on easy cases.

To verify that these gains were not due to additional training iterations, we conducted an ablation where Bagel was only fine-tuned on the original training data without synthetic augmentation. Training to 20k, 30k, and 40k steps yielded overall accuracies of 5.67%, 6.33%, and 7.67%, indicating that the gains stem from synthetic data augmentation and RSFT rather than increased training alone.

6.3 Enhanced Model Analysis and Real-World Validation

Figure 3 and Figure 4 summarize model performance after enhancements. As shown in Figure 3, the enhanced Bagel (Bagel-RSFT) achieves the highest accuracy (50%) in the low-quantity range (1–5 objects), matching GPT-Image-1, but its performance declines rapidly as object counts increase. In contrast, the enhanced LMD (LMD-PR-Regen) attains lower accuracy in low-quantity settings but exhibits more stable performance as quantities grow, suggesting that layout priors help maintain robustness under higher object counts.

Across all visual types (Figure 4), Bagel-RSFT improves over the original Bagel, performing best on container-based and second best on color-based visuals. LMD-PR-Regen improves across all visual types, performing best on object-based and second-best on spatial-based visuals, indicating that layout-based priors are effective for generating correct objects and precise spatial arrangements.

To validate performance in real educational settings, we conducted a human evaluation on a manually collected educational dataset of AWP–visual pairs (§ 3.1), sourced from textbooks and educational platforms, comparing GPT-Image-1, LMD-PR-Regen, and Bagel-RSFT (details in App.C.11). Two researchers independently evaluated 546 visuals by comparing them to ground-truth images and assigning binary scores for quantity and overall accuracy. As shown in Figure 6, model performance trends on this real-world dataset mirror those on E2V-Bench, suggesting that the benchmark effectively captures key visual and pedagogical challenges inherent in authentic classroom scenarios.

7 Conclusion

We introduced the equation-to-visual generation task and E2V-Bench, a benchmark to support the study of generating pedagogically meaningful visuals for arithmetic education. Grounded in insights from primary school math teachers and educational resources, E2V-Bench defines four visual types and includes automatic evaluation metrics. Our experiments reveal a substantial performance gap between open and closed-source text-to-image models on this task, highlighting the challenges of educational visual generation. We further show that this gap can be meaningfully reduced through targeted strategies, including rejection sampling fine-tuning, prompt refinement, and regeneration, supported by a curated high-quality training dataset. Together, these contributions establish a foundation for equation-to-visual generation and outline concrete pathways for improving model performance.

Limitations

Scope of Representation Our method is scoped to early arithmetic education (Grades 1–3), focusing on the four basic operations and quantities below 20. This design choice reflects common classroom practice and enables precise definition of pedagogical correctness. By constraining the scope, we are able to construct a well-grounded benchmark and reliable evaluation framework. Nonetheless, this focus limits direct applicability to more advanced mathematical concepts, such as fractions, ratios, or multi-step equations. Extending equation-to-visual generation beyond early arithmetic remains an important direction for future work.

Coverage of Visual Types Our method models four pedagogically grounded visual types distilled from textbook analysis and validated through teacher feedback. These visual types capture recurring and widely used instructional visual patterns in early arithmetic, enabling the task to be formalized and evaluated in a principled manner. However, they do not exhaust the full space of instructional visual representations. Future work could expand the visual taxonomy to cover a broader range of pedagogical designs.

Focus on Foundational Visual Accuracy Our evaluation focuses on visual correctness, specifically quantity and structural alignment with the intended arithmetic concepts, rather than downstream learning outcomes. This emphasis is deliberate: accurate visual representation is a prerequisite for any instructional use, and our experiments show that current T2I models still exhibit low overall accuracy on this task (with the strongest models achieving around 20% overall accuracy). Given this gap, establishing reliable generation and evaluation of correct visuals is a necessary first step before studying higher-level educational effects. Once model accuracy reaches a reliably high level, future work can incorporate human-centered studies to examine learning outcomes and pedagogical impact.

References

Accessim. 2025. Illustrative Mathematics | v.360 Curriculum — accessim.org. https://accessim.org/. [Accessed 22-12-2025].

Shivam Adarsh, Kumar Shridhar, Caglar Gulcehre, Nicholas Monath, and Mrinmaya Sachan. 2025. SIKeD: Self-guided iterative knowledge distillation for mathematical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 9868–9880, Vienna, Austria. Association for Computational Linguistics.

N. Amini-Naieni, T. Han, and A. Zisserman. 2024. Countgd: Multi-modal open-world counting. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Abraham Arcavi. 2003. The role of visual representations in the learning of mathematics. *Educational studies in mathematics*, 52(3):215–241.

Bagel. 2025. GitHub - ByteDance-Seed/Bagel: Open-source unified multimodal model — github.com. https://github.com/ByteDance-Seed/Bagel.git. [Accessed 20-09-2025].

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. 2023. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8.

Lital Binyamin, Yoad Tewel, Hilit Segev, Eran Hirsch, Royi Rassin, and Gal Chechik. 2025. Make it count: Text-to-image generation with an accurate number of objects. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 13242–13251.

Blackforest. 2024. black-forest-labs/FLUX.1-dev · Hugging Face — huggingface.co. https://huggingface.co/black-forest-labs/FLUX.1-dev. [Accessed 12-02-2025].

Anton JH Boonen, Helen C Reed, Judith Schoonenboom, and Jelle Jolles. 2016. It's not a math lesson–we're learning to draw! teachers' use of visual representations in instructing word problem solving in sixth grade of elementary school. *Frontline Learning Research*, 4(5):55–82.

Marc H Bornstein and Martha E Arterberry. 2010. The development of object categorization in young children: hierarchical inclusiveness, age, perceptual attribute, and group versus individual analyses. *Developmental psychology*, 46(2):350.

Pu Cao, Feng Zhou, Junyi Ji, Qingye Kong, Zhixiang Lv, Mingjian Zhang, Xuekun Zhao, Siqi Wu, Yinghui Lin, Qing Song, et al. 2025a. Preliminary explorations with gpt-4o (mni) native image generation. *arXiv preprint arXiv:2505.05501*.

Yuefan Cao, Xuyang Guo, Jiayan Huo, Yingyu Liang, Zhenmei Shi, Zhao Song, Jiahao Zhang, and Zhen Zhuang. 2025b. Text-to-image diffusion models cannot count, and prompt refinement cannot help. *arXiv preprint arXiv:2503.06884*.

Marianella Casasola, Leslie B Cohen, and Elizabeth Chiarello. 2003. Six-month-old infants' categorization of containment spatial relations. *Child development*, 74(3):679–693.

Monjoy Narayan Choudhury, Junling Wang, Yifan Hou, and Mrinmaya Sachan. 2025. Can vision-language models solve visual math equations? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 10799–10808, Suzhou, China. Association for Computational Linguistics.

Jennifer L Cooper, Pooja G Sidney, and Martha W Alibali. 2018. Who benefits from diagrams and illustrations in math problems? ability and attitudes matter. *Applied Cognitive Psychology*, 32(1):24–38.

Tony Cotton, Caroline Clissold, Linda Glithro, Cherri Moseley, and Janet Rees. 2021. *Oxford International Maths: Student Book 1*. Oxford University Press, Oxford, United Kingdom. Paperback edition.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. 2025. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*.

Utkarsh Dwivedi, Nitendra Rajput, Prasenjit Dey, and Blessin Varkey. 2017. Visualmath: An automated visualization system for understanding math word-problems. In *Companion Proceedings of the 22nd International Conference on Intelligent User Interfaces*, pages 105–108.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. 2024. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*.

Maria Evagorou, Sibel Erduran, and Terhi Mäntylä. 2015. The role of visual representations in scientific practices: from conceptual understanding and knowledge generation to 'seeing'how science works. *International journal of Stem education*, 2:1–13.

flaticon. 2025. flaticon. https://www.flaticon.com. [Accessed 31-12-2025].

fun2dolabs. 2025. fun2dolabs. https://fun2dolabs.com. [Accessed 22-12-2025].

Hanan Gani, Shariq Farooq Bhat, Muzammal Naseer, Salman Khan, and Peter Wonka. 2024. LLM blueprint: Enabling text-to-image generation with complex and detailed prompts. In *The Twelfth International Conference on Learning Representations*.

Gerald Goldin and Nina Shteingold. 2001. Systems of representations and the development of mathematical concepts. *The roles of representation in school mathematics*, 2001:1–23.

Google. 2025. Gemini 2.5 Flash (experimental) | Gemini API | Google AI for Developers — ai.google.dev. https://ai.google.dev/gemini-api/docs/models. [Accessed 05-10-2025].

Lauren Harris, M Joseph Schaller, and Merrill M Mitler. 1970. The effects of stimulus type on performance in a color-form sorting task with preschool, kindergarten, first-grade, and third-grade children. *Child Development*, pages 177–191.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

John Hoven and Barry Garelick. 2007. Singapore math: Simple or complex? *Educational Leadership*, 65(3):28.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. 2023. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20406–20417.

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. 2023. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Susanna Kaitera and Sari Harmoinen. 2022. Developing mathematical problem-solving skills in primary school by using visual representations on heuristics. *LUMAT: International Journal on Math, Science and Technology Education*, 10(2):111–146.

Ivana Kajic, Olivia Wiles, Isabela Albuquerque, Matthias Bauer, Su Wang, Jordi Pont-Tuset, and Aida Nematzadeh. 2024. Evaluating numerical reasoning in text-to-image models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Tony Lee, Michihiro Yasunaga, Chenlin Meng, Yifan Mai, Joon Sung Park, Agrim Gupta, Yunzhi Zhang, Deepak Narayanan, Hannah Teufel, Marco Bellagente, et al. 2023. Holistic evaluation of text-to-image models. *Advances in Neural Information Processing Systems*, 36:69981–70011.

Long Lian, Boyi Li, Adam Yala, and Trevor Darrell. 2024. LLM-grounded diffusion: Enhancing prompt understanding of text-to-image diffusion models with large language models. *Transactions on Machine Learning Research*. Featured Certification.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. 2025. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *Computer Vision – ECCV 2024*, pages 38–55, Cham. Springer Nature Switzerland.

I Loshchilov and F Hutter. 2019. " decoupled weight decay regularization", 7th international conference on learning representations, iclr. *New Orleans, LA, USA, May*, (6-9):2019.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. 2024. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations (ICLR)*.

Haohao Luo, Yang Deng, Ying Shen, See-Kiong Ng, and Tat-Seng Chua. 2024. Chain-of-exemplar: Enhancing distractor generation for multimodal educational question generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7978–7993, Bangkok, Thailand. Association for Computational Linguistics.

mathematics monster. 2025. Mathematics Monster — mathematics-monster.com. https://www.mathematics-monster.com/. [Accessed 22-12-2025].

Richard E. Mayer. 2002. Multimedia learning. volume 41 of *Psychology of Learning and Motivation*, pages 85–139. Academic Press.

Miriam J Metzger, Andrew J Flanagin, and Lara Zwarun. 2003. College student web use, perceptions of information credibility, and verification behavior. *Computers & Education*, 41(3):271–290.

Ministry of General Education and Instruction, Republic of South Sudan. 2018. *Primary Mathematics: Pupil's Book 2*. Mountain Top Publishers Ltd., Nairobi, Kenya. Funded by the Global Partnership for Education.

Wenyi Mo, Tianyu Zhang, Yalong Bai, Bing Su, Ji-Rong Wen, and Qing Yang. 2024. Dynamic prompt optimizing for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26627–26636.

Lisa L Morin, Silvana MR Watson, Peggy Hester, and Sharon Raver. 2017. The use of a bar model drawing to teach word problem solving to students with mathematics difficulties. *Learning Disability Quarterly*, 40(2):91–104.

Cherri Moseley and Janet Rees. 2021. *Cambridge Primary Mathematics Digital Learner's Book 2*, 2nd edition edition. Cambridge University Press, Cambridge, United Kingdom. Digital format.

OpenAI. 2025. OpenAI Platform — platform.openai.com. https://platform.openai.com/docs/models/gpt-image-1. [Accessed 24-09-2025].

opencv. 2025. opencv-python — pypi.org. https://pypi.org/project/opencv-python/. [Accessed 22-12-2025].

Sharifah Osman, Che Nurul Azieana Che Yang, Mohd Salleh Abu, Norulhuda Ismail, Hanifah Jambari, and Jeya Amantha Kumar. 2018. Enhancing students' mathematical problem-solving skills through bar model visualisation technique. *International Electronic Journal of Mathematics Education*, 13(3):273–279.

Prolific. 2025. Prolific | Easily collect high-quality data from real people — prolific.com. https://www.prolific.com/. [Accessed 13-02-2025].

Paul C Quinn, Ramesh S Bhatt, and Angela Hayden. 2008. Young infants readily use proximity to organize visual pattern information. *Acta Psychologica*, 127(2):289–298.

Rahul Ramachandran, Ali Garjani, Roman Bachmann, Andrei Atanov, Oğuzhan Fatih Kar, and Amir Zamir. 2025. How well does gpt-4o understand vision? evaluating multimodal foundation models on standard computer vision tasks. *arXiv preprint arXiv:2507.01955*.

Recraft. 2025. Recraft v3. https://www.recraft.ai/docs#models. [Accessed 29-01-2025].

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494.

Nazmus Saquib, Rubaiat Habib Kazi, Li-yi Wei, Gloria Mark, and Deb Roy. 2021. Constructing embodied algebra by sketching. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–16.

scikit learn. 2025. scikit-learn: machine learning in Python &x2014; scikit-learn 0.16.1 documentation — scikit-learn.org. https://scikit-learn.org/. [Accessed 22-12-2025].

Anjali Singh, Ruhi Sharma Mittal, Shubham Atreja, Mourvi Sharma, Seema Nagar, Prasenjit Dey, and Mohit Jain. 2019. Automatic generation of leveled visual assessments for young learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9713–9720.

Janvijay Singh, Vilém Zouhar, and Mrinmaya Sachan. 2023. Enhancing textbooks with visuals from the web for improved learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11931–11944, Singapore. Association for Computational Linguistics.

Marian Small and Amy Lin. 2025. *Eyes on math: A visual approach to teaching math concepts*. Teachers College Press.

Junling Wang, Anna Rutkiewicz, April Wang, and Mrinmaya Sachan. 2025. Generating pedagogically meaningful visuals for math word problems: A new benchmark and analysis of text-to-image models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 11229–11257, Vienna, Austria. Association for Computational Linguistics.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. 2024. Measuring multimodal mathematical reasoning with math-vision dataset. *arXiv preprint arXiv:2402.14804*.

Olivia Wiles, Chuhan Zhang, Isabela Albuquerque, Ivana Kajic, Su Wang, Emanuele Bugliarello, Yasumasa Onoe, Pinelopi Papalampidi, Ira Ktena, Christopher Knutsen, Cyrus Rashtchian, Anant Nawalgaria, Jordi Pont-Tuset, and Aida Nematzadeh. 2025. Revisiting text-to-image evaluation with gecko: on metrics, prompts, and human rating. In *The Thirteenth International Conference on Learning Representations*.

Jinheng Xie, Zhenheng Yang, and Mike Zheng Shou. 2025. Show-o2: Improved native unified multimodal models. *arXiv preprint arXiv:2506.15564*.

Yi Xu, Roger Smeets, and Rafael Bidarra. 2021. Procedural generation of problems for elementary math education. *International Journal of Serious Games*, 8(2):49–66.

Shih-Ying Yeh, Yu-Guan Hsieh, Zhidong Gao, Bernard BW Yang, Giyeong Oh, and Yanmin Gong. 2023. Navigating text-to-image customization: From lycoris fine-tuning to model evaluation. In *The Twelfth International Conference on Learning Representations*.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. 2022. Scaling autoregressive models for content-rich text-to-image generation. *Transactions on Machine Learning Research*. Featured Certification.

A Details of Thematic Analysis of Visual Types

A.1 Procedure

We conducted a thematic analysis to identify recurring visual types from visuals collected across six educational sources. In total, we collected 189 visuals; seven were decorative and did not correspond to their associated equations, and were therefore excluded, leaving 182 visuals. In the initial phase, we sampled 50 visuals to identify recurring themes. Through iterative discussion, two researchers identified and consolidated four distinct visual types. Coding was performed collaboratively, with a focus on refining the categories through close reading and comparison.

In the systematic evaluation phase, the same two researchers manually analyzed the remaining 132 visuals. This broader analysis allowed us to assess the distribution of problems across the four identified visual types.

A.2 Results

The results of the systematic evaluation are presented in Table 4. The distribution of examples is relatively balanced across spatial-based (54), object-based (50), container-based (49), and color-based (29) visual types, suggesting that the identified visual types can effectively categorize visuals from educational sources.

Category	Spatial	Object	Container	Color
Count	54	50	49	29

Table 4: Distribution of examples across visual types.

B Dataset Details

B.1 Statistics of E2V-Bench Dataset and Training Dataset

The statistics of the E2V-Bench dataset are shown in Table 5, while those of our curated ground-truth training dataset are presented in Table 6.

Category	Sub-category	Train	Test	Total
Visual Type	Container-based	296	75	371
	Spatial-based	296	75	371
	Object-based	296	75	371
	Color-based	296	75	371
Operation	Addition	320	80	400
	Subtraction	608	152	760
	Multiplication	112	28	140
	Division	144	40	184
Quantity Range	1–5	352	56	408
	6–10	300	100	400
	11–15	264	87	353
	16–20	268	57	325
Total		1184	300	1484

Table 5: Statistics of E2V-Bench dataset.

B.2 Comparison of E2V-Bench with Existing Benchmarks

We present the comparison of E2V-Bench with related multimodal benchmarks in Table 7.

Category	Sub-category	Count
Visual Type	Container-based	111
	Spatial-based	271
	Object-based	353
	Color-based	431
Operation	Addition	211
	Subtraction	737
	Multiplication	132
	Division	86
Quantity Range	1–5	628
	6–10	293
	11–15	140
	16–20	105
Total		1166

Table 6: Statistics of curated training dataset.

Figure 7: Example of a visual generated by GPT-Image-1 with the prompt: "Create a cartoon style image to visualize this equation:3+4=7"

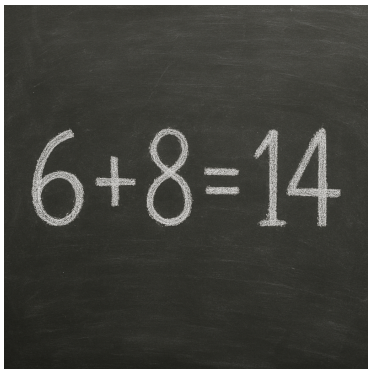

Figure 8: Example of a visual generated by GPT-Image-1 with the prompt: "Create a realistic style image to visualize this equation:6+8=14"

C Experiments Details

C.1 Naive Equation-to-Visual Generation

We first explored a straightforward approach by directly prompting a T2I model to visualize equations. Example outputs are shown in Figure 7 and Figure 8. As illustrated, the generated visuals mainly decorate the equation text with colors or stylistic effects, rather than producing meaningful representations.

C.2 Automatic Metrics

C.2.1 Quantity Accuracy

For quantity accuracy, we use CountGD (Amini-Naieni et al., 2024) to automatically count object instances in generated visuals.

C.2.2 Overall Accuracy

For overall accuracy, we adopt a rule-based evaluation framework tailored to each visual type, implemented using OpenCV (opencv, 2025) and Scikit-Learn (scikit learn, 2025). Overall accuracy is defined at the image level: a generated visual is marked as correct only if it satisfies all task-specific structural and semantic constraints implied by the corresponding equation and visual type.

Across all visual types, we first detect objects using a counting-oriented detection model (CountGD (Amini-Naieni et al., 2024)), producing object counts and bounding boxes that serve as shared inputs to downstream evaluations. Based on the annotated visual type, we then apply specialized verification rules, as described below.

Object-based visuals. For object-based visuals, overall accuracy is equivalent to quantity accuracy. A visual is considered correct if the detected number of each object type exactly matches the ground-truth counts derived from the equation.

Color-based visuals. For color-based visuals, we evaluate both count and color assignment. Using object bounding boxes from the counting stage, we extract object crops and estimate dominant colors in HSV space with Gaussian-weighted aggregation. A visual is marked correct only if (1) the number of detected objects matches the ground truth and (2) the multiset of predicted colors matches the expected color distribution.

Spatial-based visuals. For spatial-based visuals, we verify whether objects are correctly grouped according to spatial relations (e.g., left/right or separated groups). We apply a dual clustering strategy that combines spatial proximity and geometric similarity. Specifically, we first perform agglomerative hierarchical clustering with single linkage on object center coordinates to capture spatial adjacency, which is effective for separated or chain-like layouts. If this strategy fails to recover the expected grouping, we additionally apply k-means cluster-

Benchmark	#Entries	Domain	Task Type	Suitability for Arithmetic Education
Evaluating Numerical Reasoning in T2I (Kajic et al., 2024)	1,386	Numbers, objects	T2I generation	Broad coverage of numerical reasoning, but not designed for early arithmetic education.
Math2Visual (Wang et al., 2025)	1,903	Math word problems	Template-based visual generation	Targets math word problem education but lacks visual diversity due to its heavy reliance on templates and icon datasets.
MathVista (Lu et al., 2024)	6,141	Math + vision	Multimodal VQA	Evaluation on reasoning, not visual generation.
MATHVision (Wang et al., 2024)	3,040	General mathematics, competition-level problems	Visual math problem-solving; multimodal reasoning evaluation	Targets middle–high school competition-level math, not early arithmetic education.
E2V-Bench (Ours)	5,738 (1,484 AWPs + 4,072 AWP–visual pairs + 182 real educational visuals)	Arithmetic equations, arithmetic word problems	Equation-to-Visual generation	Explicitly targets early arithmetic education with diverse, non-template-based visual generation.

Table 7: Comparison of E2V-Bench with related multimodal benchmarks.

ing over object bounding box geometry (width and height) to distinguish groups based on size and position. A visual is considered correct only if the predicted group sizes exactly match the ground-truth grouping specified by the equation.

Container-based visuals. For container-based visuals, we jointly evaluate object–container assignment and count consistency. Container regions are detected using a grounding-based detector GroundingDINO (Liu et al., 2025), and objects are assigned to containers based on spatial overlap and relative position. We use GroundingDINO for container detection because it generalizes better to container-like objects and achieves higher accuracy than CountGD in our preliminary experiments. A visual is marked correct only if each container contains the exact number of objects specified by the equation.

C.3 Prompts for Generating AWP

The prompt for generating object-based AWP:

```
You are a primary school math teacher. Your task
     is to create a simple arithmetic word
    problem that corresponds to a given
    arithmetic equation. The arithmetic word
    problem should use everyday objects that
    young children (Grades 1--3) can easily
    understand.

Each operand in the equation should be
    represented by a different type of object to
     clearly distinguish the groups of numbers.
    Do not use colors or containers as
    distinguishing features; rely only on object
     types.

In addition to the word problem, you must also
    provide:
(1) the total count of each object type
    mentioned

Here are some examples:
...

Now, for the following equation, generate the
    outputs following the same style:

Equation: {equation}

Please respond in the following format:
AWP: <natural language word problem>
object_count: { "<object_name>": <count>, ... }
```

The prompt for generating color-based AWP:

```
You are a primary school math teacher. Your task
     is to create a simple arithmetic word
    problem that corresponds to a given
    arithmetic equation. The arithmetic word
    problem should use everyday objects that
    young children (Grades 1--3) can easily
    understand.

Each operand should be represented by identical
    objects, differentiated only by their colors
    .

In addition to the word problem, you must also
    provide:
(1) the total count of the objects, and
(2) a structured description of how quantities
    are grouped by color.

Here are some examples:
...

Now, for the following equation, generate the
    outputs following the same style:

Equation: {equation}

Please respond in the following format:
AWP: <natural language word problem>
object_count: { "<object_name>": <total_count> }
visual_structure: {
  "<object_name>": {
    "total": <total_count>,
    "<color_1>": <count>,
    "<color_2>": <count>
  }
}
```

The prompt for generating spatial-based AWP:

```
You are a primary school math teacher. Your task
     is to create a simple arithmetic word
    problem that corresponds to a given
    arithmetic equation. The arithmetic word
    problem should use everyday objects that
    young children (Grades 1--3) can easily
    understand.

Each operand should be represented by placing
    groups of objects in distinct spatial
    positions (e.g., left/right, near/far).

In addition to the word problem, you must also
    provide:
(1) the total count of the objects, and
(2) a structured description of how quantities
    are grouped spatially.

Here are some examples:
...

Now, for the following equation, generate the
    outputs following the same style:

Equation: {equation}

Please respond in the following format:
AWP: <natural language word problem>
object_count: { "<object_name>": <total_count> }
visual_structure: {
  "group1": { "total": <count>, "<object_name>":
       <count> },
  "group2": { "total": <count>, "<object_name>":
       <count> }
}
```

The prompt for generating container-based AWP:

```
You are a primary school math teacher. Your task
     is to create a simple arithmetic word
    problem that corresponds to a given
    arithmetic equation. The arithmetic word
    problem should use everyday objects that
    young children (Grades 1--3) can easily
    understand.

Each operand should be represented by placing
    objects into one or more containers.

In addition to the word problem, you must also
    provide:
(1) the total count of each object type, and
(2) a structured description of how objects are
    distributed across containers.

Here are some examples:
...

Now, for the following equation, generate the
    outputs following the same style:

Equation: {equation}

Please respond in the following format:
AWP: <natural language word problem>
object_count: { "<object_name>": <total_count> }
visual_structure: {
  "<container1_name>": { "total": <count>, "<
      object_name>": <count> },
  "<container2_name>": { "total": <count>, "<
      object_name>": <count> }
}
```

C.4 Prompts for Generating Visuals

The prompt for generating realistic style visuals:

```
Create a realistic style image of: <AWP>
```

The prompt for generating cartoon style visuals:

```
Create a cartoon style image of: <AWP>
```

C.5 Synthetic Data Generation and Style Conversion

Synthetic data generation. We constructed a synthetic dataset using a template-based generation pipeline grounded in arithmetic equations. We exclusively used equations from the E2V-Bench *training split* to avoid any data leakage.

Given an equation and its annotated visual type, the pipeline deterministically parses the arithmetic operands and operator, and instantiates a corresponding visual scene using a library of icon assets. This icon library was collected from flaticon (flaticon, 2025) and consists of 20 high-quality PNG icons representing common objects appearing in E2V-Bench. Each visual is composed of one or more "groups", where each group specifies (i) the object type (icon), (ii) the number of instances, and (iii) optional visual attributes such as color or container membership. A dynamic layout algorithm places these groups onto a fixed-size canvas, automatically selecting grid dimensions, icon size, padding, and inter-group spacing to ensure balanced composition across a wide range of quantities.

In parallel, we generate a paired AWP using language templates aligned with the underlying arithmetic operation. For each sample, we also record structured metadata: including object counts, color assignments, container structure, and spatial grouping, which is serialized into the accompanying CSV file. To increase diversity, each equation is rendered multiple times with randomized icon choices, color assignments, and layout configurations.

Motivation for style conversion. The resulting synthetic images are visually clean and icon-centric, which differs substantially from Bagel's native generation distribution. Directly fine-tuning Bagel on these images risks overfitting to superficial stylistic artifacts rather than improving the

model's visual reasoning over quantities and group structure. To mitigate this mismatch, we apply Bagel's image-to-image generation module to convert the synthetic images into Bagel-aligned visual styles before using them for training.

Style conversion with Bagel. We perform style conversion using the original Bagel model in an image-conditioned generation setting. Given a synthetic icon-based image and its corresponding AWP, we prompt the model to regenerate the image in a realistic Bagel style while preserving the underlying semantic structure. Specifically, each input image is passed through Bagel's vision encoder and VAE, and generation is guided by a text prompt of the form: *"Change this image to realistic style: [AWP]"*.

Filtering and final dataset. After style conversion, we filtered the generated images with our automatic metrics. This process resulted in 3,017 high-quality images, which form the final synthetic extension used in our follow-up fine-tuning experiments. By separating structural generation from stylistic alignment, this pipeline allows us to inject large-scale, logically controlled supervision while maintaining compatibility with Bagel's visual generation space.

C.6 Prompt Refinement Details

We present the original prompt and the enhanced prompt for layout generation in LMD.

Original prompt:

```
You are an intelligent bounding box generator. I will provide you with a caption for a photo, image, or painting. Your task is to generate the bounding boxes for the objects mentioned in the caption, along with a background prompt describing the scene. The images are of size 512x512. The top-left corner has coordinate [0, 0]. The bottom-right corner has coordinnate [512, 512]. The bounding boxes should not overlap or go beyond the image boundaries. Each bounding box should be in the format of (object name, [top-left x coordinate, top-left y coordinate, box width, box height]) and should not include more than one object. Do not put objects that are already provided in the bounding boxes into the background prompt. Do not include non-existing or excluded objects in the background prompt. Use "A realistic scene" as the background prompt if no background is given in the prompt. If needed, you can make reasonable guesses. Please refer to the example below for the desired format.

Caption: A realistic image of landscape scene depicting a green car parking on the left of a blue truck, with a red air balloon and a bird in the sky
Objects: [('a green car', [21, 281, 211, 159]), ('a blue truck', [269, 283, 209, 160]), ('a red air balloon', [66, 8, 145, 135]), ('a bird', [296, 42, 143, 100])]
Background prompt: A realistic landscape scene
Negative prompt:
```

Enhanced prompt:

```
You are an expert in drawing bounding box for layout-to-image generation.
        When drawing bounding box, you first think about a reasonable scene that can incorporate all objects, describe the scene in text. Then draw the bounding box for each object, output bounding box and the scene prompt.

        Note:
        1. You can draw limited amount of other objects to make the whole image realistic, but the quantity of objects specified in the prompt should be accurate.
        2. Bounding box should reflect the shape of the object, and the object mentioned in the prompt should be the focus of the image and their bounding box should be BIG for visualization.
        3. If not specified in the prompt, make sure same type of objects are grouping together. If the prompt involve same type of objects in different color, group objects of the same color together.
        4. Please place the bounding boxes in a natural and spatially sensible way --- for example, objects should not be floating in the air. If there are too many objects, you can use a top-down view as indicated in the Background prompt. Similarly, if the objects are inside a container, you may also use a top view to make both the container and the objects visible.
        5. Make sure no bounding box exceeds the image boundary.

        Example:
        1. Input prompt: There are five balloons floating in the air. A short distance away, there are eight green balloons also floating.
        Output Bounding Box:[('balloon', [8, 62, 95, 100]), ('balloon', [115, 62, 96, 102]), ('balloon', [9, 177, 93, 98]), ('balloon', [118, 176, 96, 101]), ('balloon', [14, 293, 97, 97]), ('balloon', [294, 27, 97, 103]), ('balloon', [403, 24, 100, 107]), ('balloon', [291,
```

```
 139, 102, 98]), ('balloon',
[410, 137, 95, 102]), ('balloon',
 [285, 244, 107, 104]), ('balloon
', [405, 244, 97, 105]), ('
balloon', [284, 361, 110, 93]),
('balloon', [407, 357, 100, 96])]
Output Background Prompt: A realistic
     scene
Negative prompt:
```

C.7 Details of SFT on Flux.1-dev

We fine-tuned Flux.1-dev (12B parameters) for 10 epochs with 10 repetitions, using a batch size of 5. This batch size balanced GPU memory constraints (single RTX 4090) and batch diversity, which we found to improve convergence stability in preliminary runs. Gradient checkpointing was enabled to reduce memory consumption and allow deeper model tuning without sacrificing input resolution.

For optimization, we used the AdamW_BF16 optimizer (Loshchilov and Hutter, 2019) with an initial learning rate of 1e-5. Learning rate was decayed with a polynomial scheduler and no warmup, which yielded smoother training dynamics compared to linear decay or cosine schedules. All fine-tuning was conducted with BF16 precision.

To enable parameter-efficient fine-tuning, we adopted LoRA adapters via the Lycoris framework (Yeh et al., 2023), updating only the attention layers instead of full model weights. Images were generated at 1024×1024 resolution to ensure high-quality visuals suitable for educational use cases while remaining computationally feasible. Fine-tuning required approximately 48 hours per run on a single NVIDIA RTX 4090 GPU.

C.8 Details of SFT on Bagel

We fine-tuned BAGEL-MoT (7B parameters) for 10,000 optimizer steps. We used a packed-batch scheme of one sequence per rank targeting 10,240 tokens (flush at 11,520; per-sample cap 10,240) to balance GPU memory limits with sequence diversity observed to aid convergence in preliminary runs. FSDP and non-reentrant activation checkpointing were enabled to reduce memory footprint and permit longer effective contexts without lowering input fidelity.

For optimization, we used AdamW with an initial learning rate of 2×10^{-5} ($\beta_1 = 0.9$, $\beta_2 = 0.95$, $\epsilon = 1 \times 10^{-15}$) and gradient clipping at 1.0. The learning rate followed a constant schedule with 2000 warm-up steps. All fine-tuning was conducted in `bfloat16` precision (`bf16`) using `torch.amp.autocast`.

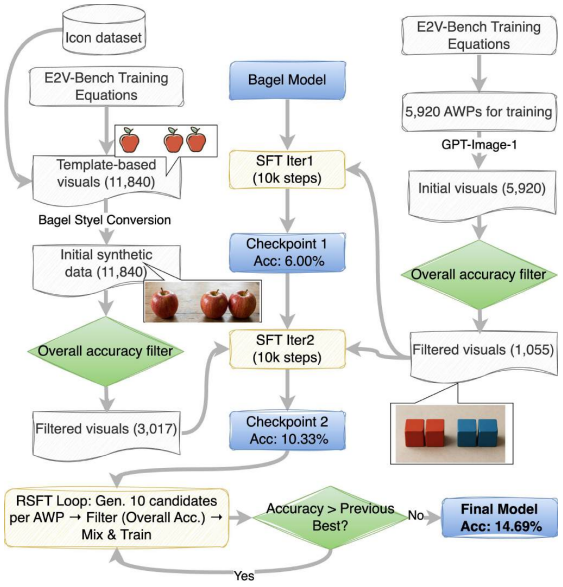

Figure 9: Bagel training pipeline. The process begins with supervised fine-tuning (SFT Iter1) on filtered GPT-Image-1 generated data. Performance is further enhanced through synthetic data augmentation (SFT Iter2) and subsequently refined through iterative rejection-sampling supervised fine-tuning (RSFT).

To limit trainable state, we froze the VAE and enabled only the visual generation pathway (i.e., visual understanding disabled). We fine-tuned from the HuggingFace BAGEL checkpoint[2], loading EMA weights only before training. Fine-tuning required approximately 12 hours per run on a 4 NVIDIA GH200 GPUs.

C.9 Details of Bagel Training

We present the training pipeline of Bagel model in Figure 9.

C.10 Additional Results

Figures 10 and 11 present the overall accuracy of cartoon style visuals generated by different models, evaluated across varying quantity ranges and visual types, respectively.

[2]https://huggingface.co/ByteDance-Seed/BAGEL-7B-MoT

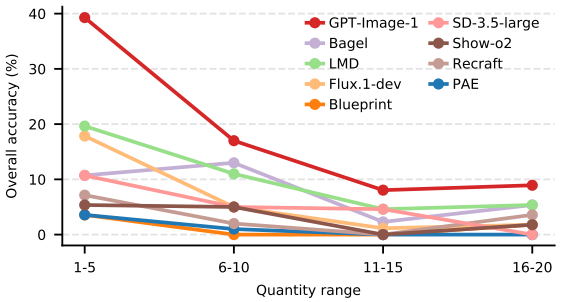

Figure 10: Overall accuracy of cartoon style visuals evaluated across different quantity ranges. Corresponding figure for realistic style visuals are in Figure 3

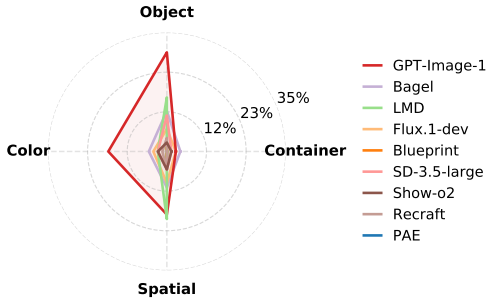

Figure 11: Overall accuracy of cartoon style visuals evaluated across different visual types. Corresponding figure for realistic style visuals are in Figure 4.

C.11 Human Evaluation Details

We conducted a human evaluation on a manually curated educational dataset of AWP–visual pairs, as described in Section 3.1. The dataset was sourced from primary-school mathematics textbooks and online educational platforms.

During manual curation, we collected visuals from arithmetic education sections covering the four basic operations that were accompanied by either an AWP or a symbolic equation. For visuals that were associated only with equations and lacked an explicit AWP, we manually authored a descriptive AWP consistent with the visual content. Among the 182 collected visuals, 52 had corresponding AWPs, while the remaining visuals were originally paired only with equations.

During human evaluation, AWPs were used as input prompts to the models. Two researchers independently evaluated a total of 546 generated visuals by comparing them against the ground-truth images and assigning binary judgments for Quantity Accuracy and Overall Accuracy. A total of three annotation disagreements were identified and resolved through discussion. A screenshot of the annotation interface used in the human evaluation is provided in Figure 12.

C.12 Example Visuals from T2I Models

We present example visuals generated by T2I models on E2V-Bench in Figure 13.

D Teacher Interview Study Details

D.1 Teacher Demographics

We recruited ten primary school math teachers through Prolific (Prolific, 2025) and paid them 16.63 USD per hour, which is adequate given the participants' country of residence. We present the participants' demographics in Table 8. All participating teachers use English as the language of instruction.

PID	Teaching Experience	Age	Gender
1	More than 10 years	46	Male
2	6–10 years	36	Female
3	1–2 years	28	Male
4	More than 10 years	54	Female
5	More than 10 years	34	Female
6	More than 10 years	51	Female
7	More than 10 years	53	Male
8	More than 10 years	42	Female
9	6–10 years	27	Female
10	More than 10 years	53	Female

Table 8: Participants' demographics including teaching experience, age, and gender.

D.2 Study Procedure

Our study obtained ethical approval, and written consent was collected from each participant. The study lasted between 45 minutes and one hour. Participants were first introduced to the background of the study and then completed four sessions, as described below.

In the first session, participants were introduced to the four visual types, accompanied by example visuals for the four arithmetic operations (+, -, ×, ÷). After reviewing each design, they completed a questionnaire to provide feedback.

In the second session, participants were presented with 32 arithmetic problems randomly selected from the E2V-Bench test set. They then completed a questionnaire rating the usefulness of these problems for teaching arithmetic skills.

In the third session, participants were shown examples of realistic and cartoon style visuals. They were asked to indicate their preference between the two styles and explain their reasons.

In the fourth session, participants were introduced to the definitions of our proposed evaluation criteria and asked to provide feedback on them.

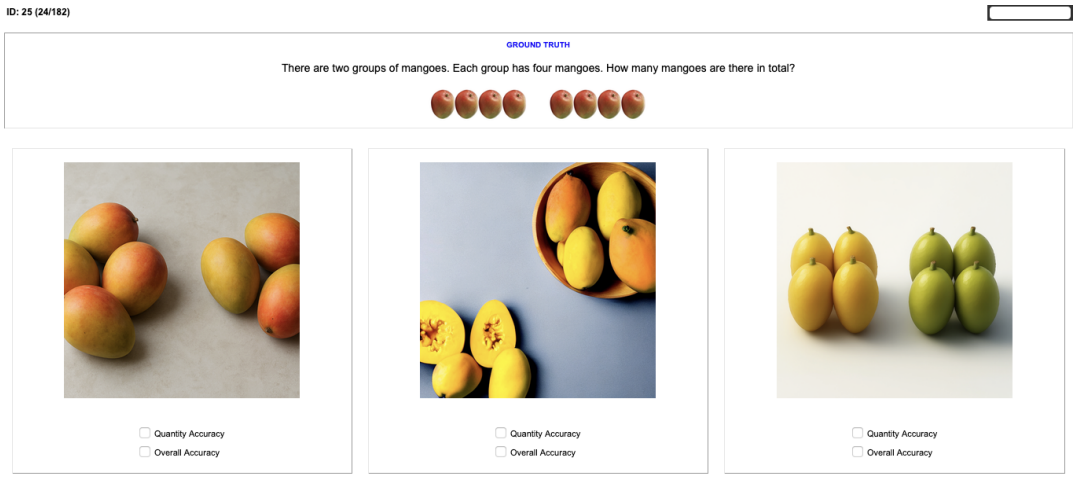

Figure 12: Screenshot of the system used for human evaluation. Annotators assign binary scores for quantity accuracy and overall accuracy: clicking the button indicates a correct judgment, while leaving it unclicked indicates an incorrect one. Visuals are shown in random order for each question.

Finally, after completing all four sessions, participants were asked about their general experiences using visuals to teach arithmetic skills and the types of visuals they typically employ in their classrooms.

D.3 Additional Results

D.3.1 Teachers' Evaluation of Visual Usefulness for Teaching

We present teachers' ratings of the usefulness of images following our proposed four visual types for teaching math equations in Table 9. Scores were collected on a seven-point Likert scale, with seven representing the most positive rating. Teachers consistently gave high ratings across all three statements, with averages close to seven (6.7-6.8) and low variance (0.18-0.46). These results indicate that, from the teachers' perspective, visuals generated with our proposed visual types are useful for teaching and support students' understanding of math equations. Teachers also indicated a strong willingness to use them in their own instruction.

Statement	Average	Variance
Useful for teaching	6.80	0.18
Helpful for student understanding	6.70	0.23
Would like to use in teaching	6.70	0.46

Table 9: Teacher ratings of the usefulness of visuals following the four visual types for teaching arithmetic skills. Scores are on a Likert scale from 1 to 7, where 7 indicates strongly agree.

D.3.2 Teachers' Evaluation of Generated Arithmetic Problems for Teaching

We present teachers' ratings of generated arithmetic problems for teaching in Table 10. Scores were collected on a seven-point Likert scale, with seven representing the most positive rating. Teachers consistently gave high ratings across all three statements, with averages close to seven (6.7-6.9) and low variance (0.10-0.23). These results indicate that, from the teachers' perspective, our generated arithmetic problems are useful for teaching and support students' understanding of math equations. Teachers also indicated a strong willingness to use them in their own instruction.

Statement	Average	Variance
Useful for teaching	6.70	0.23
Helpful for student understanding	6.70	0.23
Would like to use in teaching	6.90	0.10

Table 10: Teacher ratings of the usefulness of generated arithmetic problems following the four visual types for teaching arithmetic skills. Scores are on a Likert scale from 1 to 7, where 7 indicates strongly agree.

D.3.3 Teachers' Preference for Realistic and Cartoon Style Visuals

We present teachers' ratings of generated visuals in realistic and cartoon styles for teaching arithmetic skills in Table 11. Scores were collected on a seven-point Likert scale, with seven representing the most positive rating. Teachers gave high ratings for both styles, with the realistic style (6.3) rated slightly higher than the cartoon style (6.2), and variance

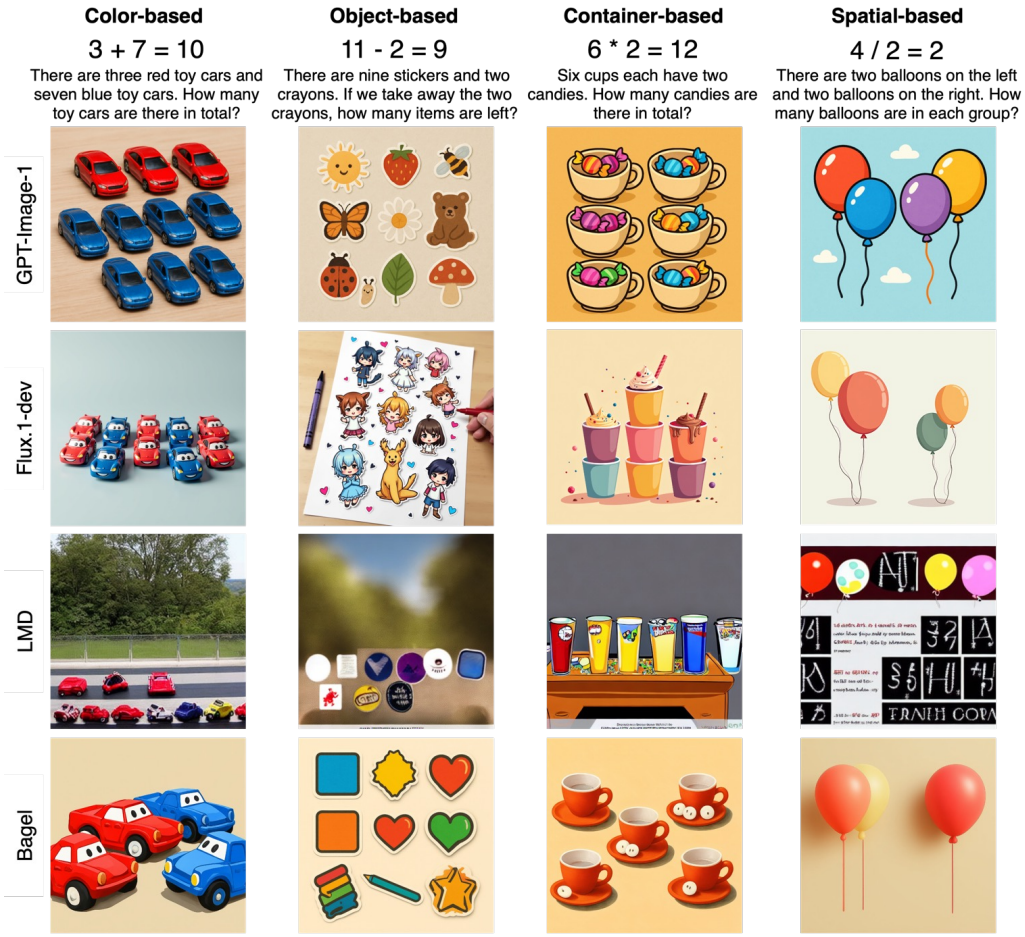

Figure 13: Example visuals from four T2I models on the E2V-Bench task. Each column contains one of the four identified structures (Color-based, Object-based, Container-based, Spatial-based), with the input equation and arithmetic word problem (AWP) description at the top. Each row shows outputs from a different model.

remaining relatively low (0.84–0.90). Out of ten teachers, four gave higher scores to the realistic style. Six teachers also commented that realistic visuals are useful because students can more easily connect them with familiar objects from daily life (P1–4, P6–7), which helps students better understand the context of the problems. Three teachers gave higher scores to the cartoon style. Two teachers noted that cartoon visuals are commonly used in their teaching (P5, P10), while two teachers emphasized that the simplified design of cartoon style can reduce student distractions by using plain backgrounds (P10) and identical objects (P7). These results suggest that both realistic and cartoon visuals are perceived as useful for teaching arithmetic skills, with realistic visuals valued for their connection to students' familiar real-world objects and cartoon visuals appreciated for their clarity and familiarity in educational contexts.

Statement	Average	Variance
Realistic-style visuals are useful for teaching arithmetic skills.	6.30	0.90
Cartoon-style visuals are useful for teaching arithmetic skills.	6.20	0.84

Table 11: Teacher ratings of the usefulness of generated visuals in realistic and cartoon styles for teaching arithmetic skills. Scores are on a Likert scale from 1 to 7, where 7 indicate most useful.

D.3.4 Teacher's Evaluation of Visuals Across Visual Types

We present teachers' ratings of the usefulness of four visual types for teaching arithmetic operations in Table 12. Scores were collected on a seven-point Likert scale, with seven representing the most positive rating.

For addition, teachers rated color-based (6.6) and container-based (6.3) visuals as most useful. Seven teachers noted that color-based visuals effectively illustrate the idea of adding new objects into an

existing group, while seven teachers emphasized that container-based visuals help clearly separate different groups, making it easier to highlight the part being added.

For subtraction, container-based visuals were rated highest (6.1), followed by spatial-based (5.7). Five teachers mentioned that container-based visuals make the subtraction process easier for students to understand. Three teachers also noted that combining visuals with narrative would be helpful for teaching the subtraction process.

For multiplication and division, container-based visuals received the highest ratings (7.0 for multiplication and 6.9 for division), with spatial-based visuals also considered useful (6.1 for multiplication and 6.4 for division). Seven teachers commented that containers are the most effective way to represent groups in multiplication and division, while four teachers mentioned spatial-based visuals are helpful for showing how items can be grouped or partitioned.

Overall, these results suggest that teachers' evaluations of visual types vary by operation. However, container-based visuals were consistently rated highly across all four operations, with particular strength for multiplication and division.

Operation	Color	Object	Spatial	Container
Addition	6.6	5.3	5.3	6.3
Subtraction	5.5	5.2	5.7	6.1
Multiplication	5.5	4.1	6.1	7.0
Division	5.6	5.1	6.4	6.9

Table 12: Teacher ratings of the usefulness of four visual types (color-based, object-based, spatial-based, container-based) for teaching different arithmetic operations. Scores are on a Likert scale from 1 to 7, where 7 indicates most useful.

D.3.5 Coverage of Visual Types and Teachers' Current Teaching Visuals

As shown in Table 13, teachers expressed very strong agreement with the statement that the four visual types capture the kinds of visuals they currently use when teaching arithmetic skills in Grades 1–3 (average = 6.90). The low variance (0.10) indicates a high level of consistency among teachers, suggesting that the proposed visual types closely align with existing classroom practices.

All ten teachers reported that they have used visuals similar to our designs following the four visual types. Six teachers mentioned using cartoon-style visuals similar to our designs, while the other four indicated that they have used realistic-style visuals similar to our designs in their classroom teaching.

Statement	Average	Variance
The four visual types cover the kinds of visuals used to teach arithmetic skills in Grades 1–3.	6.90	0.10

Table 13: Teachers' ratings of the extent to which the four visual types cover visuals used in teaching arithmetic skills in Grades 1–3. Scores are on a Likert scale from 1 to 7, where 7 indicates strongly agree.

E Ethical Consideration and Applications

E.1 Potential Risks

One potential risk is that generated visuals may be misinterpreted if they do not accurately capture the intended mathematical relationships, potentially leading to confusion among students and educators. To mitigate this risk, we collaborated closely with primary school mathematics teachers to develop a structured design space aligned with pedagogical standards. The automatic metrics we propose further help ensure the correctness of the curated dataset. However, this dataset should not be used directly for educational purposes without the supervision of educators.

E.2 Terms of Use

This section outlines the terms and conditions for the use of E2V-Bench. By using the code and datasets in this project, users agree to the following terms:

Prohibited Use The code and datasets shall not be used for commercial purposes without prior written consent from the authors.

Attribution When using or referencing the code and datasets, users must provide proper attribution to the original authors.

No Warranty This project is provided as is without any warranties of any kind, either expressed or implied, including but not limited to fitness for a particular purpose. The authors are not responsible for any damage or loss resulting from the use of this project.

Liability The authors shall not be held liable for any direct, indirect, incidental, special, exemplary,

or consequential damages arising in any way out of the use of the E2V-Bench.

Updates and Changes The authors reserve the right to make changes to the terms of this license or the E2V-Bench itself at any time.

E.3 Compliance with Artifact Usage and Intended Use Specifications

E.3.1 Compliance with Existing Artifact Usage

In our study, we utilized a range of existing artifacts, such as icons from Flaticon (flaticon, 2025) and visuals from six educational sources (Ministry of General Education and Instruction, Republic of South Sudan, 2018; Cotton et al., 2021; Moseley and Rees, 2021; Accessim, 2025; mathematics monster, 2025; fun2dolabs, 2025), to develop our datasets. We rigorously ensured that our usage of these materials was in strict accordance with their intended purposes. Additionally, we employed various computational tools within their prescribed licensing terms, thus adhering to ethical and legal standards.

E.3.2 Specification of Intended Use for Created Artifacts

Our research resulted in two primary artifacts: (1) the E2V-Bench benchmark and evaluation framework, and (2) a curated training dataset for T2I model enhancement.

E2V-Bench: Benchmark and Evaluation Framework

Intended Use: E2V-Bench is intended for academic research on multimodal learning, text-to-image generation, and educational AI. It supports systematic evaluation of models' ability to generate pedagogically meaningful visual representations from arithmetic equations, and facilitates the development and comparison of model adaptation strategies.

Restrictions: E2V-Bench is designed as a research benchmark and should not be used as a standalone instructional system or deployed directly in classroom settings without additional validation. High-stakes educational or commercial use is discouraged unless supported by further empirical studies and ethical review.

Ethical Considerations: The benchmark design is grounded in analyses of publicly available educational materials and validated through consultations with primary school mathematics teachers. Users are encouraged to preserve the pedagogical intent of the visual types and evaluation criteria when extending or applying the benchmark.

Curated Training Dataset for Model Enhancement

Intended Use: The curated training dataset is intended to support research on improving T2I models for equation-to-visual generation. It provides high-quality supervision for studying model adaptation techniques.

Restrictions: The dataset consists of automatically generated and filtered image–text pairs and may still contain inaccuracies or biases inherent to current generative models. It is therefore not recommended for direct instructional use or deployment in educational products without expert oversight.

Data Ethics: All data are synthetically generated or derived from open educational resources and do not contain personally identifiable information. The dataset is released for research purposes, and users are encouraged to adhere to responsible data usage practices consistent with educational and academic norms.

E.4 Data Collection and Anonymization Procedures

The benchmark and training datasets do not contain personal data. Arithmetic equations and corresponding word problems were generated by models, and all visuals were produced using T2I models.

For components involving human participation, including teacher interviews and evaluations, no personally identifiable information was collected or retained. All feedback was anonymized prior to analysis. In addition, we conducted manual screening to exclude offensive, sensitive, or inappropriate content. These procedures were adopted to ensure ethical data handling, participant privacy, and responsible research practice.

E.5 Artifact Documentation

E.5.1 E2V-Bench Benchmark

Domain Coverage E2V-Bench targets early arithmetic education (Grades 1–3), focusing on equation-to-visual generation for foundational arithmetic concepts.

Operation Coverage The benchmark covers four basic arithmetic operations: addition, subtraction, multiplication, and division, with quantities

restricted to values suitable for primary school instruction.

Visual Types Each equation is associated with four pedagogically grounded visual types: container-based, object-based, color-based, and spatial-based representations.

E.5.2 Curated Training Dataset

Visual Style The dataset includes realistic style visuals.

Content Scope All examples are derived from arithmetic equations and corresponding arithmetic word problems generated to support equation-to-visual learning.

Demographic Representation The dataset does not encode demographic attributes. Its educational scope reflects arithmetic instruction practices rather than representations of specific populations.

E.6 Use of AI Assistants in Research

In our study, AI assistants were used sparingly and in accordance with ACL's Policy on AI Writing Assistance. We utilized ChatGPT and Grammarly for basic paraphrasing and grammar checks, respectively. These tools were applied minimally to ensure the authenticity of our work and to adhere strictly to the regulatory standards set by ACL. Our use of these AI tools was focused, responsible, and aimed at supplementing rather than replacing human input and expertise in our research process.

E.7 Instructions Given To Participants

E.7.1 Disclaimer for Annotators

Thank you for participating in our evaluation process. Please read the following important points before you begin:

- **Voluntary Participation:** Your participation is completely voluntary. You have the freedom to withdraw from the task at any time without any consequences.
- **Confidentiality:** All data you will be working with is anonymized and does not contain any personal information. Your responses and scores will also be kept confidential.
- **Risk Disclaimer:** This task does not involve any significant risks. It primarily consists of reading and scoring generated visuals.
- **Queries:** If you have any questions or concerns during the task, please feel free to reach out to us.

E.7.2 Instructions for Experiments

Thank you for participating in our study. This research received ethical approval, and informed consent was obtained from all participants. The study took approximately one hour and consisted of four tasks. Please read the instructions below carefully.

Task 1: Visual Type Evaluation You will be presented with visuals representing arithmetic equations across four visual types (container-based, object-based, color-based, and spatial-based) and the four basic arithmetic operations. For each set of visuals, please complete a questionnaire evaluating its usefulness for teaching arithmetic concepts in the classroom.

Task 2: Arithmetic Word Problem Evaluation You will review 32 automatically generated arithmetic word problems corresponding to equations in our benchmark. Please rate whether each word problem is suitable for classroom use and alignment with instructional practice.

Task 3: Visual Style Comparison You will evaluate two visual styles: realistic and cartoon, for visualizing arithmetic equations. Please provide feedback on how effective each style may be for educational use, including perceived benefits and potential drawbacks in classroom settings.

Task 4: Evaluation Criteria Feedback We will present the definitions of our proposed automatic evaluation metrics for generated visuals. Please comment on whether these criteria adequately capture visual usefulness for teaching arithmetic and provide any suggestions for improvement.

Please answer all questions honestly and feel free to share additional feedback throughout the study. Your responses will help validate our design choices and evaluation framework. Thank you for your time and valuable input.

E.7.3 Data Consent

The data you provide during this study will be used solely for academic research purposes. All information will be anonymized and securely stored, and any published or shared data will be aggregated to ensure your privacy. By participating, you agree to the use of your data as described, but you retain the

right to withdraw your consent at any time without penalty. If you have any questions about how your data will be used, please feel free to ask the research team.