TheoremExplainAgent: Towards Multimodal Explanations for LLM Theorem Understanding

Anonymous ACL submission

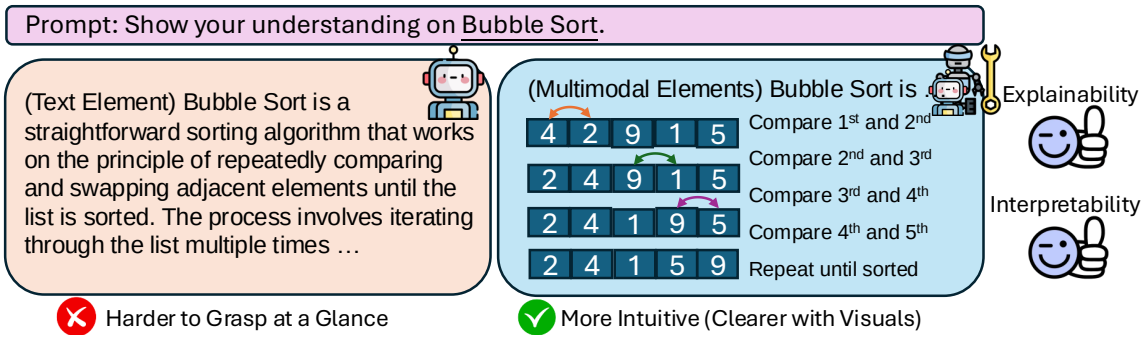

Figure 1: We do not have knowledge of a thing until we have grasped its cause (Aristotle, 1901). A strong reasoning model should not only generate correct conclusions but also communicate them effectively. Visualization enhances human intuition by making abstract concepts more concrete and revealing hidden relationships. Moreover, visual explanations expose reasoning errors more clearly than text, making it easier to diagnose model mistakes.

Abstract

Understanding domain-specific theorems often requires more than just text-based reasoning; effective communication through structured visual explanations is crucial for deeper comprehension. While large language models (LLMs) demonstrate strong performance in text-based theorem reasoning, their ability to generate coherent and pedagogically meaningful visual explanations remains an open challenge. In this work, we introduce TheoremExplainAgent, an agentic approach for generating long-form theorem explanation videos (over 5 minutes) using Manim animations. To systematically evaluate multimodal theorem explanations, we propose TheoremExplainBench, a benchmark covering 240 theorems across multiple STEM disciplines, along with 5 automated evaluation metrics. Our results reveal that agentic planning is essential for generating detailed long-form videos, and the o3-mini agent achieves a success rate of 93.8% and an overall score of 0.77. However, our quantitative and qualitative studies show that most of the videos produced exhibit minor issues with visual element layout. Furthermore, multimodal explanations expose deeper reasoning flaws that text-based explanations fail to reveal, highlighting the importance of multimodal explanations.

1 Introduction

A key objective of AI systems is to assist humans in solving complex problems, particularly in domain-specific challenges. To achieve this, AI must go beyond surface-level pattern matching to achieve deeper conceptual understanding to effectively address these problems. Recent research has proposed evaluating AI performance on theorem-driven datasets through multiple-choice question answering (Zhang et al., 2024) and open-ended short question answering (Chen et al., 2023b). However, these approaches primarily assess textual reasoning and may not fully capture an AI system's ability to grasp theorem concepts at a deeper level. Studies have shown that AI models can be sensitive to superficial cues, such as the order of answer choices in multiple-choice questions (Pezeshkpour and Hruschka, 2023; Keluskar et al., 2024). This raises concerns about the robustness of such evaluations in truly measuring comprehension. Moreover, current theorem-focused datasets are predominantly text-based, overlooking how complex concepts are often best understood through structured visualizations.

Theorem reasoning is inherently multimodal, particularly in areas such as geometry, topology,

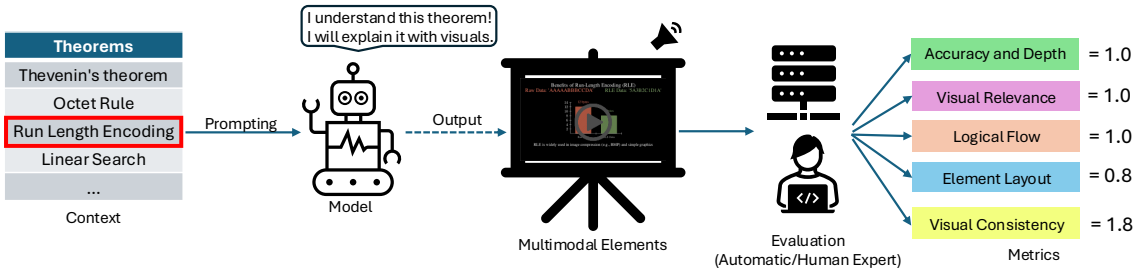

Figure 2: An overview of the multimodal theorem explanation framework.

and certain aspects of algebra, where visual representations and spatial reasoning play a crucial role in understanding structures and proving properties. Cognitive science research suggests that multimodal elements improve conceptual understanding, aiding in the comprehension of abstract ideas. Although some studies leverage multimodal input to improve AI reasoning (Zhang et al., 2023b), currently there is no standardized evaluation framework to evaluate AI's ability to generate multimodal explanations for complex concepts, which would require models to express knowledge in an interpretable manner. This raises the question: **Can AI systems effectively generate multimodal theorem explanations?**

As video is a classic example of multimodal data, we explore the question by introducing TheoremExplainAgent, an agentic AI system designed to generate theorem explanations in the form of explanatory videos. TheoremExplainAgent demonstrates the capability to plan and generate long, coherent videos by mimicking human video production processes. In this system, a planner agent generates story plans and narrations, and a coding agent generates Python animation scripts using Manim (The Manim Community Developers, 2024) to create long and meaningful videos. Additionally, to systematically evaluate AI-generated explanations, we develop TheoremExplainBench, a benchmark suite comprising 240 theorems spanning four STEM disciplines. We assess AI-generated explanations based on 5 dimensions related to factual correctness and perceptual quality, using automatic or human-evaluation metrics. An overview of the framework is illustrated in Figure 2.

Our experiments with TheoremExplainAgent yielded both promising results and clear areas for improvement in AI-generated multimodal theorem explanations. On the positive side, a key achievement was the system's ability to generate extended video explanations, reaching durations of up to 10 minutes. This represents a significant advancement over agentless approaches, which we found to be limited to approximately 20-second videos. Furthermore, TheoremExplainAgent demonstrated versatility across different STEM disciplines, successfully creating videos for Mathematics, Physics, Chemistry, and Computer Science. Importantly, we observed that video-based theorem explanations inherently expose deeper reasoning flaws in AI systems that text-based evaluations often miss. Unlike text-based multiple-choice questions, where models can exploit superficial cues, generating visual-theorem explanations necessitates that the AI explicitly encodes structural and procedural knowledge, thus making underlying errors more apparent. In particular, the o3-mini model exhibited robust performance at varying levels of theorem difficulty, indicating a capacity to handle both fundamental and complex concepts. However, despite these successes, limitations persist. While the system could generate textually accurate explanations, the visual quality and pedagogical structure of the videos require further refinement. Generated animations frequently exhibited minor visual layout inaccuracies, such as misaligned text elements, overlapping shapes, and inconsistent object sizes. These visual errors, though often subtle, became more pronounced and potentially distracting, particularly in the medium and hard difficulty levels of our TheoremExplainBench.

Therefore, the major contributions of this work:

(1) Task Definition. We introduce the novel problem of AI-generated multimodal theorem explanations and identify the key challenges associated.

(2) TheoremExplainAgent. We develop an agentic approach to generating explanatory videos, as a baseline to assess current AI capabilities.

(3) TheoremExplainBench. We curate a diverse benchmark dataset spanning 4 STEM disciplines and propose 5 automatic evaluation metrics, measuring progress toward solving this problem.

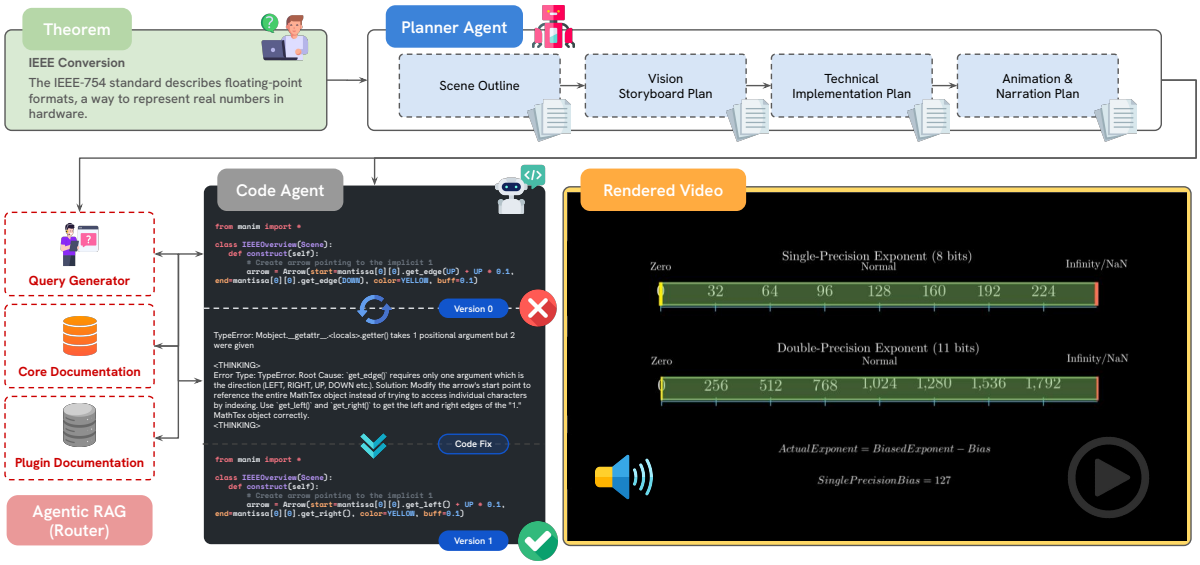

Figure 3: TheoremExplainAgent consists of two LLM agents. Taking a theorem as input, the planner agent create plans for execution. The coding agent then generates Python scripts to produce visuals and audio.

2 Related Works

2.1 LLM and Agents

The rapid advancements in large language models (LLMs) and large vision-language models (VLMs) have unlocked unprecedented capabilities in understanding multimodal content. Models such as GPT-4 (OpenAI, 2023), Gemini (Gemini-Team et al., 2024), Claude-3.5 Sonnet v1 (Anthropic, 2024), and DeepSeek (DeepSeek-AI et al., 2024) have demonstrated strong abilities in processing complex textual information and analyzing visual inputs within a unified framework (Zhang et al., 2023b). These breakthroughs have enabled transformative applications across various domains, including visual content understanding (Hu et al., 2023; Ku et al., 2023), code generation (Nijkamp et al., 2023; Jimenez et al., 2024; Yang et al., 2024a), and reasoning over structured data. To tackle complex tasks, researchers have explored LLM agents: AI systems that leverage LLMs to autonomously reason, plan, and execute tasks by interacting with structured environments or external tools. These agents have been deployed in various goal-oriented applications, such as scientific discovery (Lu et al., 2024; Si et al., 2024; Schmidgall et al., 2025), coding solutions (Abramovich et al., 2024), multimodal visual generation (He et al., 2024), and computer environment interaction (Xie et al., 2024). In this work, we extend the use of LLM agents into the domain of theorem explanation and visualization.

2.2 LLM in Theorems Understanding

LLMs have demonstrated remarkable capabilities in solving complex mathematical problems, including formal theorem proving and symbolic reasoning. To evaluate these abilities, researchers have introduced multiple benchmark datasets, primarily consisting of multiple-choice and short-answer question answering (QA) tasks (Zhang et al., 2024; Amini et al., 2019; Hendrycks et al., 2021). Early studies centered on elementary to high school-level mathematics, leading to datasets such as Math23K(Zhou et al., 2023), GSM8K(Cobbe et al., 2021), and GeoQA(Chen et al., 2022a). As LLM capabilities advanced, more domain-specific benchmarks emerged, extending evaluation to fields like science reasoning (ScienceQA) (Lu et al., 2022), financial reasoning (FinQA) (Chen et al., 2022b), and theorem comprehension (TheoremQA) (Chen et al., 2023b). These datasets collectively assess LLMs' ability to solve mathematical and scientific problems up to the university level. However, existing benchmarks remain predominantly text-based, overlooking the role of visual intuition in mathematical reasoning. Many mathematical concepts are best understood through structured diagrams and dynamic representations, which current LLM evaluations fail to capture. To address this gap, we introduce an AI framework to generate theorem explanations in long-form videos, integrating symbolic derivations with structured visualizations to enhance comprehension.

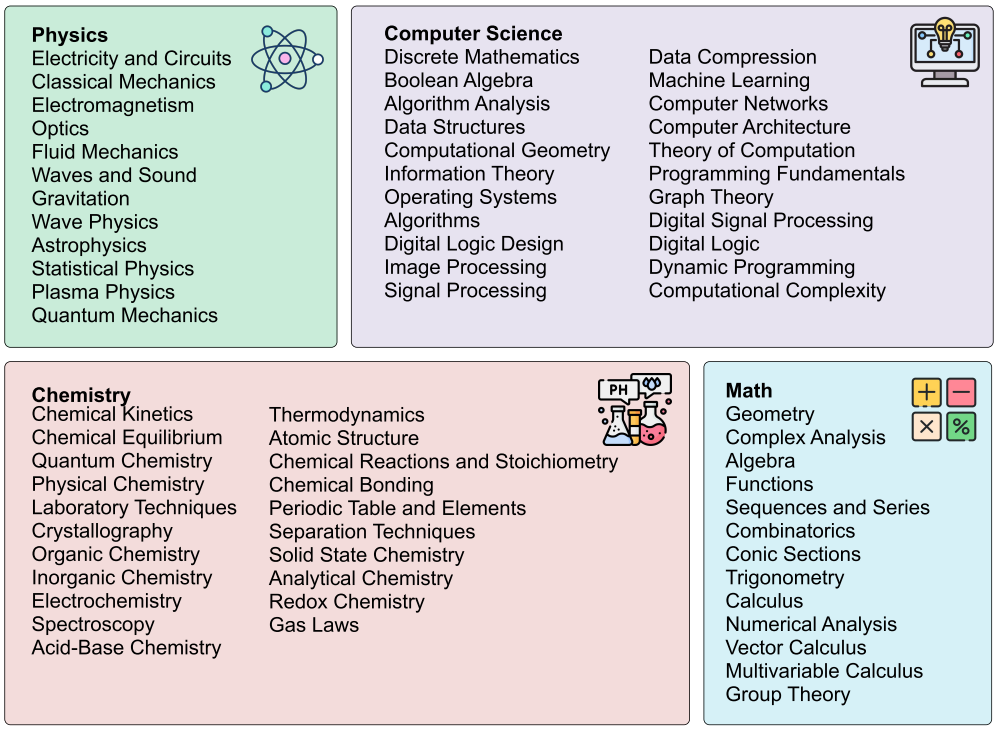

Figure 4: Subfields of TheoremExplainBench under Computer Science, Chemistry, Mathematics, and Physics.

2.3 LLM in Visualizations

Recent advancements in AI-driven visualization have enabled AI systems to generate structured visual content from textual descriptions (Li et al., 2024). These models typically process text-based inputs and produce programmatic representations, which are then converted into visual outputs (Ritchie et al., 2023; Goswami et al., 2025). This approach has been applied across various domains, including scientific visualization (Yang et al., 2024b), data representation (Galimzyanov et al., 2024), and motion graphics (Zhang et al., 2023a). Efforts such as Drawing-Pandas (Galimzyanov et al., 2024) have introduced benchmarks for evaluating code-based plotting in Matplotlib and Seaborn. Follow-up works like MatPlotAgent(Yang et al., 2024b) demonstrated that agentic approaches outperform agentless methods in visualization generation, while PlotGen (Goswami et al., 2025) incorporated multimodal feedback for iterative refinement, further improving visualization quality. Our work is the first to explore AI-driven visualization for generating animated theorem explanations, seamlessly integrating step-by-step symbolic derivations with structured motion graphics, bridging the gap between mathematical reasoning and visual comprehension.

3 Method

3.1 Task Definition

Model Input. The model receives a theorem along with a short description that provides context, which helps the model identify the theorem.

Model Output. The model is to output a video that combines animations, structured derivations, and voiceover narration to provide a multimodal and comprehensive explanation of the theorem. The video is expected to be longer than a minute, featuring long animations across different scenes, with narration guiding the viewer through step-by-step proofs and real-world applications.

3.2 TheoremExplainAgent (TEA)

We develop TheoremExplainAgent (TEA), an agentic pipeline designed to automate the generation of videos using multiple specialized agents as shown in Figure 3. The process begins with the planner agent, which creates a high-level video plan according to the specified theorem. This plan consists of multiple scenes, each corresponding to a key segment of the resulting video. Once the initial plan is created, the planner agent refines the details of each scene, breaking them down into smaller components that define the specific visual elements, animations, and transitions needed. These detailed

Agent	Easy	Medium	Hard	Math	Phys	CS	Chem	Overall
GPT-4o	61.3%	57.5%	46.2%	61.7%	55.0%	58.3%	45.0%	55.0%
GPT-4o + RAG	42.5%	57.5%	37.5%	70.0%	40.0%	41.7%	31.7%	45.8%
Claude 3.5-Sonnet v1	2.5%	1.2%	2.5%	1.7%	1.7%	1.7%	3.3%	2.1%
Claude 3.5-Sonnet v1 + RAG	18.8%	13.8%	11.2%	23.3%	10.0%	20.0%	5.0%	14.6%
Gemini 2.0-Flash	20.0%	11.2%	12.5%	16.7%	8.3%	21.7%	11.7%	14.6%
Gemini 2.0-Flash + RAG	23.8%	21.2%	16.2%	26.7%	15.0%	20.0%	20.0%	20.4%
o3-mini (medium)	93.8%	91.2%	96.2%	95.0%	93.3%	93.3%	93.3%	93.8%
o3-mini (medium) + RAG	83.8%	82.5%	80.0%	81.7%	90.0%	88.3%	68.3%	82.1%

Table 1: Agent success rate in generating complete videos across different difficulty levels and subjects.

Agent	Accuracy and Depth	Visual Relevance	Logical Flow	Element Layout	Visual Consistency	Overall Score
GPT-4o	0.79	0.79	0.89	0.59	0.87	0.78
GPT-4o + RAG	0.75	0.77	0.88	0.57	0.86	0.76
Claude 3.5-Sonnet v1	0.75	0.87	0.88	0.57	0.92	0.79
Claude 3.5-Sonnet v1 + RAG	0.67	0.79	0.69	0.65	0.87	0.71
Gemini 2.0 Flash	0.82	0.77	0.80	0.57	0.88	0.76
Gemini 2.0 Flash + RAG	0.79	0.75	0.84	0.58	0.87	0.76
o3-mini (medium)	0.76	0.76	0.89	0.61	0.88	0.77
o3-mini (medium) + RAG	0.75	0.75	0.88	0.61	0.88	0.76
Human-made Manim Videos	0.80	0.81	0.70	0.73	0.87	0.77

Table 2: Performance of our proposed metrics on successfully generated long-form videos by the agents.

scene descriptions are then passed to the coding agent, which generates the corresponding Python code. The voiceover is also generated through a text-to-speech service. Finally, the Python scripts are executed to produce the final video, which reflects the narrative or instructional goals outlined in the video plan. If the generated Python code encounters an error, the coding agent will review the error and generate a revised version of the code. We set a maximum of N attempts where $N = 5$. If this limit is exceeded, we mark the generation as unsuccessful.

Coding Toolkit. We choose Manim (The Manim Community Developers, 2024) as the coding toolkit because it is a popular open-source Python library designed for creating mathematical animations and educational videos through code-driven visualizations. YouTube channels such as 3Blue1Brown (Sanderson, 2020) have demonstrated how Manim-made videos can convey complex mathematical concepts in an intuitive way. In our context, the coding agent translates each scene's specifications into executable Manim scripts, which define objects such as text, shapes, graphs, or equations, along with their corresponding animations, timings, and transitions.

Agentic Retrieval-Augmented Generation. To enhance code generation ability, we implemented a multifaceted retrieval-augmented generation (RAG) approach, leveraging the Manim documentation as the primary knowledge base. Unlike a single monolithic retrieval step, our agentic approach first classifies whether the theorems are suitable for using specific Manim plugins. Then it generates relevant queries at different stages of the video creation process: (1) during storyboard generation, to retrieve visual examples and related concepts; (2) during technical implementation, to fetch specific code snippets and usage patterns; and (3) during error correction, to diagnose issues and suggest solutions. These queries are cached to prevent redundant computations, and the agent dynamically selects the most relevant documents based on a relevance scoring threshold, ensuring efficient and precise retrieval.

3.3 TheoremExplainBench (TEB)

We curate an evaluation dataset comprising 240 theorems from various disciplines, including Computer Science, Chemistry, Mathematics, and Physics. Each entry includes the theorem name and a contextual description, sourced from OpenStax (Baraniuk, 2025) and LibreTexts (Larsen, 2025). To facilitate structured assessment, the the-

orems are categorized into three difficulty levels: Easy (high school level), Medium (undergraduate level), and Hard (graduate level), with 80 entries in each category. TheoremExplainBench (TEB) features 68 sub-fields that cover a wide range of domains as shown in Figure 4.

To fully define this novel problem, we propose a comprehensive evaluation metric applicable to both human-created and AI-generated explanatory videos, ensuring a standardized assessment across different content sources. Our metric evaluates videos across five key dimensions. The first three dimensions assess the factual correctness of explanations, while the last two dimensions evaluate the perceptual quality of the videos.

Accuracy and Depth. Evaluates whether the narration provides a precise and well-structured explanation of the theorem, offering both intuitive insights and rigorous justifications for why it holds.

Visual Relevance. Assesses whether the video frames effectively align with the theorem's concepts and derivations, reinforcing the explanation through appropriate visual representations.

Logical Flow. Examines whether the video follows a clear and coherent structure, ensuring a logical progression that builds upon ideas effectively.

Element Layout. Evaluates whether visual elements are well-positioned and appropriately sized within the frame, avoiding unintended overlap and ensuring clarity in presentation.

Visual Consistency. Assesses whether the motions are smooth, and whether the visual style remains uniform across frames.

In our metric implementation, Accuracy & Depth and Logical Flow are assessed using text-based evaluation with GPT-4o (OpenAI, 2023). The text elements are extracted from video transcripts in SubRip (SRT) format. For Visual Relevance and Element Layout, we apply image processing techniques to identify key frames and use GPT-4o to assign scores for each dimension. To evaluate motions in Visual Consistency, we utilize Gemini 2.0-Flash (DeepMind, 2025) to analyze chunked video segments. The overall score (ranging from 0 to 1) is then computed as the geometric mean of all dimensions. To ensure output stability, we employ greedy decoding (i.e., temperature $= 0$) in the LLM evaluations.

To validate the effectiveness of our evaluation metrics, we conducted a small-scale human study. We sampled 40 videos from our results, selecting 10 from each discipline in TheoremExplainBench. We then recruited 12 experienced STEM student annotators to participate in the study. The rating process followed the same five evaluation dimensions as our proposed metrics, with human raters selecting scores from [0, 0.5, 1]. To assess alignment between our metrics and human evaluations, we computed the Spearman correlation on the sampled subset. To ensure result reliability, we measured inter-rater agreement using Krippendorff's alpha(Krippendorff, 2011), which is more suitable than Fleiss' Kappa(Fleiss and Cohen, 1973) due to the ordinal nature of the ratings. Additionally, to contextualize human performance, we sourced 10 human-made theorem explanation videos from YouTube for comparison.

4 Experimental Results

For the agent candidates in TheoremExplainAgent, we experimented with GPT-4o(OpenAI, 2023), Gemini 2.0 Flash(DeepMind, 2025), Claude 3.5 v1(Anthropic, 2024), and o3-mini(OpenAI, 2025). Each candidate was used for both the planner agent and coding agent, ensuring consistency across configurations. We evaluated all agents across 240 theorems from TheoremExplainBench, comparing their performance under different setups. Our findings indicate that an agentless approach fails to generate videos longer than 20 seconds, whereas TheoremExplainAgent successfully produces videos of up to 10 minutes. Consequently, all experimental results presented below are based on the agentic approach.

Table 1 reveals that the success rate in generating long-form theorem explanation videos varies significantly across difficulty levels and subjects. Overall, o3-mini consistently outperforms other models, maintaining high success rates across both easy and hard tasks, as well as across different STEM domains. In contrast, GPT-4o performs moderately well but show a declining success rate as complexity increases, suggesting difficulties in handling longer and more structured explanations. Gemini 2.0-Flash struggles the most, with notably lower success rates across all conditions. Across subjects, Mathematics tends to have the highest success rates, whereas Chemistry appear to be the most challenging domain. This observation may be attributed to the fact that complex objects in Chemistry, such as flask shapes and atoms, are more challenging to illustrate than simpler primitives in Mathematics, like triangles.

	Spearman	Krippendorff's α
Accuracy and Depth	0.14	0.45
Visual Relevance	0.72	0.36
Logical Flow	0.16	0.56
Element Layout	0.42	0.31
Visual Consistency	0.17	0.36

Table 3: Correlation on Metric-Human correlation (Spearman) and Inter-rate Agreement (Krippendorff's alpha) for the five evaluation dimensions.

Given the successfully generated videos, we compiled Table 2 to present the metric results. Among the evaluated models, GPT-4o and o3-mini performed the best overall, both achieving strong scores across multiple dimensions. GPT-4o excelled in accuracy and depth, as well as logical flow, while o3-mini demonstrated the strongest performance in logical flow and a solid element layout. On the other hand, Gemini 2.0 Flash with RAG performed the weakest overall, struggling particularly with element layout and logical flow, indicating challenges in maintaining structured and visually coherent outputs. Human-made Manim videos, while scoring the similar overall among AI-generated results, achieved the highest visual relevance and element layout. This may be because AI-generated videos tend to exhibit minor issues like overlapping elements and misalignment, which can affect clarity and structure. Interestingly, human-made videos scored lower in logical flow. This may be due to the more natural and less structured narration in human explanations, which often prioritize engagement over strict logical progression. In contrast, AI-generated videos tend to maintain a consistent logical structure, adhering closely to predefined formats. However, this rigidity may sometimes come at the cost of expressiveness and contextual adaptability, making human explanations feel more fluid and accessible despite their lower scores in formal evaluation metrics.

Our experiments with the RAG setup yielded mixed results, as shown in Table 1 and Table 2. While RAG was expected to enhance function understanding and streamline object construction, its effectiveness proved inconsistent. Although retrieving documentation and code examples provided additional context, the results often misaligned with specific use cases. Many retrieved references were too generic or lacked relevance, leading to incorrect function calls and suboptimal parameter choices. These findings align with previous research emphasizing that retrieval quality is crucial. Poorly structured documentation and imprecise retrieval can significantly reduce the effectiveness of RAG-based approaches (Soman and Roychowdhury, 2024).

4.1 Correlation Study

From Table 3, we observe that our proposed metrics show strong alignment with human ratings in Visual Relevance and Element Layout, while demonstrating weaker correlations in Accuracy & Depth, Logical Flow, and Visual Consistency. This suggests that humans are particularly sensitive to visual aspects, such as spatial layouts, but may struggle with evaluating long-form text or audio-based content in detail. Visual Consistency appears to be more subjective, which may explain its relatively lower correlation with human ratings. Additionally, Accuracy & Depth and Logical Flow exhibits the weakest correlation with human judgments, likely due to differences in how LLM and humans assess coherence. Humans can tolerate informal flow, while LLMs may penalize it. On the other hand, human ratings across all dimensions show moderate inter-rater agreement, as indicated by Krippendorff's alpha values. Notably, text-based dimensions achieve slightly higher agreement than visual-based ones, suggesting that textual evaluations are more consistently interpreted among raters.

4.2 Error Analysis

We analyzed the error logs from unsuccessful runs in the TheoremExplainAgent video generation process and identified three primary failure categories. The most common issue was Manim code hallucinations, which accounted for the majority of failures. These errors involved nonexistent functions, modules, object properties, or image assets, as well as incorrect function signatures with invalid parameter types and numbers, reflecting a misunderstanding of the Manim API. The second major issue stemmed from LaTeX rendering errors, primarily due to syntax mistakes and improper handling of special characters in mathematical expressions. Lastly, general coding errors were observed, including missing imports, undefined variables, and computational mistakes in NumPy-based operations. These findings reveal key challenges across LLMs, underscoring the need for better code reliability and API understanding in AI-generated videos.

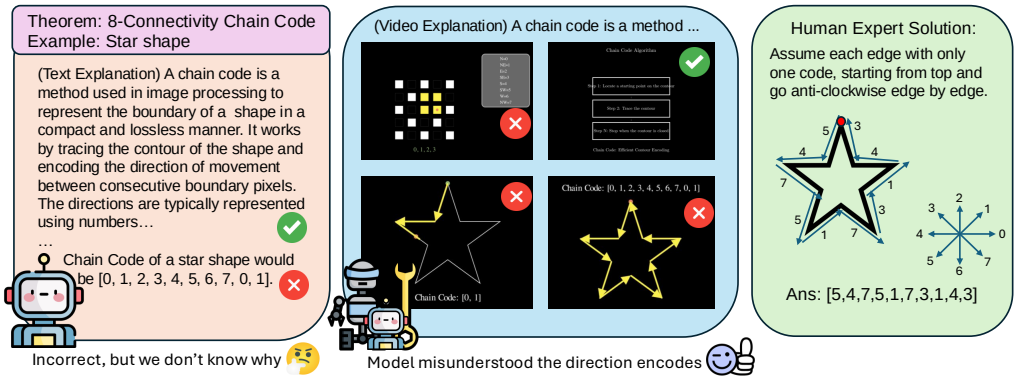

Figure 5: Visualizations expose reasoning errors more clearly than text, making it easier to diagnose model mistakes.

4.3 Case Study

We included representative video outputs in Figure 6. This figure demonstrates that TheoremExplainAgent is capable of generating high-quality exploratory videos. For example, in Mathematics, the model effectively visualizes concepts such as Riemann sums, using animated grids and function plots to illustrate integral approximations. In Chemistry, the system successfully explains the Octet Rule, leveraging atomic models to depict electron sharing and bonding interactions. In Physics, it generates electromagnetic wave simulations, showcasing wave propagation and spectral analysis. In Computer Science, it produces a clear demonstration of Run-Length Encoding, using side-by-side comparisons of raw and compressed data representations. We examined more generated videos carefully and observed that videos in Mathematics, Physics, and Computer Science generally exhibit higher visual quality and coherence compared to those in Chemistry. One notable observation is that Chemistry-related visualizations often rely on simple geometric primitives to depict complex lab apparatus and molecular structures, which can limit their clarity and effectiveness. Additionally, most of the generated videos exhibit minor element layout issues, such as overlapping texts, inconsistent sizes, or suboptimal object positioning, which slightly affects the overall presentation quality, as illustrated in Figure 7.

We also found that visual explanations more effectively reveal reasoning errors than text, facilitating error diagnosis. From Figure 5, we observe that while the text-based explanation allows us to detect that the model's answer is incorrect, it does not provide insight into why the mistake occurred. It seems the model understand the chain code theorem, but it applies it incorrectly. Such explanation is making it difficult to pinpoint the exact reasoning flaw. In contrast, the video-based explanation clearly exposes the model's misunderstanding, as incorrect movement direction encodes and misplaced arrows reveal how the model misinterpreted the chain coding process. This demonstrates that visual explanations not only confirm incorrect reasoning but also uncover the underlying cause of errors, making them a more effective diagnostic tool for analyzing AI-generated outputs.

5 Conclusion

This paper introduces TheoremExplainAgent, a novel agentic approach for generating multimodal theorem explanations through structured video content. Our study demonstrates that integrating visual explanations significantly enhances the clarity and interpretability of theorem reasoning, surpassing text-based methods alone. To systematically evaluate AI-generated explanations, we present a benchmark spanning multiple disciplines with five automated evaluation metrics. Our experiments reveal that agentic planning is crucial for producing long-form, coherent explanations, with o3-mini achieving the highest success rate and overall performance. However, challenges remain in visual element layout, emphasizing the need for improved spatial reasoning and refinement in AI-generated animations. Additionally, our findings underscore the importance of multimodal explanations in identifying reasoning flaws that text-based assessments often miss, reinforcing the role of structured visual communication in AI-driven theorem understanding. Looking ahead, future work should focus on enhancing visual structuring techniques, improving agent coordination, and advancing video understanding to further refine multimodal explanations for LLM-driven theorem comprehension.

6 Limitations

While our approach demonstrates the potential of AI-generated multimodal theorem explanations, several limitations remain. AI models still struggle with complex visual structuring, particularly in consistent elements layout in long-form explanations. Retrieval-augmented generation (RAG) also requires more tokens, increasing computational costs and inference time, which may impact scalability.

7 Potential Risks

AI-generated explanations have the potential to mislead users if errors go undetected, leading to false confidence in incorrect reasoning. This poses a risk where unverified AI-generated content could propagate misconceptions or misinformation if widely disseminated without proper validation. Ensuring the accuracy and reliability of AI-generated explanations remains a critical challenge.

8 Artifacts

We experimented TheoremExplainAgent with GPT-4o (OpenAI, 2023), Gemini 2.0 Flash (DeepMind, 2025), Claude 3.5 v1 (Anthropic, 2024), and o3-mini (OpenAI, 2025). We are releasing the TheoremExplainBench on Huggingface dataset with MIT licence. It features 240 theorems across Computer Science, Physics, Chemistry and Math subjects.

9 Computational Experiments

All the experiments were conducted on a NVIDIA A100-SXM4-80GB GPU. Approximately 1500 US dollars were spent on API call for closed-source model experiments.

References

Talor Abramovich, Meet Udeshi, Minghao Shao, Kilian Lieret, Haoran Xi, Kimberly Milner, Sofija Jancheska, John Yang, Carlos E. Jimenez, Farshad Khorrami, Prashanth Krishnamurthy, Brendan Dolan-Gavitt, Muhammad Shafique, Karthik Narasimhan, Ramesh Karri, and Ofir Press. 2024. Enigma: Enhanced interactive generative model agent for ctf challenges. *Preprint*, arXiv:2409.16165.

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *Preprint*, arXiv:1905.13319.

Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. Online. Accessed: 2025-02-11.

Aristotle. 1901. *Aristotle's Posterior Analytics*. B.H. Blackwell.

Richard Baraniuk. 2025. Openstax: Free textbooks online with no catch.

Jiaqi Chen, Jianheng Tang, Jinghui Qin, Xiaodan Liang, Lingbo Liu, Eric P. Xing, and Liang Lin. 2022a. Geoqa: A geometric question answering benchmark towards multimodal numerical reasoning. *Preprint*, arXiv:2105.14517.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. 2023a. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Preprint*, arXiv:2211.12588.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023b. Theoremqa: A theorem-driven question answering dataset. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. 2022b. Finqa: A dataset of numerical reasoning over financial data. *Preprint*, arXiv:2109.00122.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

DeepMind. 2025. Gemini 2.0 flash. Online. Accessed: 2025-02-11.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Han Bao, Hanwei Xu, Haocheng Wang, Haowei Zhang, Honghui Ding, Huajian Xin, Huazuo Gao, Hui Li, Hui Qu, J. L. Cai, Jian Liang, Jianzhong Guo, Jiaqi Ni, Jiashi Li, Jiawei Wang, Jin Chen, Jingchang Chen, Jingyang Yuan, Junjie Qiu, Junlong Li, Junxiao Song, Kai Dong, Kai Hu, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Xu, Leyi Xia, Liang Zhao, Litong Wang, Liyue Zhang, Meng Li, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Ning Tian, Panpan Huang, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qinyu Chen, Qiushi Du, R. J. Chen, R. L. Jin, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runxin Xu, Ruoyu Zhang, Ruyi Chen, S. S. Li, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shaoqing Wu,

Shengfeng Ye, Shengfeng Ye, Shirong Ma, Shiyu Wang, Shuang Zhou, Shuiping Yu, Shunfeng Zhou, Shuting Pan, T. Wang, Tao Yun, Tian Pei, Tianyu Sun, W. L. Xiao, Wangding Zeng, Wanjia Zhao, Wei An, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, X. Q. Li, Xiangyue Jin, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaojin Shen, Xiaokang Chen, Xiaokang Zhang, Xiaosha Chen, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingkai Yu, Xinnan Song, Xinxia Shan, Xinyi Zhou, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, Y. K. Li, Y. Q. Wang, Y. X. Wei, Y. X. Zhu, Yang Zhang, Yanhong Xu, Yanhong Xu, Yanping Huang, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Li, Yaohui Wang, Yi Yu, Yi Zheng, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Ying Tang, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yu Wu, Yuan Ou, Yuchen Zhu, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yukun Zha, Yunfan Xiong, Yunxian Ma, Yuting Yan, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Z. F. Wu, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhen Huang, Zhen Zhang, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong Shao, Zhipeng Xu, Zhiyu Wu, Zhongyu Zhang, Zhuoshu Li, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Ziyi Gao, and Zizheng Pan. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Joseph L Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

Timur Galimzyanov, Sergey Titov, Yaroslav Golubev, and Egor Bogomolov. 2024. Drawing pandas: A benchmark for llms in generating plotting code. *Preprint*, arXiv:2412.02764.

Gatekeep. 2024. Gatekeep ai: Start learning faster with personalized videos.

Gemini-Team, Machel Reid, Nikolay Savinov, Denis Teplyashin, Dmitry, Lepikhin, Timothy Lillicrap, Jean baptiste Alayrac, Radu Soricut, Angeliki Lazaridou, Orhan Firat, Julian Schrittwieser, Ioannis Antonoglou, Rohan Anil, Sebastian Borgeaud, Andrew Dai, Katie Millican, Ethan Dyer, Mia Glaese, Thibault Sottiaux, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, James Molloy, Jilin Chen, Michael Isard, Paul Barham, Tom Hennigan, Ross McIlroy, Melvin Johnson, Johan Schalkwyk, Eli Collins, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Clemens Meyer, Gregory Thornton, Zhen Yang, Henryk Michalewski, Zaheer Abbas, Nathan Schucher, Ankesh Anand, Richard Ives, James Keeling, Karel Lenc, Salem Haykal, Siamak Shakeri, Pranav Shyam, Aakanksha Chowdhery, Roman Ring, Stephen Spencer, Eren Sezener, Luke Vilnis, Oscar Chang, Nobuyuki Morioka, George Tucker, Ce Zheng, Oliver Woodman, Nithya Attaluri, Tomas Kocisky, Evgenii Eltyshev, Xi Chen, Timothy Chung, Vittorio Selo, Siddhartha Brahma, Petko Georgiev, Ambrose Slone, Zhenkai Zhu, James Lottes, Siyuan Qiao, Ben Caine, Sebastian Riedel, Alex Tomala, Martin Chadwick, Juliette Love, Peter Choy, Sid Mittal, Neil Houlsby, Yunhao Tang, Matthew Lamm, Libin Bai, Qiao Zhang, Luheng He, Yong Cheng, Peter Humphreys, Yujia Li, Sergey Brin, Albin Cassirer, Yingjie Miao, Lukas Zilka, Taylor Tobin, Kelvin Xu, Lev Proleev, Daniel Sohn, Alberto Magni, Lisa Anne Hendricks, Isabel Gao, Santiago Ontanon, Oskar Bunyan, Nathan Byrd, Abhanshu Sharma, Biao Zhang, Mario Pinto, Rishika Sinha, Harsh Mehta, Dawei Jia, Sergi Caelles, Albert Webson, Alex Morris, Becca Roelofs, Yifan Ding, Robin Strudel, Xuehan Xiong, Marvin Ritter, Mostafa Dehghani, Rahma Chaabouni, Abhijit Karmarkar, Guangda Lai, Fabian Mentzer, Bibo Xu, YaGuang Li, Yujing Zhang, Tom Le Paine, Alex Goldin, Behnam Neyshabur, Kate Baumli, Anselm Levskaya, Michael Laskin, Wenhao Jia, Jack W. Rae, Kefan Xiao, Antoine He, Skye Giordano, Lakshman Yagati, Jean-Baptiste Lespiau, Paul Natsev, Sanjay Ganapathy, Fangyu Liu, Danilo Martins, Nanxin Chen, Yunhan Xu, Megan Barnes, Rhys May, Arpi Vezer, Junhyuk Oh, Ken Franko, Sophie Bridgers, Ruizhe Zhao, Boxi Wu, Basil Mustafa, Sean Sechrist, Emilio Parisotto, Thanumalayan Sankaranarayana Pillai, Chris Larkin, Chenjie Gu, Christina Sorokin, Maxim Krikun, Alexey Guseynov, Jessica Landon, Romina Datta, Alexander Pritzel, Phoebe Thacker, Fan Yang, Kevin Hui, Anja Hauth, Chih-Kuan Yeh, David Barker, Justin Mao-Jones, Sophia Austin, Hannah Sheahan, Parker Schuh, James Svensson, Rohan Jain, Vinay Ramasesh, Anton Briukhov, Da-Woon Chung, Tamara von Glehn, Christina Butterfield, Priya Jhakra, Matthew Wiethoff, Justin Frye, Jordan Grimstad, Beer Changpinyo, Charline Le Lan, Anna Bortsova, Yonghui Wu, Paul Voigtlaender, Tara Sainath, Shane Gu, Charlotte Smith, Will Hawkins, Kris Cao, James Besley, Srivatsan Srinivasan, Mark Omernick, Colin Gaffney, Gabriela Surita, Ryan Burnell, Bogdan Damoc, Junwhan Ahn, Andrew Brock, Mantas Pajarskas, Anastasia Petrushkina, Seb Noury, Lorenzo Blanco, Kevin Swersky, Arun Ahuja, Thi Avrahami, Vedant Misra, Raoul de Liedekerke, Mariko Iinuma, Alex Polozov, Sarah York, George van den Driessche, Paul Michel, Justin Chiu, Rory Blevins, Zach Gleicher, Adrià Recasens, Alban Rrustemi, Elena Gribovskaya, Aurko Roy, Wiktor Gworek, Sébastien M. R. Arnold, Lisa Lee, James Lee-Thorp, Marcello Maggioni, Enrique Piqueras, Kartikeya Badola, Sharad Vikram, Lucas Gonzalez, Anirudh Baddepudi, Evan Senter, Jacob Devlin, James Qin, Michael Azzam, Maja Trebacz, Martin Polacek, Kashyap Krishnakumar, Shuo yiin Chang, Matthew Tung, Ivo Penchev, Rishabh Joshi, Kate Olszewska, Carrie Muir, Mateo Wirth, Ale Jakse Hartman, Josh Newlan, Sheleem Kashem, Vijay Bolina, Elahe Dabir, Joost van Amersfoort, Zafarali Ahmed, James Cobon-Kerr, Aishwarya Kamath, Arnar Mar Hrafnkelsson, Le Hou, Ian Mackinnon, Alexandre Frechette, Eric Noland, Xiance Si, Emanuel Taropa, Dong Li, Phil Crone, Anmol Gulati, Sébastien Cevey, Jonas Adler, Ada Ma, David Silver, Simon Tokumine, Richard Powell, Stephan Lee, Ki-

ran Vodrahalli, Samer Hassan, Diana Mincu, Antoine Yang, Nir Levine, Jenny Brennan, Mingqiu Wang, Sarah Hodkinson, Jeffrey Zhao, Josh Lipschultz, Aedan Pope, Michael B. Chang, Cheng Li, Laurent El Shafey, Michela Paganini, Sholto Douglas, Bernd Bohnet, Fabio Pardo, Seth Odoom, Mihaela Rosca, Cicero Nogueira dos Santos, Kedar Soparkar, Arthur Guez, Tom Hudson, Steven Hansen, Chulayuth Asawaroengchai, Ravi Addanki, Tianhe Yu, Wojciech Stokowiec, Mina Khan, Justin Gilmer, Jaehoon Lee, Carrie Grimes Bostock, Keran Rong, Jonathan Caton, Pedram Pejman, Filip Pavetic, Geoff Brown, Vivek Sharma, Mario Lučić, Rajkumar Samuel, Josip Djolonga, Amol Mandhane, Lars Lowe Sjösund, Elena Buchatskaya, Elspeth White, Natalie Clay, Jiepu Jiang, Hyeontaek Lim, Ross Hemsley, Zeyncep Cankara, Jane Labanowski, Nicola De Cao, David Steiner, Sayed Hadi Hashemi, Jacob Austin, Anita Gergely, Tim Blyth, Joe Stanton, Kaushik Shivakumar, Aditya Siddhant, Anders Andreassen, Carlos Araya, Nikhil Sethi, Rakesh Shivanna, Steven Hand, Ankur Bapna, Ali Khodaei, Antoine Miech, Garrett Tanzer, Andy Swing, Shantanu Thakoor, Lora Aroyo, Zhufeng Pan, Zachary Nado, Jakub Sygnowski, Stephanie Winkler, Dian Yu, Mohammad Saleh, Loren Maggiore, Yamini Bansal, Xavier Garcia, Mehran Kazemi, Piyush Patil, Ishita Dasgupta, Iain Barr, Minh Giang, Thais Kagohara, Ivo Danihelka, Amit Marathe, Vladimir Feinberg, Mohamed Elhawaty, Nimesh Ghelani, Dan Horgan, Helen Miller, Lexi Walker, Richard Tanburn, Mukarram Tariq, Disha Shrivastava, Fei Xia, Qingze Wang, Chung-Cheng Chiu, Zoe Ashwood, Khuslen Baatarsukh, Sina Samangooei, Raphaël Lopez Kaufman, Fred Alcober, Axel Stjerngren, Paul Komarek, Katerina Tsihlas, Anudhyan Boral, Ramona Comanescu, Jeremy Chen, Ruibo Liu, Chris Welty, Dawn Bloxwich, Charlie Chen, Yanhua Sun, Fangxiaoyu Feng, Matthew Mauger, Xerxes Dotiwalla, Vincent Hellendoorn, Michael Sharman, Ivy Zheng, Krishna Haridasan, Gabe Barth-Maron, Craig Swanson, Dominika Rogozińska, Alek Andreev, Paul Kishan Rubenstein, Ruoxin Sang, Dan Hurt, Gamaleldin Elsayed, Renshen Wang, Dave Lacey, Anastasija Ilić, Yao Zhao, Adam Iwanicki, Alejandro Lince, Alexander Chen, Christina Lyu, Carl Lebsack, Jordan Griffith, Meenu Gaba, Paramjit Sandhu, Phil Chen, Anna Koop, Ravi Rajwar, Soheil Hassas Yeganeh, Solomon Chang, Rui Zhu, Soroush Radpour, Elnaz Davoodi, Ving Ian Lei, Yang Xu, Daniel Toyama, Constant Segal, Martin Wicke, Hanzhao Lin, Anna Bulanova, Adrià Puigdomènech Badia, Nemanja Rakićević, Pablo Sprechmann, Angelos Filos, Shaobo Hou, Víctor Campos, Nora Kassner, Devendra Sachan, Meire Fortunato, Chimezie Iwuanyanwu, Vitaly Nikolaev, Balaji Lakshminarayanan, Sadegh Jazayeri, Mani Varadarajan, Chetan Tekur, Doug Fritz, Misha Khalman, David Reitter, Kingshuk Dasgupta, Shourya Sarcar, Tina Ornduff, Javier Snaider, Fantine Huot, Johnson Jia, Rupert Kemp, Nejc Trdin, Anitha Vijayakumar, Lucy Kim, Christof Angermueller, Li Lao, Tianqi Liu, Haibin Zhang, David Engel, Somer Greene, Anaïs White, Jessica Austin, Lilly Taylor, Shereen Ashraf, Dangyi Liu, Maria Georgaki, Irene Cai, Yana Kulizhskaya, Sonam Goenka, Brennan Saeta, Ying Xu, Christian Frank, Dario de Cesare, Brona Robenek, Harry Richardson, Mahmoud Alnahlawi, Christopher Yew, Priya Ponnapalli, Marco Tagliasacchi, Alex Korchemniy, Yelin Kim, Dinghua Li, Bill Rosgen, Kyle Levin, Jeremy Wiesner, Praseem Banzal, Praveen Srinivasan, Hongkun Yu, Çağlar Ünlü, David Reid, Zora Tung, Daniel Finchelstein, Ravin Kumar, Andre Elisseeff, Jin Huang, Ming Zhang, Ricardo Aguilar, Mai Giménez, Jiawei Xia, Olivier Dousse, Willi Gierke, Damion Yates, Komal Jalan, Lu Li, Eri Latorre-Chimoto, Duc Dung Nguyen, Ken Durden, Praveen Kallakuri, Yaxin Liu, Matthew Johnson, Tomy Tsai, Alice Talbert, Jasmine Liu, Alexander Neitz, Chen Elkind, Marco Selvi, Mimi Jasarevic, Livio Baldini Soares, Albert Cui, Pidong Wang, Alek Wenjiao Wang, Xinyu Ye, Krystal Kallarackal, Lucia Loher, Hoi Lam, Josef Broder, Dan Holtmann-Rice, Nina Martin, Bramandia Ramadhana, Mrinal Shukla, Sujoy Basu, Abhi Mohan, Nick Fernando, Noah Fiedel, Kim Paterson, Hui Li, Ankush Garg, Jane Park, DongHyun Choi, Diane Wu, Sankalp Singh, Zhishuai Zhang, Amir Globerson, Lily Yu, John Carpenter, Félix de Chaumont Quitry, Carey Radebaugh, Chu-Cheng Lin, Alex Tudor, Prakash Shroff, Drew Garmon, Dayou Du, Neera Vats, Han Lu, Shariq Iqbal, Alex Yakubovich, Nilesh Tripuraneni, James Manyika, Haroon Qureshi, Nan Hua, Christel Ngani, Maria Abi Raad, Hannah Forbes, Jeff Stanway, Mukund Sundararajan, Victor Ungureanu, Colton Bishop, Yunjie Li, Balaji Venkatraman, Bo Li, Chloe Thornton, Salvatore Scellato, Nishesh Gupta, Yicheng Wang, Ian Tenney, Xihui Wu, Ashish Shenoy, Gabriel Carvajal, Diana Gage Wright, Ben Bariach, Zhuyun Xiao, Peter Hawkins, Sid Dalmia, Clement Farabet, Pedro Valenzuela, Quan Yuan, Ananth Agarwal, Mia Chen, Wooyeol Kim, Brice Hulse, Nandita Dukkipati, Adam Paszke, Andrew Bolt, Kiam Choo, Jennifer Beattie, Jennifer Prendki, Harsha Vashisht, Rebeca Santamaria-Fernandez, Luis C. Cobo, Jarek Wilkiewicz, David Madras, Ali Elqursh, Grant Uy, Kevin Ramirez, Matt Harvey, Tyler Liechty, Heiga Zen, Jeff Seibert, Clara Huiyi Hu, Andrey Khorlin, Maigo Le, Asaf Aharoni, Megan Li, Lily Wang, Sandeep Kumar, Norman Casagrande, Jay Hoover, Dalia El Badawy, David Soergel, Denis Vnukov, Matt Miecnikowski, Jiri Simsa, Praveen Kumar, Thibault Sellam, Daniel Vlasic, Samira Daruki, Nir Shabat, John Zhang, Guolong Su, Jiageng Zhang, Jeremiah Liu, Yi Sun, Evan Palmer, Alireza Ghaffarkhah, Xi Xiong, Victor Cotruta, Michael Fink, Lucas Dixon, Ashwin Sreevatsa, Adrian Goedeckemeyer, Alek Dimitriev, Mohsen Jafari, Remi Crocker, Nicholas FitzGerald, Aviral Kumar, Sanjay Ghemawat, Ivan Philips, Frederick Liu, Yannie Liang, Rachel Sterneck, Alena Repina, Marcus Wu, Laura Knight, Marin Georgiev, Hyo Lee, Harry Askham, Abhishek Chakladar, Annie Louis, Carl Crous, Hardie Cate, Dessie Petrova, Michael Quinn, Denese Owusu-Afriyie, Achintya Singhal, Nan Wei, Solomon Kim, Damien Vincent, Milad Nasr, Christopher A. Choquette-Choo, Reiko Tojo, Shawn Lu, Diego de Las Casas, Yuchung Cheng, Tolga Bolukbasi, Katherine Lee, Saaber

Fatehi, Rajagopal Ananthanarayanan, Miteyan Patel, Charbel Kaed, Jing Li, Shreyas Rammohan Belle, Zhe Chen, Jaclyn Konzelmann, Siim Põder, Roopal Garg, Vinod Koverkathu, Adam Brown, Chris Dyer, Rosanne Liu, Azade Nova, Jun Xu, Alanna Walton, Alicia Parrish, Mark Epstein, Sara McCarthy, Slav Petrov, Demis Hassabis, Koray Kavukcuoglu, Jeffrey Dean, and Oriol Vinyals. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

GenerativeManim. 2024. Generative manim: Ai-driven animations for mathematics and education.

Kanika Goswami, Puneet Mathur, Ryan Rossi, and Franck Dernoncourt. 2025. Plotgen: Multi-agent llm-based scientific data visualization via multimodal feedback. *Preprint*, arXiv:2502.00988.

Liu He, Yizhi Song, Hejun Huang, Daniel Aliaga, and Xin Zhou. 2024. Kubrick: Multimodal agent collaborations for synthetic video generation. *Preprint*, arXiv:2408.10453.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *NeurIPS*.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. 2023. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. *arXiv preprint arXiv:2303.11897*.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik R Narasimhan. 2024. SWE-bench: Can language models resolve real-world github issues? In *The Twelfth International Conference on Learning Representations*.

Aryan Keluskar, Amrita Bhattacharjee, and Huan Liu. 2024. Do llms understand ambiguity in text? a case study in open-world question answering. *Preprint*, arXiv:2411.12395.

Klaus Krippendorff. 2011. Computing krippendorff's alpha-reliability.

Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. 2023. Viescore: Towards explainable metrics for conditional image synthesis evaluation. *Preprint*, arXiv:2312.14867.

Delmar Larsen. 2025. Libretexts: The future is open.

Guozheng Li, Xinyu Wang, Gerile Aodeng, Shunyuan Zheng, Yu Zhang, Chuangxin Ou, Song Wang, and Chi Harold Liu. 2024. Visualization generation with large language models: An evaluation. *Preprint*, arXiv:2401.11255.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. 2024. The AI Scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*.

Pan Lu, Swaroop Mishra, Tony Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. In *The 36th Conference on Neural Information Processing Systems (NeurIPS)*.

Erik Nijkamp, Bo Pang, Hiroaki Hayashi, Lifu Tu, Huan Wang, Yingbo Zhou, Silvio Savarese, and Caiming Xiong. 2023. Codegen: An open large language model for code with multi-turn program synthesis. *Preprint*, arXiv:2203.13474.

OpenAI. 2023. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI. 2025. Openai o3 mini. Online. Accessed: 2025-02-11.

Pouya Pezeshkpour and Estevam Hruschka. 2023. Large language models sensitivity to the order of options in multiple-choice questions. *Preprint*, arXiv:2308.11483.

Daniel Ritchie, Paul Guerrero, R. Kenny Jones, Niloy J. Mitra, Adriana Schulz, Karl D. D. Willis, and Jiajun Wu. 2023. Neurosymbolic models for computer graphics. *Preprint*, arXiv:2304.10320.

G. [3Blue1Brown] Sanderson. 2020. Group theory, abstraction, and the 196,883-dimensional monster. YouTube.

Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Zicheng Liu, and Emad Barsoum. 2025. Agent laboratory: Using llm agents as research assistants. *Preprint*, arXiv:2501.04227.

Krish Shah, Chris Abey, and Hargun Mujral. 2024. 3brown1blue: Ai-generated educational videos with manim.

Chenglei Si, Diyi Yang, and Tatsunori Hashimoto. 2024. Can llms generate novel research ideas? a large-scale human study with 100+ nlp researchers. *Preprint*, arXiv:2409.04109.

Sumit Soman and Sujoy Roychowdhury. 2024. Observations on building rag systems for technical documents. *Preprint*, arXiv:2404.00657.

The Manim Community Developers. 2024. Manim – Mathematical Animation Framework.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Tianbao Xie, Danyang Zhang, Jixuan Chen, Xiaochuan Li, Siheng Zhao, Ruisheng Cao, Toh Jing Hua, Zhoujun Cheng, Dongchan Shin, Fangyu Lei, Yitao Liu,

Yiheng Xu, Shuyan Zhou, Silvio Savarese, Caiming Xiong, Victor Zhong, and Tao Yu. 2024. Osworld: Benchmarking multimodal agents for open-ended tasks in real computer environments. *Preprint*, arXiv:2404.07972.

John Yang, Carlos E. Jimenez, Alex L. Zhang, Kilian Lieret, Joyce Yang, Xindi Wu, Ori Press, Niklas Muennighoff, Gabriel Synnaeve, Karthik R. Narasimhan, Diyi Yang, Sida I. Wang, and Ofir Press. 2024a. Swe-bench multimodal: Do ai systems generalize to visual software domains? *Preprint*, arXiv:2410.03859.

Zhiyu Yang, Zihan Zhou, Shuo Wang, Xin Cong, Xu Han, Yukun Yan, Zhenghao Liu, Zhixing Tan, Pengyuan Liu, Dong Yu, Zhiyuan Liu, Xiaodong Shi, and Maosong Sun. 2024b. Matplotagent: Method and evaluation for llm-based agentic scientific data visualization. *Preprint*, arXiv:2402.11453.

Sharon Zhang, Jiaju Ma, Jiajun Wu, Daniel Ritchie, and Maneesh Agrawala. 2023a. Editing motion graphics video via motion vectorization and transformation. *ACM Trans. Graph.*

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. 2023b. Multimodal chain-of-thought reasoning in language models. *arXiv preprint arXiv:2302.00923*.

Ziyin Zhang, Zhaokun Jiang, Lizhen Xu, Hongkun Hao, and Rui Wang. 2024. Multiple-choice questions are efficient and robust llm evaluators. *Preprint*, arXiv:2405.11966.

Zihao Zhou, Maizhen Ning, Qiufeng Wang, Jie Yao, Wei Wang, Xiaowei Huang, and Kaizhu Huang. 2023. Learning by analogy: Diverse questions generation in math word problem. *Preprint*, arXiv:2306.09064.

A Gallery

In Figure 6 we present the high-quality videos generated by TheoremExplainAgent across four STEM domains. The images are extracted from different scenes in the videos, showing the consistency of the topic. In Figure 7 we present the poorly generated videos from TheoremExplainAgent and examine their artifacts. In Figure 8 we compare a high quality animation and a low quality animation, and how they were rated with our proposed metric.

B Prompt Templates

We adapt Chain-of-Thoughts (CoT) (Wei et al., 2023) and Program-of-Thoughts (PoT) (Chen et al., 2023a) when we design the prompt for TheoremExplainAgent. We present our prompts templates in the end of the Appendix.

C Supplementary Information

C.1 Human Annotation Process

We recruited 12 student volunteers in our annotation process. We explained to the annotators that their annotations were to be used in our study only and would not be released publicly.

We show the user interface of our annotation website in Figure 9, including the instructions presented to our annotators. We supplement each of the dimensions with guiding questions to clarify what the annotators should score.

C.2 Runtime Statistics

We report the runtime and cost statistics in Table 4, assuming 4 fixed codes and 7 scenes per video, we evaluate the cost, inference time, and latency of different language models, and find that the Claude 3.5-Sonnet v1 model has the longest inference time (2240-2380s), while the Gemini 2.0-Flash and GPT-4o are the fastest (around 1120s).The RAG integration increases the number of input tokens significantly. RAG integration significantly increases the number of input tokens, with Claude 3.5-Sonnet v1 + RAG being the most used (1,050,000). Output tokens are less variable, with the o3-mini model generating the most tokens (154,000). The Gemini 2.0-Flash model is the most cost-effective ($0.10-$0.16), while the Claude 3.5-Sonnet v1 + RAG is the most expensive ($4.67).

C.3 Potentials for Future Research

Recent community efforts (Shah et al., 2024; Gatekeep, 2024; GenerativeManim, 2024) have explored AI-driven Manim-based video generation for educational purposes. However, no scientific studies have systematically evaluated the effectiveness and robustness of these approaches. Our work introduces a novel agentic framework for generating multimodal theorem explanations and demonstrates that AI-generated videos can achieve performance comparable to human-made content, although the robustness is still limited. Nevertheless, further research is needed to assess their impact on AI's reasoning capabilities, visualization quality, and learning outcomes. Future directions include establishing benchmarks for AI-generated educational videos (within EdTech), integrating interactive elements to enhance engagement (within HCI/Visualization), and refining evaluation metrics to assess LLMs' multimodal explanation abilities (within NLP).

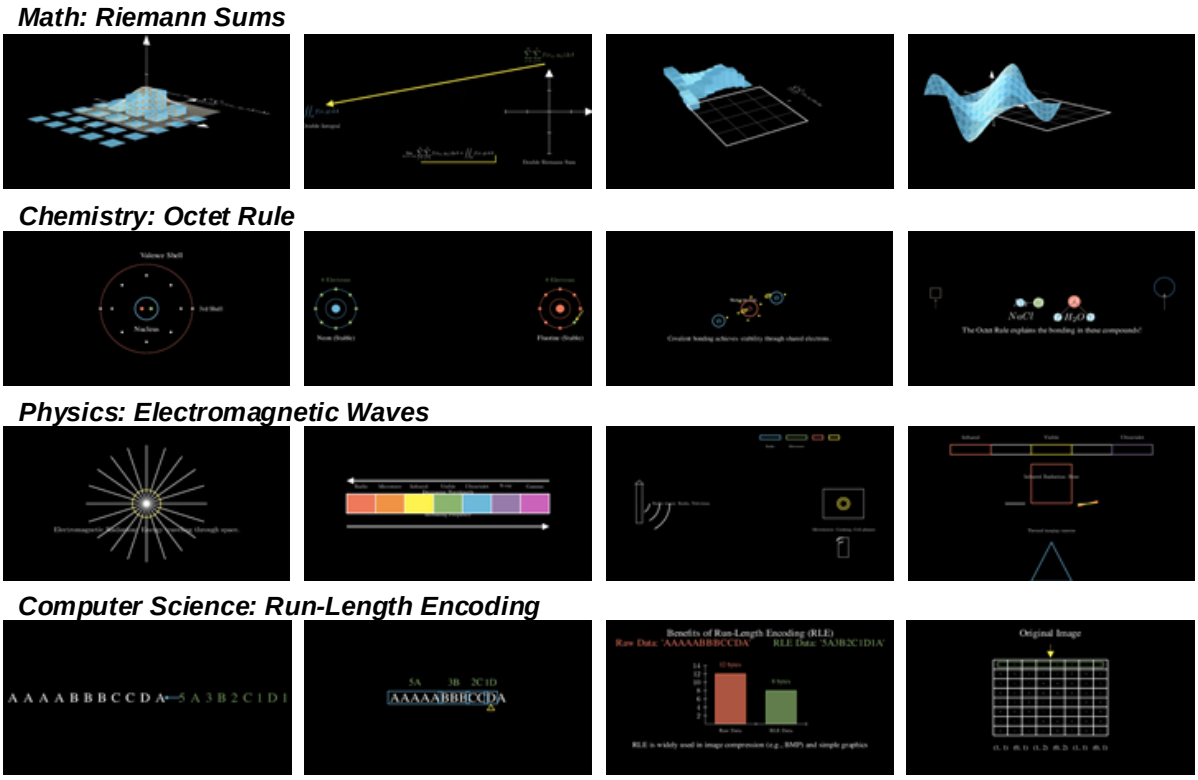

Figure 6: We show the high-quality videos generated by TheoremExplainAgent,across the four STEM domains.

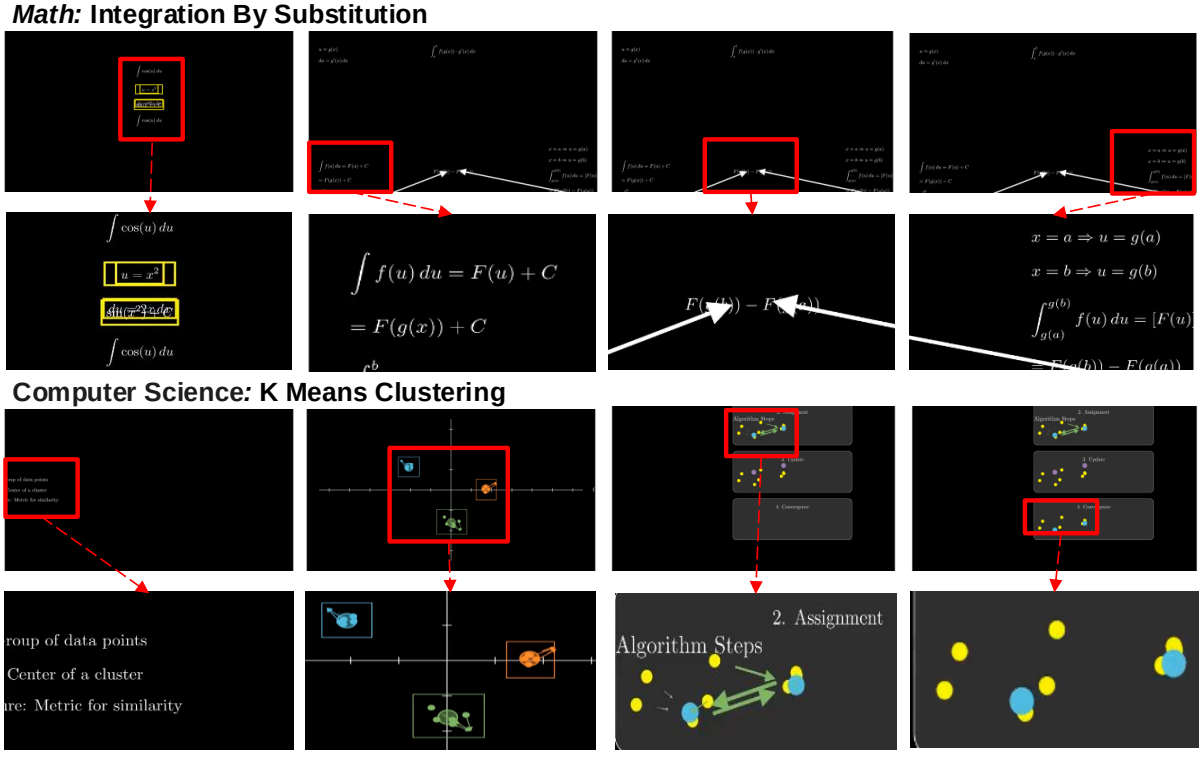

Figure 7: We show the poorly generated videos from TheoremExplainAgent, zooming in the artifacts.

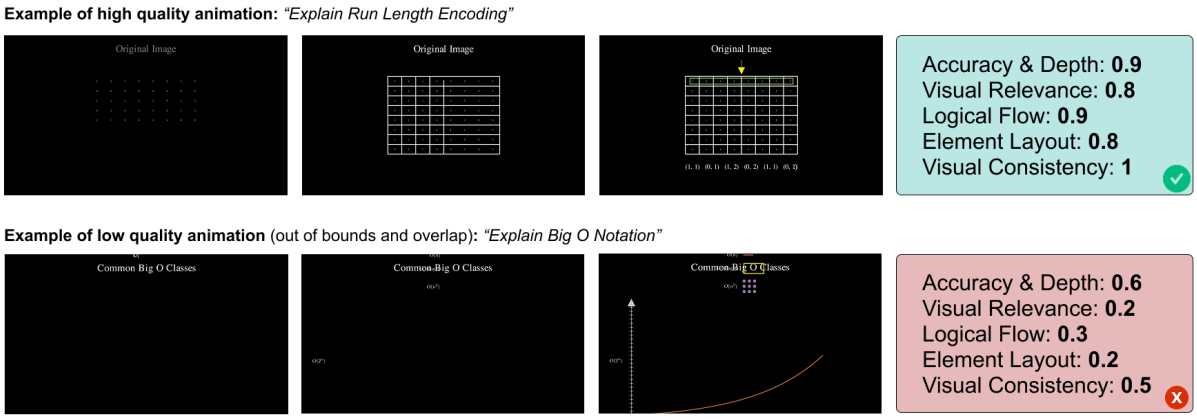

Figure 8: Comparison on a scene of a high quality animation and a low quality animation.

Scene Plan Generation Prompt Template

You are an expert in video production, instructional design, and {topic}. Please design a high-quality video to provide in-depth explanation on {topic}.

Video Overview:
Topic: {topic}
Description: {description}

Scene Breakdown:
Plan individual scenes. For each scene please provide the following:

- Scene Title: Short, descriptive title (2-5 words).
- Scene Purpose: Objective of this scene. How does it connect to previous scenes?
- Scene Description: Detailed description of scene content.
- Scene Layout: Detailed description of the spatial layout concept. Consider safe area margins and minimum spacing between objects.

Please generate the scene plan for the video in the following format: ...

Agent	Input Tokens	Output Tokens	Cost(USD)	Time(s)
GPT-4o	350000	84000	1.71	1120
GPT-4o + RAG	840000	84000	2.94	1260
Claude 3.5-Sonnet v1	350000	91000	2.42	2240
Claude 3.5-Sonnet v1 + RAG	1050000	101500	4.67	2380
Gemini 2.0-Flash	595000	119000	0.1	1120
Gemini 2.0-Flash + RAG	1120000	119000	0.16	1260
o3-mini (medium)	434000	154000	1.16	1680
o3-mini (medium) + RAG	945000	154000	1.72	1820

Table 4: Average output tokens, cost, and inference time for TheoremExplainAgent generating one full video.

Code Generation Prompt Template

You are an expert Manim (Community Edition) developer. Generate executable Manim code implementing animations as specified, strictly adhering to the provided Manim documentation context, technical implementation plan, animation and narration plan, and all defined spatial constraints.

Think of reusable animation components for a clean, modular, and maintainable library, prioritizing code structure and best practices as demonstrated in the Manim documentation context. Throughout code generation, rigorously validate all spatial positioning and animations against the defined safe area margins and minimum spacing constraints. If any potential constraint violation is detected, generate a comment in the code highlighting the issue for manual review and correction.

Input Context:
...

Code Generation Guidelines:
...

Code Fixing Prompt Template

You are an expert Manim developer specializing in debugging and error resolution. Based on the provided implementation plan and Manim code, analyze the error message to provide a comprehensive fix and explanation.

Implementation Plan: {implementation_plan}

Manim Code: {manim_code}

Error Message: {error_message}

Requirements:

1. Provide complete error analysis with specific line numbers where possible.
2. Include exact instructions for every code change.
3. Explain why the error occurred in plain language.
4. ...

Evaluation Prompt Template

You are a specialist in evaluating theorem explanation videos, known for giving clear and objective feedback. You will be given the transcript of a video. Your task is to evaluate and score the content of the video in several dimensions.

Evaluation Criteria:

1. **Accuracy and Depth**
    - Does the narration explain the theorem accurately?
    - Does the video provide intuitive and/or rigorous explanations for why the theorem holds?
2. **Logical Flow**
    - Does the video follow a clear and logical structure?
    - Does the video present a coherent buildup of ideas?

Scoring Instructions:
Conduct a comprehensive evaluation and score each dimension from 0 to 1:
(Score Descriptions)

You are tasked with analyzing and scoring a frame taken from a theorem explanation video. Note that you may not have the context of the video, so the captured frame may be a frame where some motion of visual elements is taking place. Your job is to assign a score from 1 to 5 for each criterion. Please provide a brief justification for your scores.

Evaluation Criteria:

1. **Visual Relevance**
    - Does the video frame align with the theorem's concepts and derivations?
2. **Element Layout**
    - Are the visual elements well-placed and appropriately sized within the frame?
    - Are the visual elements free of unintentional overlap?
    - Is the visual information conveyed in the frame clear and easy to understand?

...

You are tasked with analyzing and scoring a chunk of a theorem explanation video. Note that you may not have the full context of the video. Your job is to assign a score from 1 to 5 for each criterion. Please provide a brief justification for your scores.

Evaluation Criteria:

1. **Visual Consistency**
    - Does the visual style remain consistent across frames?
    - Are the motions and transitions smooth?

...

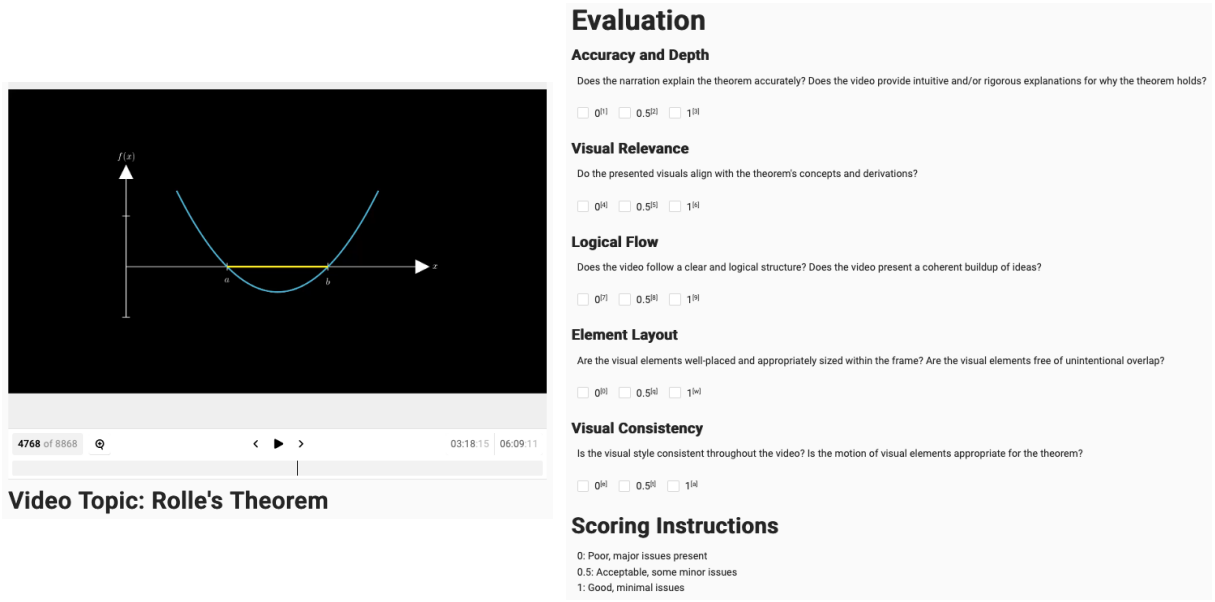

Figure 9: The user interface of our annoatation website.