Math2Visual: A Framework for Generating Pedagogically Meaningful Visuals for Teaching Math Word Problems

Anonymous ACL submission

Abstract

Visuals are valuable tools in teaching math word problems (MWPs), helping young learners interpret textual descriptions into mathematical expressions before solving them. However, creating such visuals is labor-intensive and we lack automated methods. In this paper, we present Math2Visual, an automatic framework for generating pedagogically meaningful visuals from MWP text descriptions. Math2Visual leverages a pre-defined visual language and a structured design space for visuals, informed by math teachers, to effectively capture the essential mathematical relationships within MWPs. Using Math2Visual, we construct an annotated dataset of 1,903 visuals and evaluate Text-to-Image (TTI) models in generating visuals that align with our design. We further fine-tune various TTI models with our dataset, demonstrating improvements in educational visual generation. Our work establishes a new benchmark for automated pedagogical meaningful visual generation and offers insights into the challenge of generating multimodal educational content.

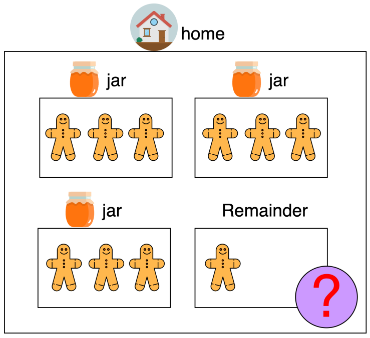

Figure 1: Surplus operation example in Intuitive design (Formal version: Figure 18). MWP: At home, Marian made 10 gingerbread cookies, which she will distribute equally among tiny glass jars. If each jar is to contain 3 cookies, how many cookies will remain unplaced?

1 Introduction

Math word problems (MWPs) are textual descriptions of mathematical scenarios that require interpreting linguistic and numerical information to derive mathematical expressions for problem-solving (Verschaffel et al., 2014). MWPs are a key part of primary school math education and have been the subject of significant research (Verschaffel et al., 2020). Solving MWPs is a complex cognitive task that progresses through several stages: problem understanding, solution planning and solution execution (Opedal et al., 2023; Polya, 2014). It is a challenge for learners to interpret the text and map it to a mental model that captures the described mathematical relationships (Cummins et al., 1988; Stern, 1993), especially for young students (e.g., grades 1–3) who are still developing their reading and comprehension skills (Duke and Block, 2012). Beyond comprehension challenges, recent findings reveal that children's arithmetic skills do not readily transfer between applied and academic contexts (Banerjee et al., 2025), highlighting the need to bridge intuitive experiences with formal instruction. Visuals designed specifically for MWPs can bridge this gap by transforming mathematical concepts into intuitive representations (Cooper et al., 2018), thus supporting student understanding and problem solving (Mayer, 2002).

Although primary school math teachers have long recognized the value of visuals when teaching MWPs (Kaitera and Harmoinen, 2022; Boonen et al., 2016), manually creating these visuals is time-consuming and requires considerable effort (Xu et al., 2021). Current Text-to-Image (TTI) models face limitations in generating visuals that accurately reflect mathematical reasoning (Kajic et al., 2024). In response, recent methods have explored ways for automating the generation or retrieval of instructional images. For instance, Singh et al. introduced a text-image matching task aimed at retrieving and assigning web images to textbook content (Singh et al., 2023). Building on the role of images in learning, another study explored using image semantics to generate visual multiple-choice questions for young learners (Singh et al., 2019). More recently, the Chain-of-Exemplar framework has been applied to generate both multiple-choice questions and their distractors using multimodal educational content that combines text and images (Luo et al., 2024). However, these methods do not generate visuals for narrative contexts such as MWPs. Furthermore, there is no universally accepted visual representation for MWPs that is both pedagogically meaningful and scalable for automated generation.

In response to this gap, we develop a pedagogically meaningful visual design for MWPs along with primary school math teachers. Here, we define pedagogical meaningful visuals as visuals that semantically and logically represent MWPs, helping learners to comprehend them accurately and clearly. Then, we introduce MATH2VISUAL, a framework for generating such visuals in image format from MWP text descriptions. Using MATH2VISUAL, we generate and annotate a dataset containing ∼2K pedagogical visuals for MWPs in grades 1–3. Finally, we evaluate the ability of state-of-the-art TTI models to directly generate visuals that align with our proposed pedagogical design. We use our annotated dataset to fine-tune various TTI models and demonstrate a performance improvement after fine-tuning. In summary, our contributions are:

1. MATH2VISUAL, a scalable framework that incorporates a tree-based visual language and a structured design space to generate pedagogically meaningful visuals from MWP text descriptions.
2. An annotated visual dataset that benchmarks models' ability to generate mathematically reasoned visuals and supports TTI model training.

2 Related Work

Math Word Problems in NLP Math word problems have long been a focus of interest in the NLP community (Roy and Roth, 2015; Kushman et al., 2014; Huang et al., 2017; Amini et al., 2019; Xie and Sun, 2019; Drori et al., 2022), with research primarily aiming to improve computational models' ability to solve MWPs accurately. Approaches such as mapping text to expression trees (Koncel-Kedziorski et al., 2015; Yang et al., 2022; Roy and Roth, 2017) and explicitly modeling arithmetic operations (Mitra and Baral, 2016a; Roy and Roth, 2018) have enhanced machine processing of mathematical expressions in natural language. However, most existing methods focus on producing numerical answers without human-interpretable reasoning, which is essential in educational settings (Opedal et al., 2023; Shridhar et al., 2022). To address this limitation, recent work has explored integrating mental models and human-centered representations into MWP solving. The MathWorld framework (Opedal et al., 2023) represents MWPs using a graph-based semantic formalism aligned with human reasoning. However, it supports only the four basic arithmetic operations, and lacks coverage of "second-order" MWPs.

Visuals in Primary School Math Education Visuals have long been recognized as critical tools in primary school education, particularly in math teaching (Kaitera and Harmoinen, 2022; Boonen et al., 2016). Research indicates that well-designed pedagogical visuals help students grasp abstract concepts more readily (Small and Lin, 2025; Mayer, 2002; Evagorou et al., 2015) while increasing their engagement (Cooper et al., 2018), and improving study efficiency (Arcavi, 2003). Many visual designs have been proposed for primary school math teaching. One common design is bar model (Hoven and Garelick, 2007). The bar model illustrates numerical relationships of math problems through bars representing quantities, enabling visualization of mathematical concepts and operations (Hoven and Garelick, 2007). The bar model has proven to be effective in improving children's problem solving skills (Osman et al., 2018) and their ability to use correct cognitive stategies to solve the problem (Morin et al., 2017). Another modern design is Noyon, which introduces a modular approach to expressing mathematical problems visually (Saquib et al., 2021). Noyon employs iconic elements to construct representations of mathematical concepts, offering a structured yet flexible way to depict mathematical relationships.

Automated Visual Generation and Retrieval in Education Although educational visuals are widely recognized for their benefits and are frequently used by primary school math teachers in instruction (Jitendra and Woodward, 2019; Boonen et al., 2016), the manual creation of such visuals remains a time-consuming and resource-intensive task (Xu et al., 2021). Recent advances in NLP and educational technology have explored automated methods for generating or retrieving visual content. For instance, tasks such as text-image matching have been proposed to assign web images to textbook content (Singh et al., 2023), while other studies have leveraged image semantics to generate visual multiple-choice questions (Singh et al., 2019) and employed frameworks like Chain-of-Exemplar to combine multimodal educational content for question generation (Luo et al., 2024). However, these approaches fail to generate visuals that reveal the underlying mathematical reasoning in MWPs.

3 From MWP to Visual

This section introduces MATH2VISUAL framework. We first present the desiderata for a good visual (Section 3.1), followed by an overview of MATH2VISUAL (Section 3.2). Then, we explain each component of MATH2VISUAL (Section 3.3 to 3.5). Finally, we detail the process of developing visual designs with teachers and the evaluation criteria (Sections 3.6 and 3.7).

3.1 Desiderata for a Good Visual

For visuals aimed at supporting primary school educators and enhancing student understanding of MWPs (targeting grades 1–3), the following criteria are essential: (1) clearly convey the central ideas of an MWP (Evagorou et al., 2015; Jitendra and Woodward, 2019), (2) reduce unnecessary cognitive load of students (Mayer, 2002), and (3) enhance student engagement (Cooper et al., 2018). Rather than focusing on decorative aesthetics, the design should maintain a semantic and logical alignment with the MWP content.

3.2 MATH2VISUAL Framework Overview

Drawing inspiration from the desiderata above, we present an overview of the MATH2VISUAL framework in Figure 2. MATH2VISUAL follows a text-to-semantics-to-visual pipeline, similar to previous visual generation works (Belouadi et al., 2023, 2024). Given an MWP text description (T_{MWP}) and, optionally, a solution formula (F_{solution}), the framework uses an LLM to produce a visual language VL. VL is a semantic visual representation that holds the information needed to generate the visuals (see Section 3.3). The VL is then paired with a manually collected dataset of icons, called SVG and processed by two rendering programs (R_{formal}, $R_{\text{intuitive}}$) to generate two types of visuals: "Formal" (V_{formal}) and "Intuitive" ($V_{\text{intuitive}}$). Details of the visual design and rendering program are presented in Sections 3.4 and 3.5, respectively.

3.3 Semantic Representation of MWP

To bridge the gap between formal mathematical structure and visual expressiveness, we introduce a tree-based Visual Language (VL) specifically tailored for visual generation and clarity.

VL is a tree-based hierarchical language with a structure closely resembling the expression tree (Wang et al., 2018; Zhang et al., 2023) of the problem solution F_{solution}. In the VL, we represent an MWP using three primary components: entity, container, and operation. We illustrate the mapping from an MWP to these VL components using the example in Figure 2. Note that the mapping from an MWP to VL is not strictly deterministic—it requires an intuitive understanding of visualization. Therefore, we use an LLM with in-context learning to perform the conversion. The full procedure is detailed in Section 4.2.

(1) **Entity** is the smallest unit in VL and represents an element to be visualized. For instance, the flower in Figure 2 is an entity. An entity is identified by attributes `entity_name`, `entity_type` and `entity_quantity`. `entity_name` represents the name of the entity as given in the MWP, `entity_type` is entity's category for visualization. In Figure 2, the phrase "colorful flower" from the MWP maps to `entity_name`, while "flower" becomes the `entity_type`. The `entity_quantity` attribute specifies how many entities there are, which are then logically grouped within a container.

(2) **Container** represents the grouping or possession of entities as indicated in the MWP, similar to the definition in (Opedal et al., 2023). For example, in Figure 2, Faye is a container that possesses 88 colorful flowers. A container is identified by attributes `container_name`, `container_type`, `attr_name` and `attr_type`. The `container_name` describes the container's name as stated in the MWP, while `container_type` defines its category for visualization. In Figure 2, "Faye" is the `container_name` and "girl" is the `container_type`. The `attr_name` and `attr_type` are optional attributes that provide additional contextual details of the container.

(3) **Operation** represents mathematical or logical relationships between containers. In addition to basic arithmetic operations such as addition, subtraction, multiplication, and division, we incorporate additional operations including surplus, comparison and unit transformation. These operations enable us to cover 94.4% of grade 1-3 MWPs in

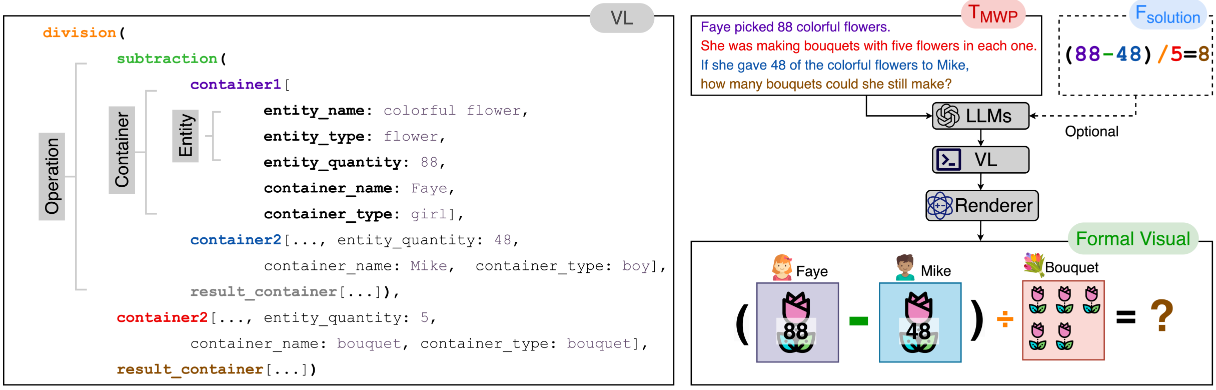

Figure 2: MATH2VISUAL Framework: Our approach first converts the MWP text description into a Visual Language (VL) expression using an LLM. The VL is then passed to a rendering program that generates the corresponding visual. The presented visual is in "Formal" design.

the ASDiv dataset (Miao et al., 2020). Operations are denoted as:

$$\texttt{operation}(\texttt{container1}, \texttt{container2}, \quad \texttt{result\_container}) \tag{1}$$

The **final VL** is a composition of the solution tree $F_{solution}$ and the operations. Thus, container1 and container2 in eq. 1 can themselves be operations, enabling nested operations and supporting hierarchical representations for more complex MWPs. We use identical attributes for container1, container2, and result_container to ensure consistency and ease LLM interpretation. For example, in Figure 2, an inner subtraction operation is performed between container Faye and Mike, and the resulting value is divided by container bouquet through an outer division operation.

3.4 Visual Design

In this section, we describe how elements from a VL are visualized. Our design, informed by an exploratory study with five primary school math teachers (Section 3.6), is inspired by the bar model (Hoven and Garelick, 2007) and Noyon's modular design (Saquib et al., 2021) (Section 2).

Container with Entity: Inspired by Noyon's modular design and the bar model's structure, we depict containers as rectangles enclosing visualized entities. For quantities over ten, a single entity is shown with its number overlaid, consistent with Twinkl datasets (twinkl, 2025). The attributes `container_name` and `container_type` are visualized as a small icon accompanied by text above the container rectangle, as shown in Figure 2. Additionally, if `attr_name` and `attr_type` have non-empty values, they are displayed as the icon alongside the container icon.

Operation: As informed by exploratory study (see Section 3.6), we visualize operations using two visual variations: "Formal" and "Intuitive". The "Formal" variation represents operations using mathematical symbols (e.g. "+", "-", "×", "÷") accompanied by text, as shown in Figure 2. More examples are in Appendix B.1.

In the "Intuitive" variation, each operation is represented through a specific visual arrangement, we present high level description below, with more details in Appendix C.5.

- **Addition:** Containers in the addition operation are enclosed in a large rectangle (see Figure 12).
- **Subtraction**: The minuend container is visualized first, with the subtracted entities crossed out (see Figure 13).
- **Multiplication**: The multiplicand container is repeated to represent multiplication (see Figure 14). For special area computing problems, it is depicted as a single entity with dimensions matching the MWP's width and length (see Figure 15).
- **Division**: The division operation is visualized as the post-division state, with multiple entity rectangles representing groups enclosed within a larger rectangle (see example in Figure 16 and 17).
- **Surplus**: Similar to division, but the surplus entity is visualized separately (e.g., see Figure 1).
- **Comparison**: This operation involves comparing different entities by visualizing them on a balance scale. Each entity is placed on one side of the scale (see example in Figure 19).
- **Unit Transformation**: The unit transformation operation is for questions that involve changes in measurement units. We adopt a purple bubble above each entity to display its value in the transformed unit (see example in Figure 20).

Finally, for MWPs with multiple operations, we follow these visualization rules for each operation and dynamically combine them to form the overall expression tree (see Figure 21).

3.5 From Visual Language to Visual

We convert our Visual Languages (VLs) into visuals using dedicated rendering programs. Each entity in VL is mapped to a visual icon from an SVG dataset, while preserving the operations and relationships between the containers. To achieve this, we convert the VL into a tree structure that captures the hierarchical relationships between operations and containers. We traverse the tree to compute the relative positions of each container in the visual based on its attributes (such as `entity_quantity`) and the layout corresponding to the involved operations (see Section 3.4). The overall process produces a global layout plan for rendering. Finally, we traverse the tree, assigning a corresponding SVG icon for each "type" attribute (`entity_type`, `container_type`, and `attr_type`) and render the complete visual based on the global layout plan. Note that the attributes in `result_container` are only used in "Intuitive" visual generation. The complete algorithm is presented in Algorithm 1.

3.6 Validating designs with Teachers

Co-Designing Visuals with Teachers: We conducted an exploratory study with five experienced primary school math teachers (grades 1–3; demographics in Table 5) who regularly use visuals to teach MWPs. During the study, participants evaluated six alternative visual designs, provided suggestions, and discussed evaluation criteria for our generated visuals. Further details on the designs, study protocol, and results are in Appendix C.

Participants Recognize Our Design's Value for Teaching Our exploratory study results (Tables 6 and 7) confirmed that teachers perceive our visuals as effective in clearly conveying the central ideas of MWP, reducing unnecessary cognitive load, and enhancing student engagement. We asked participants to rate our visual design on a 7-point Likert scale (7 being the highest). Every participant awarded a perfect 7.0 for both "usefulness for teaching" and "likelihood of frequent use in class," and the average score for "helpfulness for student understanding of MWPs" was 6.8. These ratings indicate that our design is pedagogically meaningful.

Suggestions for Refining the Visual Representation Participants highlighted two key insights: the "Formal" design, which incorporates math symbols, best enhances the clarity of mathematical expressions, while the "Intuitive" design best improves student engagement and reduces unnecessary cognitive load. Their feedback on how quantities should be represented led to refinements in our approach. Consequently, our final design offers two variations: "Formal" design emphasizing clarity and "Intuitive" design tailored for engagement and optimization of cognitive load.

3.7 Evaluation Criteria for Generated Visuals

After discussions with five math teachers, we established the following criteria to evaluate our generation approach in reproducing our design.

(i) Accuracy measures how accurately the quantity of entities and relationships between entities in the visual reflect the MWP. This criterion is crucial in education as it is important for students to learn accurate information. (Metzger et al., 2003; Goldin and Shteingold, 2001).

(ii) Completeness evaluates whether all elements necessary for solving the MWP—including entities, quantities, mathematical relationships, and contextual cues that affect problem interpretation—are present in the visual. This criterion is vital in education, as teachers should provide complete and necessary information to learners (Crosby, 2000).

(iii) Clarity measures how easily students can interpret the visual without confusion or ambiguity. This includes clear distinctions between entities, appropriate use of labels and unambiguous spatial arrangements. Clarity is important in math teaching, as it supports effective learning (Metzger et al., 2003; Goldin and Shteingold, 2001).

(iv) Cognitive Load Optimization assesses whether the visual minimizes unnecessary cognitive load caused by distractions or redundant details that do not contribute to problem-solving. Minimizing unnecessary cognitive load is crucial since learners' working memory can process only a few elements at a time (Kirschner, 2002).

4 Visual Dataset Generation

In this section, we describe the process of generating a visual dataset from MWPs.

4.1 MWP Data Source

We select the ASDiv dataset (Miao et al., 2020) as our source of MWPs as it covers a diverse range of problem types and includes grade-level annotations for each question. We collect 1,268 MWPs suitable for our MATH2VISUAL framework, constituting 94.4% of the grade 1–3 MWPs in ASDiv.

4.2 Dataset Creation

In this section, we explain our dataset creation process. First, we manually wrote 30 VL examples that serve as in-context demonstrations for LLMs. Using these examples, we prompt the GPT-4o1 mini model (OpenAI, 2024b) to generate the remaining VL for our collected MWPs. The prompt is shown in Appendix F.1. For each generated VL, we automatically retrieve the entities for visualization and manually collect the corresponding SVG icons of these entities from multiple sources (svgrepoRepoFree, 2025; iconfont, 2025; svgen, 2025; Condino, 2022; YILDIRIM, 2023; pexels, 2025). These SVG icons are then combined with the VL to render a total of 1,903 visuals—comprising 1,268 "Formal" visuals and 635 "Intuitive" visuals. Finally, two researchers manually validate each rendered visual and its associated VL to ensure it accurately represents the corresponding MWP. The process, including SVG collection and manual verification, required approximately 160 hours of dedicated effort. Table 8 provides an overview of our annotated dataset and comparisons with other math pedagogical visual datasets.

5 Results and Analysis

In this section, we aim to address the following experimental questions regarding MATH2VISUAL:
1. How does the choice of generation framework affect the quality of the generated visuals?
2. How does incorporating the solution formula of an MWP impact the generation results?
3. How does fine-tuning on synthesized visual dataset enhance model performance in generating pedagogical meaningful visuals?

5.1 Experiment Design

To assess various strategies for producing visuals, we conduct two sets of experiments: one to evaluate how effectively LLMs generate VL, and another to compare our MATH2VISUAL framework with the latest TTI models for generating visuals.

Evaluating LLMs for Generating Visual Language We create a test set of 257 VL instances using stratified sampling based on "Grade" (e.g., Grade 1) and "Question Type" (e.g., addition) from our annotated dataset. We compare two recent LLMs with strong reasoning capabilities: OpenAI o3-mini (OpenAI, 2025) and Gemini 2.0 Flash (Google, 2025), to see how accurately they can generate VL. We provide both models with prompts in Appendix F.1 and vary whether we include solution formulas in the prompt to test the effect on generation quality. We measure performance by computing: (1) Logic Match Ratio: Ratio of generated VLs correctly match the ground truth VLs from annotation in terms of operations and the quantity of entities. (2) Edit Distance: The average distance between generated and ground-truth VL. We compute this using Zhang–Shasha tree edit distance algorithm (Zhang and Shasha, 1989), implemented through the zss package[1].

Evaluating Methods for Generating Visuals For evaluating different methods of generating visuals, we conduct a two-stage assessment. In the first stage, we perform initial human evaluation using two test sets (Formal and Intuitive), each containing 24 visuals. These visuals are stratifiedly sampled based on "Grade" and "Question Type" from our annotated dataset.

We evaluate two state-of-the-art TTI models, DALLE-3 (OpenAI, 2024a) and Recraft-V3 (Recraft, 2024), using prompts detailed in Appendix F.2 while experimenting with both prompts with and without solution formulas. We compare the visuals generated by DALLE-3 and Recraft-V3 with those rendered from VL generated by o3-mini and Gemini 2.0 Flash, alongside ground-truth visuals from our annotated dataset. Two researchers independently evaluate each visual based on the criteria described in Section 3.7.

To validate results of initial evaluation, we selected the best-performing TTI model and MATH2VISUAL with the best LLM for an expanded human evaluation using the same settings. For this phase, we created two additional test sets (Formal and Intuitive), each with 72 visuals.

5.2 Results of Visual Language Evaluation

In the upper part of Table 1, we show evaluation results for VL generated by various methods. The o3-mini with formula achieves a logic match ratio of 96.89%, indicating close alignment with the ground truth in operations and entity quantities. Additionally, the Gemini-2-flash model with formula records the lowest edit distance, suggesting its generated VL closely matches the ground truth

[1]https://github.com/timtadh/zhang-shasha

attribute values.

Within the same model, including the solution formula reduces the edit distance while increasing the logic match ratio of the generated VL. However, advanced models like o3-mini can still achieve a 91% logic match even without formula.

Criterion	Edit Dist↓	LM Ratio↑
o3-mini(F)	2.82	**96.89**
o3-mini	2.90	91.05
gemini-2-flash(F)	**2.67**	90.27
gemini-2-flash	2.96	72.76
ft_llama-3.1-large(F)	**2.28**	**89.50**
ft_llama-3.1-large	2.52	80.54
zs_llama-3.1-large(F)	4.67	1.95
zs_llama-3.1-large	4.47	3.11

Table 1: Visual Language Generation Results: F indicates generation with the solution formula. Scores are averaged over 257 VL instances per method.

5.3 Results of Visual Evaluation

The upper part of Table 2 shows evaluation results for visuals generated by different methods. Based on these, we selected o3-mini(F) and recraft-v3(F) for the expanded human evaluation, with results in the lower part of Table 2. These results confirm the trends observed in our initial human evaluation. Our key findings are as follows:

MATH2VISUAL Scores Highly on All Criteria The MATH2VISUAL framework, equipped with the latest LLMs, outperforms other TTI models across all criteria, demonstrating its capability to generate accurate visuals aligned with our design. The o3-mini model performs best on the Formal dataset, while the Gemini-2-flash model achieves better results on the Intuitive dataset. The scores for the Formal dataset are consistently higher across criteria compared to the Intuitive dataset. This discrepancy may be due to the Intuitive dataset containing slightly more complex questions for converting to VL. However, the score difference remains relatively small, around 0.4.

Solution Formula Increases Performance Within the same model, including the solution formula as input increases performance in most cases, possibly because it offers a structured representation of the MWP that helps the model understand the mathematical relationships between containers.

5.4 Fine-Tuning for Visual Generation

In this section, we evaluate the effectiveness of fine-tuning LLMs and TTI models using our annotated dataset. We fine-tuned the LLMs using 80% of the annotated data, while for the TTI models, we fine-tuned Formal models with 80% of the Formal data and Intuitive models with 80% of the Intuitive data (details in Appendix G). Specifically, we fine-tuned two LLMs, Llama-3.1-8B (Dubey et al., 2024) and Mistral-7B-v0.3 (Mistral, 2024), as well as two TTI models, Flux.1-dev (Blackforest, 2024) and Stable Diffusion-3.5-large (Esser et al., 2024). For each model, we fine-tuned two versions: one using dataset with solution formula input and one without formula. Results for Llama-3.1-8B are presented in Tables 1 and 2, for Flux.1-dev in Table 2, and for the other models in Tables 9 and 10.

As shown in the lower part of Table 1, the Llama model fine-tuned with formula achieves the lowest edit distance among all models, significantly reducing the edit distance compared to its zero-shot version. It also achieves a logic match ratio comparable to the latest LLMs and higher than that of the model fine-tuned without formula input.

The middle section of Table 2 shows that the visuals generated by MATH2VISUAL with the Llama model fine-tuned with the formula achieve scores comparable to those of the latest LLMs across all criteria. Similarly, the Flux model fine-tuned with the formula performs comparably to the latest TTI models. In every instance, models fine-tuned on datasets with formula outperform those without formula. Our expanded human evaluation (see lower section of Table 2) further validates these findings.

5.5 Qualitative Analysis on TTI Models and Discussion

To identify and understand common errors in visuals generated by TTI models, we performed a qualitative analysis. We use thematic analysis to identify recurring error patterns. This process involved two phases: an initial exploration with 120 visuals to identify error types and then a systematic evaluation of visuals using these categories with 576 visuals generated by three representative methods. The error types include: (1) **Quantity Error**: an incorrect number of entities; (2) **Relation Error**: incorrect mathematical relationships between containers; (3) **Structural Misalignment**: visuals that do not align structurally with our design, featuring misaligned elements or disorganized groupings; (4) **Missing Entities**: visuals missing necessary entities for solving MWP; and (5) **Missing Contextual Cues**: visuals lacking essential contextual cues for solving MWP. Table 3 reports the ratio of each error type, and the findings are discussed below.

	Method	Accuracy		Completeness		Clarity		Cog Load Opt	
		Formal	Intuitive	Formal	Intuitive	Formal	Intuitive	Formal	Intuitive
prompting	o3-mini(F)	**4.92**	4.58	**5.00**	**4.67**	4.88	4.67	**4.96**	4.54
	o3-mini	4.83	4.54	4.96	4.50	**5.00**	4.67	**4.96**	4.42
	gemini-2-flash(F)	4.79	4.50	4.92	4.57	4.96	**4.79**	**4.96**	**4.65**
	gemini-2-flash	4.54	**4.62**	4.58	4.57	4.79	4.67	4.75	4.61
	recraft-v3(F)	3.33	2.96	3.75	3.62	3.75	4.00	3.63	3.96
	recraft-v3	3.26	2.96	3.50	3.33	3.54	3.75	3.54	3.92
	dalle-3(F)	2.96	3.04	3.21	3.42	2.54	2.33	2.54	2.50
	dalle-3	2.79	2.96	2.83	3.33	2.12	2.29	2.17	2.46
fine-tuning	ft_llama-3.1-8B(F)	**4.79**	**4.83**	**4.83**	**4.83**	**4.83**	**4.83**	**4.83**	**4.83**
	ft_llama-3.1-8B	4.58	**4.83**	4.63	**4.83**	4.67	**4.83**	4.67	**4.83**
	zs_llama-3.1-8B(F)	1.25	1.33	1.25	1.33	1.33	1.33	1.29	1.33
	zs_llama-3.1-8B	1.08	1.00	1.04	1.00	1.17	1.00	1.17	1.00
	ft_flux.1-dev(F)	3.21	2.62	3.38	3.38	3.38	3.12	3.50	3.33
	ft_flux.1-dev	3.12	2.21	3.33	3.38	3.33	3.17	3.29	3.25
	zs_flux.1-dev(F)	3.13	2.50	3.21	2.83	3.33	3.25	3.42	3.63
	zs_flux.1-dev	3.13	2.42	3.21	2.83	3.33	3.25	3.42	3.63
exp. eval	o3-mini(F)	**4.97**	**4.96**	**5.00**	**4.97**	4.94	**4.94**	**4.96**	4.96
	recraft-v3(F)	2.65	3.00	3.57	3.82	3.58	3.76	3.18	3.29
	ft_llama-3.1-8B(F)	4.93	4.92	4.92	4.92	**4.99**	4.92	**4.96**	**4.97**
	ft_flux.1-dev(F)	2.49	2.53	2.60	2.64	3.67	3.54	3.89	3.92

Table 2: Human Evaluation of Visual Representations: In the upper and middle parts of the table, 48 visuals (24 Formal, 24 Intuitive) were evaluated with scores averaged from two researchers on a 1–5 scale. In the lower part (expanded evaluation), 144 visuals (72 Formal, 72 Intuitive) were further evaluated with the best performing models. (F) indicates use of the solution formula as input; "ft" denotes a fine-tuned model and "zs" a zero-shot model.

Method	Quantity Err		Relation Err		Struct Misalign		Miss Visual Items		Miss Contex Cues	
	Formal	Intuitive	Formal	Intuitive	Formal	Intuitive	Formal	Intuitive	Formal	Intuitive
ft_flux.1-dev(F)	0.72	0.74	0.85	**0.81**	**0.35**	**0.23**	**0.44**	0.30	0.57	**0.49**
zs_flux.1-dev(F)	0.77	0.78	0.92	0.85	1.00	1.00	0.74	0.66	0.62	0.60
recraft-v3(F)	**0.41**	**0.38**	**0.82**	**0.81**	0.64	0.94	**0.44**	**0.18**	**0.35**	0.50

Table 3: Statistical Results for Qualitative Analysis: For each method, 192 visuals (96 Formal and 96 Intuitive) were evaluated, with each score representing the ratio of corresponding error.

Fine-tuning Improves Structural Alignment and Entities Inclusion Table 3 shows that fine-tuning the Flux model significantly reduces both structural misalignment and missing entity errors compared to the zero-shot model. The fine-tuned model generated visuals align to our design by consistently representing containers as rectangles encompassing entities. In contrast, while the zero-shot version generally represents quantities accurately as numbers, it often fails to properly visualize the corresponding entities. Overall, fine-tuning decreases error rates across all evaluated categories.

Relation Errors Remain a Severe Problem Despite improvements from fine-tuning, all models exhibit high relation error ratio (ranging from .82 to .92 in Formal and .81 to .85 in Intuitive), indicating a persistent challenge in accurately depicting mathematical relationships between containers. We find that visuals generated by TTI models frequently employ incorrect operations or fail to depict the intended relationships. While existing work has explored methods for generating precise numerical quantities in visuals (Binyamin et al., 2024), further research is needed to develop techniques that effectively visualize mathematical relationships.

6 Conclusion

This work introduces MATH2VISUAL, an automatic framework for generating scalable and pedagogically meaningful visuals from MWP text descriptions. MATH2VISUAL leverages a graph-based visual language and a structured visual design space—developed in collaboration with math teachers—to effectively capture the essential mathematical relationships within MWPs. Using MATH2VISUAL, we generated and annotated a dataset of 1,903 visuals and evaluated state-of-the-art Text-to-Image (TTI) models on their ability to produce visuals that align with our design. We further demonstrated that fine-tuning these models on our dataset improves the quality of visual generation. While our results represent a promising step toward the automated generation of pedagogically meaningful visuals, challenges remain in directly generating such visuals with current TTI models. Future work will explore more scalable and flexible generation frameworks and further refine our visual design to better support educational outcomes.

Limitations

(i) Scope of Representation MATH2VISUAL is currently limited to math word problems involving the seven operations defined in this paper (addition, subtraction, multiplication, division, surplus, comparison, unit transformation). Although our framework can handle MWPs that require multiple operations, the solution must be expressible in a single formula. Future work could extend MATH2VISUAL to support more complex problem formulations that require multiple interconnected equations.

(ii) Language Restriction Our study focuses solely on MWPs written in English. While MATH2VISUAL should, in principle, be applicable to similar problems in other languages, adapting the system for multilingual support remains an avenue for future exploration.

(iii) Predefined Visual Style and Input Requirements Despite achieving 94.4% coverage of grade 1-3 MWPs in the ASDiv dataset (Miao et al., 2020), MATH2VISUAL relies on a predefined visual style and requires a SVG dataset of entity icons as input. Although this controlled approach ensures the pedagogical validity of visuals and is an effective strategy given current model capabilities, it inherently limits generation flexibility. Future research may explore more versatile frameworks, such as adapting advanced Text-to-Image models, to generate pedagogically valuable visuals without predefined styles.

References

Aida Amini, Saadia Gabriel, Shanchuan Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. MathQA: Towards interpretable math word problem solving with operation-based formalisms. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2357–2367, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Abhijit V Banerjee, Swati Bhattacharjee, Raghabendra Chattopadhyay, Esther Duflo, Alejandro J Ganimian, Kailash Rajah, and Elizabeth S Spelke. 2025. Children's arithmetic skills do not transfer between applied and academic mathematics. *Nature*, pages 1–9.

Jonas Belouadi, Anne Lauscher, and Steffen Eger. 2023. Automatikz: Text-guided synthesis of scientific vector graphics with tikz. *arXiv preprint arXiv:2310.00367*.

Jonas Belouadi, Simone Paolo Ponzetto, and Steffen Eger. 2024. DeTikZify: Synthesizing graphics programs for scientific figures and sketches with TikZ. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Lital Binyamin, Yoad Tewel, Hilit Segev, Eran Hirsch, Royi Rassin, and Gal Chechik. 2024. Make it count: Text-to-image generation with an accurate number of objects. *arXiv preprint arXiv:2406.10210*.

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.

Victor Condino. 2022. SVG Icons — kaggle.com. `https://www.kaggle.com/datasets/victorcondino/svgicons`. [Accessed 13-02-2025].

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.

Joy Crosby, RM Harden. 2000. Amee guide no 20: The good teacher is more than a lecturer-the twelve roles of the teacher. *Medical teacher*, 22(4):334–347.

Denise Dellarosa Cummins, Walter Kintsch, Kurt Reusser, and Rhonda Weimer. 1988. The role of understanding in solving word problems. *Cognitive Psychology*, 20(4):405–438.

Iddo Drori, Sarah Zhang, Reece Shuttleworth, Leonard Tang, Albert Lu, Elizabeth Ke, Kevin Liu, Linda Chen, Sunny Tran, Newman Cheng, Roman Wang, Nikhil Singh, Taylor L. Patti, Jayson Lynch, Avi Shporer, Nakul Verma, Eugene Wu, and Gilbert Strang. 2022. A neural network solves, explains, and generates university math problems by program synthesis and few-shot learning at human level. *Proceedings of the National Academy of Sciences*, 119(32):e2123433119.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Nell K Duke and Meghan K Block. 2012. Improving reading in the primary grades. *The Future of Children*, pages 55–72.

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.

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 2.0 Flash (experimental) | Gemini API | Google AI for Developers — ai.google.dev. https://ai.google.dev/gemini-api/docs/models/gemini-v2. [Accessed 05-02-2025].

Mohammad Javad Hosseini, Hannaneh Hajishirzi, Oren Etzioni, and Nate Kushman. 2014. Learning to solve arithmetic word problems with verb categorization. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 523–533.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Danqing Huang, Shuming Shi, Chin-Yew Lin, and Jian Yin. 2017. Learning fine-grained expressions to solve math word problems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 805–814, Copenhagen, Denmark. Association for Computational Linguistics.

iconfont. 2025. iconfont.cn. https://www.iconfont.cn/. [Accessed 12-01-2025].

Asha K. Jitendra and John Woodward. 2019. Chapter 11 - the role of visual representations in mathematical word problems. In David C. Geary, Daniel B. Berch, and Kathleen Mann Koepke, editors, *Cognitive Foundations for Improving Mathematical Learning*, volume 5 of *Mathematical Cognition and Learning*, pages 269–294. Academic Press.

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*.

Paul A. Kirschner. 2002. Cognitive load theory: implications of cognitive load theory on the design of learning. *Learning and Instruction*, 12(1):1–10.

Rik Koncel-Kedziorski, Hannaneh Hajishirzi, Ashish Sabharwal, Oren Etzioni, and Siena Dumas Ang. 2015. Parsing algebraic word problems into equations. *Transactions of the Association for Computational Linguistics*, 3:585–597.

Nate Kushman, Yoav Artzi, Luke Zettlemoyer, and Regina Barzilay. 2014. Learning to automatically solve algebra word problems. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 271–281, Baltimore, Maryland. Association for Computational Linguistics.

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.

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.

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.

Shen-yun Miao, Chao-Chun Liang, and Keh-Yih Su. 2020. A diverse corpus for evaluating and developing English math word problem solvers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 975–984, Online. Association for Computational Linguistics.

Mistral. 2024. mistralai/Mistral-7B-v0.3 · Hugging Face — huggingface.co. https://huggingface.co/mistralai/Mistral-7B-v0.3. [Accessed 12-02-2025].

Arindam Mitra and Chitta Baral. 2016a. Learning to use formulas to solve simple arithmetic problems. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2144–2153, Berlin, Germany. Association for Computational Linguistics.

Arindam Mitra and Chitta Baral. 2016b. Learning to use formulas to solve simple arithmetic problems. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2144–2153.

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.

Andreas Opedal, Niklas Stoehr, Abulhair Saparov, and Mrinmaya Sachan. 2023. World models for math story problems. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9088–9115, Toronto, Canada. Association for Computational Linguistics.

OpenAI. 2024a. dalle-3. https://openai.com/index/dall-e-3/. [Accessed 29-01-2025].

OpenAI. 2024b. gpt4o1-mini. https://platform.openai.com/docs/models#o1. [Accessed 28-01-2025].

OpenAI. 2025. o3-mini. https://openai.com/index/openai-o3-mini/. [Accessed 05-02-2025].

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.

pexels. 2025. pexels — pexels.com. https://www.pexels.com/. [Accessed 12-01-2025].

George Polya. 2014. *How to solve it: A new aspect of mathematical method*, volume 34. Princeton university press.

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

Recraft. 2024. Recraft v3. https://www.recraft.ai/docs#models. [Accessed 29-01-2025].

Subhro Roy and Dan Roth. 2015. Solving general arithmetic word problems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1743–1752, Lisbon, Portugal. Association for Computational Linguistics.

Subhro Roy and Dan Roth. 2017. Unit dependency graph and its application to arithmetic word problem solving. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 3082–3088. AAAI Press.

Subhro Roy and Dan Roth. 2018. Mapping to declarative knowledge for word problem solving. *Transactions of the Association for Computational Linguistics*, 6:159–172.

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.

Kumar Shridhar, Jakub Macina, Mennatallah El-Assady, Tanmay Sinha, Manu Kapur, and Mrinmaya Sachan. 2022. Automatic generation of socratic subquestions for teaching math word problems. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4136–4149, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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.

Elsbeth Stern. 1993. What makes certain arithmetic word problems involving the comparison of sets so difficult for children? *Journal of educational psychology*, 85(1):7.

svgen. 2025. svgen-500k. umuthopeyildirim/svgen-500k. [Accessed 12-01-2025].

svgrepoRepoFree. 2025. SVG Repo - Free SVG Vectors and Icons — svgrepo.com. https://www.svgrepo.com/. [Accessed 12-01-2025].

twinkl. 2025. twinkl.ch. https://www.twinkl.ch/resource/t-w-35749-numbers-0-50-on-lions. [Accessed 25-01-2025].

Lieven Verschaffel, Fien Depaepe, and Wim Van Dooren. 2014. *Word Problems in Mathematics Education*, pages 641–645. Springer Netherlands, Dordrecht.

Lieven Verschaffel, Stanislaw Schukajlow, Jon Star, and Wim Van Dooren. 2020. Word problems in mathematics education: A survey. *Zdm*, 52:1–16.

Junling Wang, Jakub Macina, Nico Daheim, Sankalan Pal Chowdhury, and Mrinmaya Sachan. 2024. Book2Dial: Generating teacher student interactions from textbooks for cost-effective development of educational chatbots. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9707–9731, Bangkok, Thailand. Association for Computational Linguistics.

Lei Wang, Yan Wang, Deng Cai, Dongxiang Zhang, and Xiaojiang Liu. 2018. Translating a math word problem to a expression tree. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1064–1069, Brussels, Belgium. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Zhipeng Xie and Shichao Sun. 2019. A goal-driven tree-structured neural model for math word problems. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5299–5305. International Joint Conferences on Artificial Intelligence Organization.

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.

Zhicheng Yang, Jinghui Qin, Jiaqi Chen, Liang Lin, and Xiaodan Liang. 2022. LogicSolver: Towards interpretable math word problem solving with logical prompt-enhanced learning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 1–13, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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*.

Umut Hope YILDIRIM. 2023. umuthopeyildirim/svgen-500k · Datasets at Hugging Face — huggingface.co. `https://huggingface.co/datasets/umuthopeyildirim/svgen-500k`. [Accessed 13-02-2025].

Kaizhong Zhang and Dennis Shasha. 1989. Simple fast algorithms for the editing distance between trees and related problems. *SIAM journal on computing*, 18(6):1245–1262.

Wenqi Zhang, Yongliang Shen, Qingpeng Nong, Zeqi Tan, Yanna Ma, and Weiming Lu. 2023. An expression tree decoding strategy for mathematical equation generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 439–456, Singapore. Association for Computational Linguistics.

A Visual Language Details

A.1 Example of Visual Language

In the MWP description "Jake picked up three apples in the morning..." the container1 could be specified as `entity_name: apple`, `entity_type: apple`, `entity_quantity: 3`, `container_name: Jake`, `container_type: boy`, `attr_name: morning`, `attr_type: morning`. These additional attributes are not fixed and may vary according to different interpretations.

A.2 Comparison of Visual Language with Other MWP Works

We show the comparison of our Visual Language with other MWP works in Table 4.

B Example of Visuals

B.1 Example of Formal Visual

We provide examples of "Formal" visuals in Figures 3 to 11.

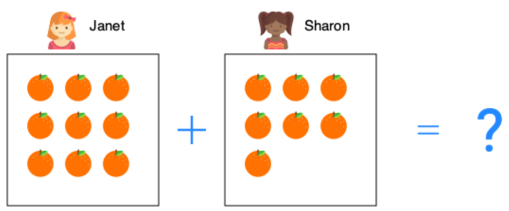

Figure 3: Example of addition operation in Formal design (Intuitive version: Figure 12). Corresponding MWP: Janet has nine oranges, and Sharon has seven oranges. How many oranges do Janet and Sharon have together?

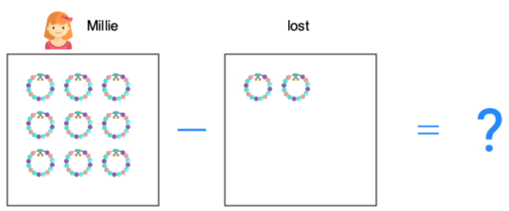

Figure 4: Example of subtraction operation in Formal design (Intuitive version: Figure 13). Corresponding MWP: Millie had 9 bracelets. She lost 2 of them. How many bracelets does Millie have left?

Work	Arithmetic Coverage	Conceptual Coverage	Semantic Granularity	Problem Depth
Visual Language (ours)	(+, -, ×, ÷, surplus, >,<)	Transfer, Rate, Comparison, Part-whole, Surplus, Unit Transformation, Multiple Steps	Concepts & equations	multiple-order MWP
(Opedal et al., 2023)	(+, -, ×, ÷)	Transfer, Rate, Comparison, Part-whole	World model	first-order MWP
(Hosseini et al., 2014)	(+, -)	Transfer	World model	first-order MWP
(Mitra and Baral, 2016b)	(+, -)	Transfer, Comparison (add), Part-whole	Concepts & equations	first-order MWP
(Roy and Roth, 2018)	(+, -, ×, ÷)	Transfer, Rate, Comparison, Part-whole, Concepts & equations	Concepts & equations	multiple-order MWP

Table 4: Comparison of our Visual Language approach with existing MWP methods.

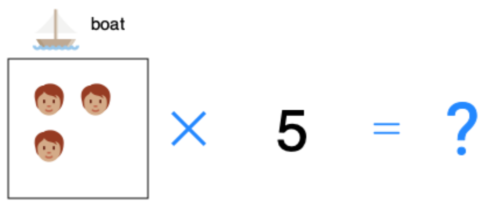

Figure 5: Example of multiplication operation in Formal design (Intuitive version: Figure 14). Corresponding MWP: 5 boats are in the lake. Each boat has 3 people. How many people are on boats in the lake?

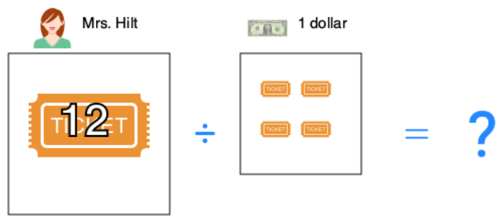

Figure 7: Example of division operation in Formal design (Intuitive version: Figure 17). Corresponding MWP: Mrs. Hilt bought carnival tickets. The tickets cost $1 for 4 tickets. If Mrs. Hilt bought 12 tickets, how much did she pay?

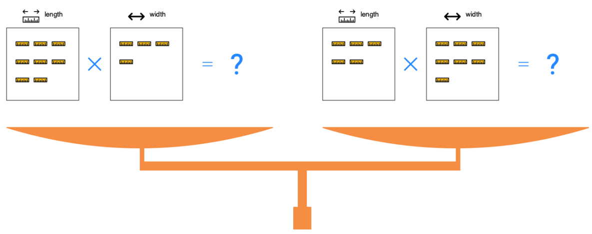

Figure 6: Example of area operation (a special type of multiplication operation) in Formal design (Intuitive version: Figure 15). We use the ruler icon to represent measurement units like feet, meters, etc. Corresponding MWP: Rug A is 8 feet by 4 feet, and Rug B is 5 feet by 7 feet. Which rug should Mrs. Hilt buy if she wants the rug with the biggest area?

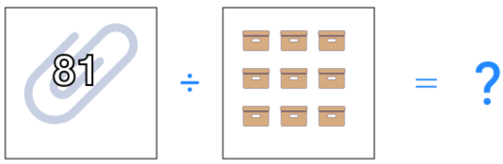

Figure 8: Example of division operation in Formal design (Intuitive version: Figure 16). It represents visuals of a division operation in an MWP, asking for the quantity per group. Corresponding MWP: Lexie's younger brother helped pick up all the paper clips in Lexie's room. He was able to collect 81 paper clips. If he wants to distribute the paper clips in 9 boxes, how many paper clips will each box contain?

B.2 Example of Intuitive Visual

and examples of "Intuitive" visuals in Figures 12 to 21.

C Details of Exploration Study

C.1 Participants' Demographics

We recruited primary school math teachers through Prolific (Prolific, 2025) and paid them 15 USD per hour, which is adequate given the participants' country of residence. We present the participants' demographics in Table 5.

C.2 Study Protocol

Our study obtained ethical approval and collected consent forms from each participant. During the study, participants were first introduced to the background of the study. They then completed four ses-

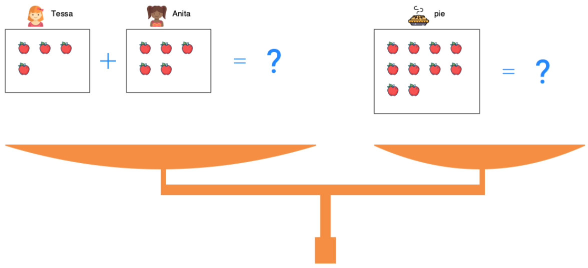

Figure 9: Example of comparison operation in Formal design (Intuitive version: Figure 19). Corresponding MWP: Tessa has 4 apples. Anita gave her 5 more. She needs 10 apples to make a pie. Does she have enough to make a pie?

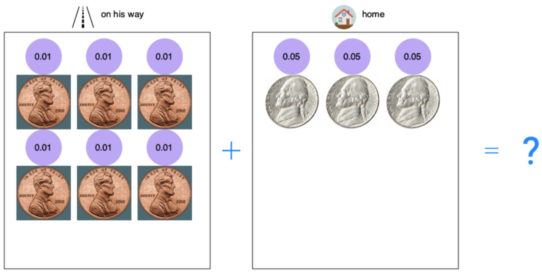

Figure 10: Example of unit transformation operation in Formal design (Intuitive version: Figure 20). Corresponding MWP: Charles found 6 pennies on his way to school. He also had 3 nickels already at home. How much money does he now have in all?

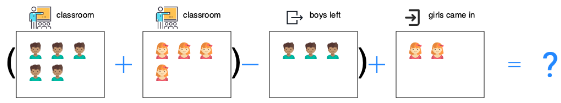

Figure 11: Example of multiple steps operation in Formal design (Intuitive version: Figure 21). Corresponding MWP: There are 5 boys and 4 girls in a classroom. After 3 boys left the classroom, another 2 girls came in the classroom. How many children were there in the classroom in the end?

sions, as described below. The entire study ranged from 1.5h to 2h.

In the first session, participants were asked to indicate their preference between two visual approaches: (1) using multiple visuals, where each visual represents one sentence of the MWP, or (2) using a single visual to represent the entire MWP.

In the second session, we presented six design variations to the participants. These variations differed based on two design choices:

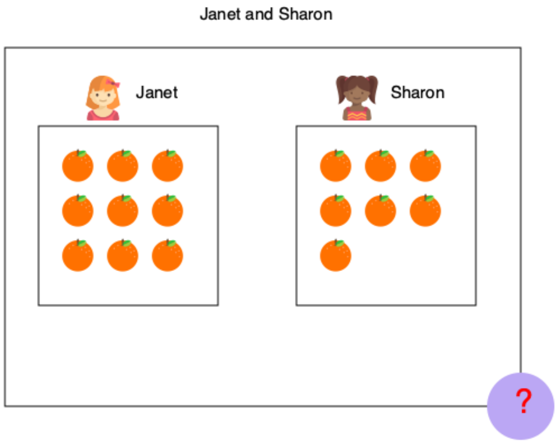

Figure 12: Example of addition operation in Intuitive design (Formal version: Figure 3). Corresponding MWP: Janet has nine oranges and Sharon has seven oranges. How many oranges do Janet and Sharon have together?

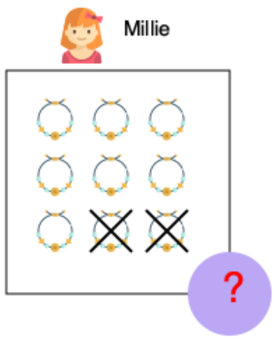

Figure 13: Example of subtraction operation in Intuitive design (Formal version: Figure 4). Corresponding MWP: Millie had 9 bracelets. She lost 2 of them. How many bracelets does Millie have left?

1. How Quantities Are Visualized:
   - **Abstract**: Quantities are represented as text from the MWP.
   - **Hybrid**: A single item is visualized with a label at the bottom-right corner indicating its quantity.
   - **Visual**: Items are directly drawn in quantities matching their number.
2. How Operations Are Visualized:

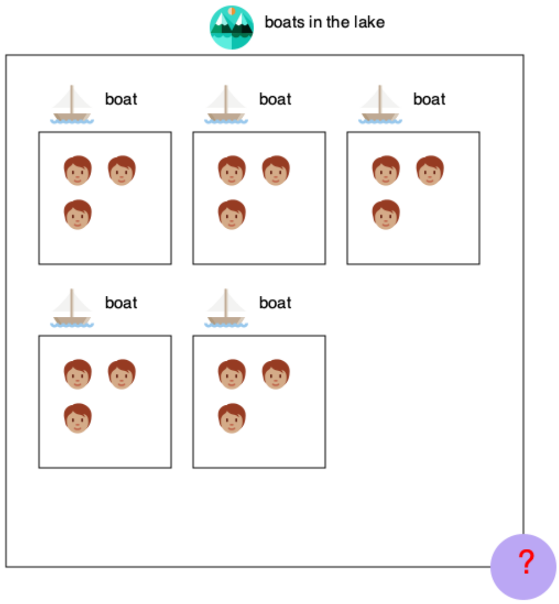

Figure 14: Example of multiplication operation in Intuitive design (Formal version: Figure 5). Corresponding MWP: 5 boats are in the lake. Each boat has 3 people. How many people are on boats in the lake?

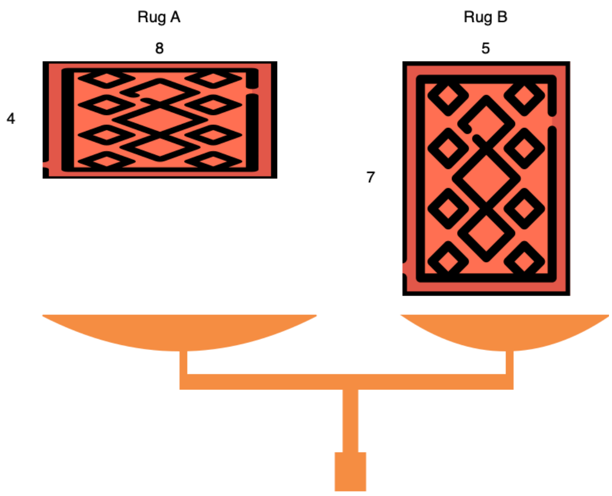

Figure 15: Example of area operation (a special type of multiplication operation) in Intuitive design (Formal version: Figure 6). Corresponding MWP: Rug A is 8 feet by 4 feet, and Rug B is 5 feet by 7 feet. Which rug should Mrs. Hilt buy if she wants the rug with the biggest area?

Figure 16: Example of division operation in Intuitive design (Formal version: Figure 8). It represents visuals of a division operation in an MWP, asking for the quantity per group. Corresponding MWP: Lexie's younger brother helped pick up all the paper clips in Lexie's room. He was able to collect 81 paper clips. If he wants to distribute the paper clips in 9 boxes, how many paper clips will each box contain?

- **Formal**: Mathematical operations are represented using standard symbols (e.g., +, -, ×, ÷).
- **Intuitive**: Operations are visualized using specific arrangements for each operation, as described in Section 3.4.

By combining the three approaches for quantities and the two approaches for operations, we created six unique design variations. Each variation was introduced to the participants and their feedback was sought based on the following criteria: (1) **Clarity**: The extent to which the visual design clearly represents the math word problem. (2) **Engagement**: Whether the visual design helps improve student engagement. (3) **Cognitive Load**: Whether the visual design avoids introducing unnecessary cognitive load for students.

We asked participants to complete a questionnaire after reviewing each design and collected their suggestions for improving the respective design variations. We randomized the presentation order of the design variations to minimize order effects.

In the third session, we aimed to gather feedback

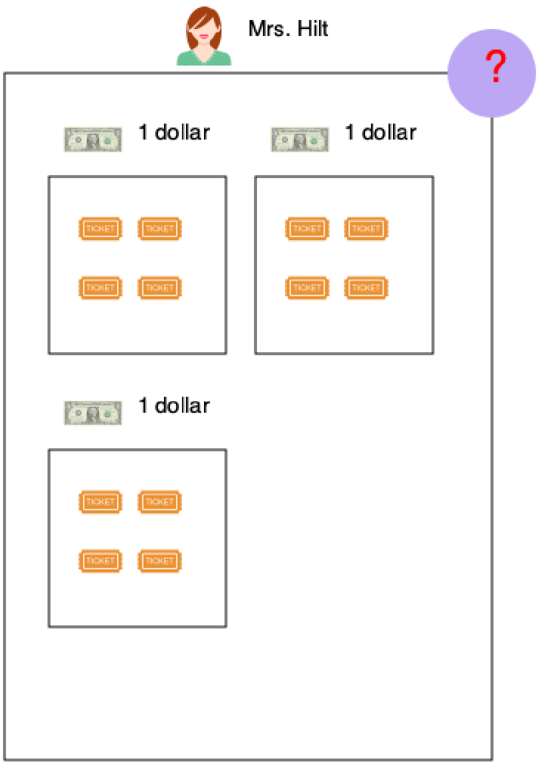

Figure 17: Example of division operation in Intuitive design (Formal version: Figure 7). It represents visuals of a division operation in an MWP, asking for the number of groups. Corresponding MWP: Mrs. Hilt bought carnival tickets. The tickets cost $1 for 4 tickets. If Mrs. Hilt bought 12 tickets, how much did she pay?

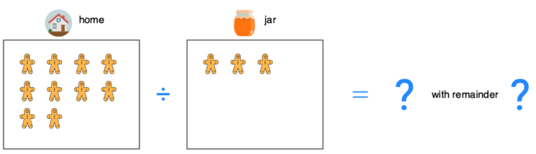

Figure 18: Example of surplus operation in Intuitive design (Intuitive version: Figure 1). Corresponding MWP: At home, Marian made 10 gingerbread cookies which she will distribute equally in tiny glass jars. If each jar is to contain 3 cookies each, how many cookies will not be placed in a jar?

on our "Intuitive" design, which visualizes different operations. The design details are presented in Section 3.4. We used the same criteria as in session two and asked participants to complete a questionnaire after reviewing each operation design, collecting their suggestions for improvement. We also randomized the presentation sequence of

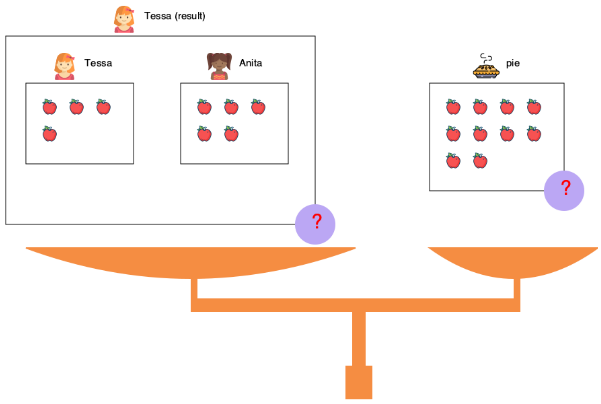

Figure 19: Example of comparison operation in Intuitive design (Formal version: Figure 9). Corresponding MWP: Tessa has 4 apples. Anita gave her 5 more. She needs 10 apples to make a pie. Does she have enough to make a pie?

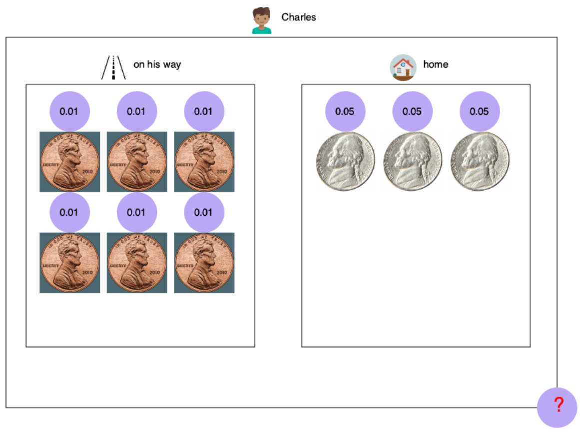

Figure 20: Example of unit transformation operation in Intuitive design (Formal version: Figure 10). Corresponding MWP: Charles found 6 pennies on his way to school. He also had 3 nickels already at home. How much money does he now have in all?

designs in session three.

In session four, we discussed with each participant the criteria to use for analyzing the subsequently generated visuals. This evaluation focused not on the design itself but on how effectively our generation approach could reproduce the intended design. After completing all the sessions, we asked participants to complete a post-task questionnaire assessing the pedagogical value of our visual design. The results are presented in Section 3.6.

C.3 Additional Results

Single Visual is Preferred for Clarity and Simplicity Most of the participants (4) prefered the

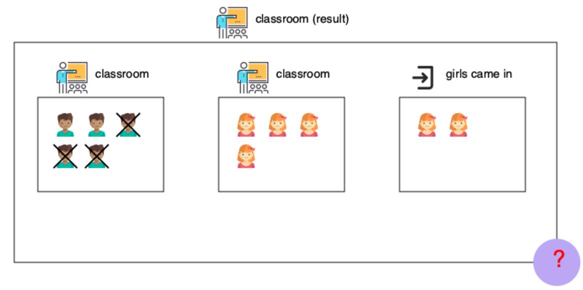

Figure 21: Example of multiple steps operation in Intuitive design (Formal version: Figure 11). Corresponding MWP: There are 5 boys and 4 girls in a classroom. After 3 boys left the classroom, another 2 girls came in the classroom. How many children were there in the classroom in the end?

PID	Language in Teaching	Age	Gender
1	English	52	Male
2	English	45	Male
3	English	35	Female
4	English	44	Female
5	English	37	Female

Table 5: Participants' Demographics: We recruited five primary school math teachers who teach grades 1–3 through Prolific. All teachers consider themselves experienced educators in using visuals to teach MWPs.

single visual design than multiple visual per MWP. They mentioned single visual have better clarity and is explicit enough for simple MWP for grade 1-3 students.

Paticipants' Suggestions on Design Decisions The results of study session two are presented in Table 6. Participants noted that the use of math symbols in the 'Formal' design enhances clarity, while the 'Visual' and 'Intuitive' designs increase engagement and reduce unnecessary cognitive load. However, they also mentioned that the purple circle with a quantity inside caused confusion for learners. They recommended displaying the quantity directly on the visual item and reserving the circle exclusively for question marks. Based on participants' feedback, we refined the designs and developed the final version, which includes two variations: the 'Formal' design using math symbols and the 'Intuitive' design featuring specific arrangements for different operations. More details about our final design are provided in Section 3.4.

Design	Clarity	Engagement	Cog Load Opt
AF	**5.0**	5.4	4.6
AI	3.6	4.6	3.0
HF	3.0	3.6	1.6
HI	2.0	4.2	1.4
VF	4.6	5.6	3.4
VI	4.8	**6.0**	**5.2**

Table 6: Results of exploratory study session 2. In "Design" column, "A" represents "Abstract"; "H" means "Hybrid"; "V" means "Visual"; "F" means "Formal"; "I" means Intuitive. Different combinations reflect different designs, which we discuss in Appendix C.3. All scores are on a 7-point Likert scale, where higher values indicate better performance. Four participants indicated that, after slight modifications to the question mark, the clarity score of the AI design would be 7.

Participants Satisfied with the Intuitive Design The results of study session three are presented in Table 7. Overall, participants expressed satisfaction with the current "Intuitive" design for different operations, with scores ranging from 4.8 to 7 across various criteria. They suggested that using a balance scale to represent comparison problems could further enhance engagement and reduce cognitive load. Additionally, they recommended including less text in the visuals to minimize cognitive load for learners. Our final design incorporates these suggestions, as detailed in Section 3.4.

Operation	Clarity	Engagement	Cog Load Opt
Addition	5.4	6.4	5.0
Subtraction	5.0	6.4	5.2
Multiplication	**7.0**	6.4	**6.0**
Division	6.4	**6.6**	5.4
Surplus	6.8	**6.6**	5.8
Comparison	5.6	6.0	4.8
UnitTrans	6.6	**6.6**	5.6
MultiSteps	6.6	**6.6**	5.8

Table 7: Results of exploratory study session 3. They reflect experts' evaluations of the "Intuitive" design for different operations. All scores are on a 7-point Likert scale, where higher values indicate better performance.

Potential Application of Our Visuals Participants suggested several potential applications for our visuals. They noted that our visuals can be easily attached to slides or textbooks and help with the following:

- **Facilitating MWP Understanding:** Four teachers mentioned that displaying our visuals in class can help students, especially those with learning difficulties, better access MWPs and build confidence in solving them.

- **Enhancing Student Engagement:** Two teachers suggested using the visuals interactively—pointing to different entities in visuals and asking students to link them to corresponding parts of the MWP—can enhance engagement and learning.
- **Teaching Mathematical Operations:** All five teachers agreed that the Intuitive design aids in teaching operations by representing them intuitively, thereby making abstract operations concrete and easier to understand.

C.4 Details of Entity Visualization Design

If the entity_quantity does not exceed ten, we visualize each entity individually. For quantities greater than ten, we represent a single entity accompanied by the quantity number overlaid on it. This approach aligns with common designs in popular educational visual datasets like Twinkl (twinkl, 2025).

C.5 Details of Operation Visualization Designs

Operations define the relationships between different containers. In addition to basic arithmetic operations such as addition, subtraction, multiplication, and division, we incorporate additional operations including surplus, comparison, unit transformation, and multi-step calculations. These operations enable our approach to cover 94.4% of Grade 1-3 MWPs in the ASDiv dataset (Miao et al., 2020).

We visualize these operations using two visual variations: “Formal” and “Intuitive”. In the “Formal” variation, operations are represented using mathematical symbols such as “+”, “-”, “×”, and “÷”, accompanied by text. We show examples in Appendix B.1.

In the “Intuitive” variation, each operation is represented through a specific visual arrangement (see visual examples in Appendix B.2):

Addition: Containers involved in the addition are enclosed within a rectangle. A purple circle with a question mark is placed at the bottom-right corner of the rectangle.

Subtraction: The minuend container is visualized first, with the subtracted items crossed out. A purple circle with a question mark is placed at the bottom-right corner of the rectangle.

Multiplication: The multiplicand container is visualized repeatedly to indicate multiplication. All entities are enclosed within a larger rectangle, with a purple circle and a question mark added at the bottom-right corner, similar to addition. A special type of multiplication involves computing “area”. For such problems, we visualize it as a single item with dimensions corresponding to the width and length described in the MWP.

Division: The division operation is visualized as the post-division state, with multiple container rectangles representing groups enclosed within a larger rectangle. If the MWP asks for the quantity per group (e.g., "10 apples divided into 5 boxes, how many per box?"), a purple question mark circle is placed at the bottom-right of the last container. If it asks for the number of groups (e.g., "10 apples, 2 per box, how many boxes?"), the question mark is placed at the top-right of the larger rectangle.

Surplus: Similar to division, but the surplus container is visualized separately as the remainder. The remainder is placed at last, with a purple circle and a question mark at the bottom-right corner of its rectangle.

Comparison: This operation involves comparing different entities by visualizing them on a balance scale. Each container is placed on one side of the scale.

Unit Transformation: We adopt a purple bubble above each visual item to display its value in the transformed unit.

Finally, for MWPs with multiple operations, we follow these visualization rules for each operation and dynamically combine them to form the overall expression tree (see Figure 21).

D Annotated Dataset Statistics

We present the annotated dataset statistics in Table 8.

E Details of Rendering Programs

We present the algorithm for rendering programs in Algorithm 1. We use rendering programs to map from VL to the desirable visual. The rendering program first converts the VL into a tree structure T, where each operation becomes a parent node and each container becomes a child node. Next, we traverse T in a bottom-up manner. During this traversal, when a container node is encountered, its relative position is computed based on its attributes (e.g. the quantity of entity in this container). Conversely, when an operation node is encountered, the relative positions of its child nodes are updated

Dataset	Visuals	Domain	Use Cases	Grade Level
MATH2VISUAL (ours)	1,903	Primary School Math Word Problems	Supporting primary school students' math understanding; Evaluating and training Text-to-Image models on pedagogical visual generation	Primary school grade 1-3
MATH-Vision	3,040	General mathematics, competition-level problems	Visual math problem-solving; Evaluating multimodal models on math reasoning	Middle school to high school (competition-level difficulty)
MathVista	6,141	Logical, algebraic, and scientific reasoning	Math visual question answering; Puzzle-solving ; logical reasoning; Function analysis ; diagram understanding	Varied (elementary to advanced reasoning)

Table 8: Dataset Statistics

according to the operation type. Note that the positioning of "Formal" and "Intuitive" visuals differs, as detailed in Section 3.4. Once all relative positions are determined, a global layout plan is computed from these values. Finally, we traverse the tree in a top-down order and render each container and operation node according to the global layout plan, using the corresponding elements from the SVG dataset. We retrieve the SVG icon corresponding to the `entity_type`, `container_type` and `attr_type` and map it as the source to the visual. The complete algorithm is presented in Algorithm 1.

Algorithm 1 Rendering Visuals from MWP Visual Language

Require:
VL: A visual language representation of the MWP
SVG: A dataset of SVG elements for rendering

Ensure: Rendered MWP visualization

Step 1: Parse VL
Convert the visual language (VL) into a tree structure T, ignoring result_container when generating "Formal" Visuals.
Step 2: Plan Layout
for each node n in T (traverse in bottom-up order) **do**
    if n represents a container **then**
        Determine the relative position of n based on its attributes (e.g., type, entity_quantity)
    else if n represents an operation **then**
        Update the relative position of n's child node based on the operation type
    end if
end for
Step 3: Compute Global Layout
Integrate the relative positions from all nodes to form a coherent global layout plan
Step 4: Render SVG
for each node n in T (traverse in top-down order) **do**
    Retrieve the final coordinates for n from the global layout plan
    Render n using the corresponding SVG element from the SVG dataset
end for

F Generation Prompts

F.1 Prompt For Visual Language generation

We present the prompt we used for generating Visual Language from MWP as below:

```
You are an expert in converting math word
    problems into a structured 'visual language
    '. Your task is to generate a visual
    language expression based on the given math
    word problem.

**Background Information**
You should use the following fixed format for
    each problem:
<operation>(
   container1[entity_name: <name>, entity_type:
       <type>, entity_quantity: <number>,
       container_name: <container>,
       container_type: <container type>,
       attr_name: <attr>, attr_type: <attr type
       >],
   container2[entity_name: <name>, entity_type:
       <type>, entity_quantity: <number>,
       container_name: <container>,
       container_type: <container type>,
       attr_name: <attr>, attr_type: <attr type
       >],
   result_container[entity_name: <name>,
       entity_type: <type>, entity_quantity: <
       number>, container_name: <container>,
       container_type: <container type>,
```

```
        attr_name: <attr>, attr_type: <attr type>]
)

operation can be "addition", "subtraction", "multiplication", "division", "surplus", "area", "comparison", or "unittrans".

Each container has the attributes: entity_name, entity_type, entity_quantity, container_name, container_type, attr_name, attr_type.
For example, a girl named Lucy may be represented as:
entity_name: Lucy, entity_type: girl.

The optional attributes container_name, container_type, attr_name, and attr_type allow extended descriptions.
In the MWP description "Jake picked up three apples in the morning...", the container1 could be:
entity_name: apple, entity_type: apple, entity_quantity: 3, container_name: Jake, container_type: boy, attr_name: morning, attr_type: morning.
These additional attributes are not fixed and may vary according to different interpretations.

Example of Visual Languages: ...

Once you are ready to perform the task, you may write down your thought process, but please ensure that you provide the final visual language expression in the following format at the end:

visual_language: <the visual language result>
Question:
Formula:
```

F.2 Prompt for Visual Generation

F.2.1 Prompt for Formal Visual Generation

```
Please Create an educational visual for this math word problem: ...
Suppose this problem has formula: ...

The visual consists of:
1. Container: We use rectangular sections to represent different containers or group of entities. Inside each rectangle, display the entities of this container (e.g., apples, balls, etc.).
2. Container Name: Above each rectangle, place a container icon (e.g., an orange basket, jar, or other container type) and label it with the container's name (e.g., 'basket,' 'jar,' etc).
3. Operation Symbol: Between each two rectangles, include an operation symbol that varies depending on the problem
4. Outcome Section: To the right, place an '=' symbol followed by a '?' to symbolize the unknown solution.

Example:
For problem: Lucy has five oranges and Jake has two oranges. How many oranges do they have together?
formula: 5+2=7
The visual consists of two containers, "Lucy" and "Jake," as rectangulars labeled with their names and icons (boy icon for Jake and girl for Lucy) on the top of each rectangle. Each rectangle contains oranges corresponding to their quantities (Lucy: 5, Jake: 2). A "+" symbol between the rectangles indicates the addition operation, and an "=" followed by a question mark represents the unknown solution.

Special cases:
1. For comparison problem, please use a balance scale to weigh different entities. For problem 'Lucy has 4 strawberries. Jake gave her 5 more. She needs 10 strawberries to make a cake. Does she have enough to make a cake?' We draw a balance scale. On the left side of the scale, two rectangular sections represent 'Lucy' and 'Jake,' each labeled with their names and icons. Lucy's section contains 4 strawberries, and Jake's section contains 5 strawberries. A "+" symbol indicates the addition of their strawberries. To the right of this, an "=" symbol and a question mark. On the right side of the scale, another rectangular section labeled "cake" contains 10 strawberries, representing the required amount. An "=" symbol and a question mark follow it.
2. For unit transformation problem, please use a purple buble with the converted value in it on the top of each item to represent the unit value of the current item. For example, a problem like 'Charles found 6 pennies on his way to school. He also had 3 nickels already at home. How much money does he now have in all?' can be represented as a visual: on the left side, a rectangular section labeled "on his way" contains 6 pennies, each with a purple bubble above it displaying its converted value of 0.01 (representing dollars). On the right side, another rectangular section labeled "home" contains 3 nickels, each with a purple bubble above it displaying its converted value of 0.05. A "+" symbol is placed between the two sections to indicate the addition of their values. To the right of the sections, an "=" symbol is followed by a question mark.
3. For surplus problem, please use text remainder with a new question mark after previous question mark.
4. If any container have item quantity higher than 10, please visualize only one item inside this container rectangle to be bigger and put the quantity number to cover the item. For example, if the problem is 'Lucy has 15 apples and Jake has 3 apples. How many apples do they have together?', the visual should show 15 apples for Lucy and 3 apples for Jake. Lucy's apples should be represented by a single apple that is larger than Jake's apples, and the number 15 should be placed on top of it to indicate the quantity. Jake's apples should be represented by three smaller apples. The "+"
```

```
symbol between the two entities indicates
the addition operation, and an "=" symbol
followed by a question mark.
```

F.2.2 Prompt for Intuitive Visual Generation

```
Please Create an educational visual for this
    math word problem: ...
Suppose this problem has formula: ...

The visual consists of:
1. Container: We use rectangular sections to
   represent different containers or group of
   items. Inside each rectangle, display the
   items of this container (e.g., apples, balls
   , etc.).
2. Container Name: Above each rectangle, place a
    container icon (e.g., an orange basket, jar
   , or other container type) and label it with
    the container's name (e.g., 'basket,' 'jar,
    etc).

Handle different operations:
1. For addition, use a big rectangle to cover
   all container rectangles need to be added
   together. And place a purple circle with
   question mark inside at the right bottom
   side of the big rectangle.
2. For subtraction, first visualize minuend
   container then cross out item that been
   subtracted. Place a purple circle with
   question mark inside at the right bottom
   side of the minuend container rectangle.
3. For multiplication, repeatedly visualize the
   multiplicand container. Use a big rectangle
   to cover all container. Place purple circle
   with question mark similar as addition.
4. For division, visualize it as the state after
    division, with many container rectangles
   represent different groups. If asking about
   quantity in single container, place purple
   circle at the right bottom of the last
   container rectangle. If asking about number
   of container, place purple circle at the
   right top of the big rectangle.
5. For surplus, similar as division, only
   difference is you should visualize the
   surplus container at the last and place the
   purple circle at the right bottom side of
   surplus container rectangle.
6. For comparison, use a balance scale to weigh
   different containers. Visualize entities on
   the left and right side of the scale
   separately.
7. For unit transformation, use a purple buble
   with the converted value in it on the top of
    each item to represent the unit value of
   the current item.
8. For problem involving multiple addition and
   subtraction, use the same visualization rule
    and combine dynamically.

Example:
For problem: Lucy has five oranges and Jake has
   two oranges. How many oranges do they have
   together?
formula: 5+2=7
The visual consists of two containers, "Lucy"
   and "Jake," as rectangulars labeled with
   their names and icons (boy icon for Jake and
    girl for Lucy) on the top of each rectangle
   . Each rectangle contains oranges
   corresponding to their quantities (Lucy: 5,
   Jake: 2). A bigger rectangle encompasses the
    two containers to indicate addition.
```

G Fine-tuning Details

G.1 Fine-tuning LLMs

For fine-tuning LLMs, we create a training set containing 1,011 VL instances using stratified sampling based on "Grade" and "Question Type," which occupies 80% of the entire dataset. We fine-tuned four variations of LLMs in total: Llama-3.1-8B with formula, Llama-3.1-8B without formula, Mistral-7B-v0.3 with formula, and Mistral-7B-v0.3 without formula.

In line with the methodology described in (Wang et al., 2024), all models are fine-tuned for 10 epochs using a per-device batch size of 2 with gradient checkpointing enabled. We set an initial learning rate of 2.5e-5 and employ a linear learning rate decay scheduler with a warmup phase comprising 3% of the total training steps. Model optimization is performed using the paged_adamw_8bit optimizer (Loshchilov and Hutter, 2019) from the transformers library (Wolf et al., 2020) to minimize the negative log-likelihood of the ground-truth responses. Additionally, LoRA adapters (Hu et al., 2022) are incorporated to efficiently fine-tune the models. We use one RTX 4090 GPU to fine-tune each model; the Llama-3.1-8B with formula and Llama-3.1-8B without formula models have 8 Billion parameters and take 12 hours to train; the Mistral-7B-v0.3 with formula and Mistral-7B-v0.3 without formula models have 7 Billion parameters and take 11 hours to train. We report the human evaluation of a single inference result for each model.

G.2 Fine-tuning Text-to-Image Models

For fine-tuning TTI models, we create a Formal visual training set containing 1,011 visuals corresponding to the training set used by the LLM, and an Intuitive visual training set containing 502 visuals. Both training sets occupy 80% of their corresponding ground truth datasets. We fine-tuned four variations of TTI models in total: Flux.1-dev with formula, Flux.1-dev without formula, Stable Diffusion-3.5-large with formula, and Stable Diffusion-3.5-large without formula.

All TTI models are fine-tuned for 10 epochs, with a training batch size of 5 and gradient check-

Method	Accuracy		Completeness		Clarity		Cog Load Opt	
	Formal	Intuitive	Formal	Intuitive	Formal	Intuitive	Formal	Intuitive
ft_ mistral-7B-v0.3(F)	2.83	2.71	3.00	2.67	**3.00**	2.71	**2.96**	2.71
ft_ mistral-7B-v0.3(NF)	2.54	2.08	2.67	2.04	2.67	2.08	2.63	2.08
zs_ mistral-7B-v0.3(F)	1.33	1.00	1.38	1.00	1.46	1.00	1.46	1.00
zs_ mistral-7B-v0.3(NF)	1.25	1.00	1.21	1.00	1.29	1.00	1.33	1.00
ft_stable-diffusion-3.5-large(F)	**2.96**	**2.88**	**3.12**	**3.08**	2.92	**3.75**	2.83	**3.58**
ft_stable-diffusion-3.5-large(NF)	**2.96**	2.75	3.08	**3.08**	2.79	3.58	2.83	3.54
zs_stable-diffusion-3.5-large(F)	2.71	2.67	2.96	2.92	2.83	3.08	2.83	2.96
zs_stable-diffusion-3.5-large(NF)	2.71	2.67	2.96	2.71	2.83	3.08	2.83	2.96

Table 9: Other evaluation results for different visual generation methods. For each method, 48 visuals (24 Formal and 24 Intuitive) were evaluated, with each score representing the average rating from two researchers on a 1–5 scale (higher is better). (F) indicates the method used the solution formula as input, while (NF) indicates it did not. "ft" means this model is fine-tuned on our annotated dataset, while "zs" means zero-shot.

pointing enabled; we employ the AdamW_BF16 optimizer (Loshchilov and Hutter, 2019) with an initial learning rate of 1e-5 and a polynomial learning rate scheduler (with zero warmup steps). We integrate LoRA adapters via the Lycoris approach within a diffusers attention framework (Yeh et al., 2023). Additionally, images are generated at a resolution of 1024. We use one RTX 4090 GPU to fine-tune each model; the Flux.1-dev with formula and Flux.1-dev without formula models have 12 billion parameters and take 48 hours to train, while the Stable Diffusion-3.5-large with formula and Stable Diffusion-3.5-large without formula models have 8.1 billion parameters and take 15 hours to train. We report the human evaluation of a single inference result for each model.

G.3 Other Fine-tuning Results

We present other fine-tuning experiment results in Table 9.

H Details of Qualitative Analysis

H.1 Procedure

The thematic analysis was performed on a sample of 120 visuals generated by the fine-tuned Flux model with formula, zero-shot Flux model with formula and Recraft-v3 with formula, and a total of eight error types were identified. However, three of these error types occurred fewer than eight times. After discussing the findings, we consolidated the labels and focus on five major types of error in close coding.

In the systematic evaluation phase, two researchers manually analyzed 576 visuals generated by the fine-tuned Flux model with formula, the zero-shot Flux model with formula, and Recraft-v3 with formula.

H.2 Statistical Results

We present the statistical results for supporting qualitative analysis in Table 3.

Criterion	Edit Dist↓	LM Ratio↑
ft_ mistral-7B-v0.3(F)	**2.98**	**39.30**
ft_ mistral-7B-v0.3(NF)	3.14	19.07
zs_ mistral-7B-v0.3(F)	7.05	0
zs_ mistral-7B-v0.3(NF)	6.86	0

Table 10: Other results of Visual Language generation. F denotes generation with the solution formula as input, while NF denotes generation without the formula.

I Ethical Consideration and Applications

I.1 Potential Risks

One potential risk is that the generated visuals might be misinterpreted if they do not accurately capture the intended mathematical relationships, potentially leading to confusion among students and educators. To minimize this risk, we collaborated closely with primary school math teachers to develop the structured design space that aligns with pedagogical standards. We further annotate the generated dataset and ensure clarity and accuracy in the visuals.

I.2 Terms of Use

This section outlines the terms and conditions for the use of MATH2VISUAL. 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 MATH2VISUAL project.

Updates and Changes The authors reserve the right to make changes to the terms of this license or the MATH2VISUAL itself at any time.

I.3 Compliance with Artifact Usage and Intended Use Specifications

I.3.1 Compliance with Existing Artifact Usage

In our study, we utilized a range of existing artifacts, such as open-source SVG datasets from various sources (svgrepoRepoFree, 2025; iconfont, 2025; svgen, 2025; Condino, 2022; YILDIRIM, 2023; pexels, 2025) and ASDiv dataset (Miao et al., 2020), to develop our visual datasets. We rigorously ensured that our usage of these materials was in strict accordance with their intended purposes, aligning with each dataset's vision of freely accessible content. Additionally, we employed various computational tools within their prescribed licensing terms, thus adhering to ethical and legal standards.

I.3.2 Specification of Intended Use for Created Artifacts

Our research led to the development of two significant artifacts:

Framework for Generating Pedagogically Meaningful Visuals

Intended Use: This framework is designed for academic research and educational technology development. It facilitates the generation of pedagogically meaningful visuals, aiming to enhance AI-driven educational tools.

Restrictions: The framework should be used within the bounds of educational and research settings. Any commercial or high-stakes educational application is advised against without further validation and ethical review.

Ethical Considerations: We emphasize the responsible use of this framework, particularly in maintaining the integrity and context of the source textbooks.

Dataset of Generated Visuals

Intended Use: The dataset is primarily intended for research in educational technologies. It offers a resource for developing and testing Text-to-Image models in educational contexts.

Restrictions: This dataset is not recommended for direct application in live educational settings without substantial vetting, as it may contain synthetic inaccuracies.

Data Ethics: As the dataset is derived from open-source SVG datasets, it respects the principles of open access. We encourage users to keep the dataset within academic and research domains, in line with the ethos of the source material.

I.4 Data Collection and Anonymization Procedures

In our research, rigorous steps were taken to ensure that the data collected and used did not contain any personally identifiable information or offensive content. The data, primarily sourced from open-access MWP datasets and SVG datasets, inherently lacked individual personal data. For the components involving human interaction, such as feedback or evaluation, all identifying information was carefully removed to maintain anonymity. Additionally, we implemented a thorough review process to screen for and exclude any potentially offensive or sensitive material from our dataset. These measures were taken to uphold the highest standards of privacy, ethical data usage, and respect for individual confidentiality.

I.5 Artifact Documentation

I.5.1 Visual Generation Framework

Domain Coverage The framework is designed to generate pedagogically meaningful visuals from MWP for teaching MWP.

Operation Coverage It covers seven operations including: addition, subtraction, multiplication, division, surplus, comparison and unit transformation.

I.5.2 Dataset of Generated Visuals

Visual and Style The visuals are primarily generated from English MWPs. The style is educational and academic, suited for educational purposes.

Content Diversity The dataset spans multiple academic disciplines, offering a rich variety of topics and themes.

Demographic Representation While the dataset itself does not directly represent demographic groups (as it is synthesized from MWP dataset), the diversity in the source material reflects a broad spectrum of cultural and societal contexts.

I.6 Use of AI Assistants in Research

In our study, AI assistants were used sparingly and in accordance with ACL's ethical guidelines. 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.

I.7 Instructions Given To Participants

I.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.

I.7.2 Instructions for Experiments

Thank you for participating in our study. This research has received ethical approval, and your consent has been obtained. The entire study will take approximately 1.5 to 2 hours and consists of four sessions. Please read the instructions below carefully:

Session One – Visual Approach Preference: You will be shown two visual approaches for representing math word problems (MWPs):

1. Multiple Visuals: Each visual represents one sentence of the MWP.

2. Single Visual: One visual represents the entire MWP. Please indicate your preference between these two approaches.

Session Two – Design Variation Evaluation: You will review six design variations for visualizing MWPs. These variations differ based on:

How Quantities Are Visualized:

1. How Quantities Are Visualized:
   - **Abstract**: Quantities are represented as text from the MWP.
   - **Hybrid**: A single item is visualized with a label at the bottom-right corner indicating its quantity.
   - **Visual**: Items are directly drawn in quantities matching their number.
2. How Operations Are Visualized:
   - **Formal**: Mathematical operations are represented using standard symbols (e.g., +, -, ×, ÷).
   - **Intuitive**: Operations are visualized using specific arrangements for each operation.

For each design variation, please complete a questionnaire rating:

- Clarity: How clearly the visual design represents the MWP.
- Engagement: Whether the design appears to improve student engagement.
- Cognitive Load: Whether the design avoids introducing unnecessary cognitive load.

The order of presentation will be randomized to minimize order effects.

Session Three – Operation Design Feedback: In this session, you will review our "Intuitive" design for visualizing mathematical operations. Using the same criteria (Clarity, Engagement, Cognitive Load), please provide your feedback via a questionnaire. The presentation order will also be randomized.

Session Four – Evaluation Criteria Discussion: We will discuss with you the criteria that will be used to analyze the visuals generated by our system. This discussion focuses on how effectively our automated generation approach reproduces the intended design. After this discussion, you will complete a post-task questionnaire assessing the pedagogical value of the visual design.

Please answer all questions honestly and provide any suggestions for improvement. Your feedback is crucial for enhancing our framework. If you have any questions during the study, feel free to ask the researcher.

Thank you for your time and valuable input!

I.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.