ViStruct: Simulating Expert-Like Reasoning Through Task Decomposition and Visual Attention

Oliver Huang [*]
University of Toronto

Carolina Nobre [†]
University of Toronto

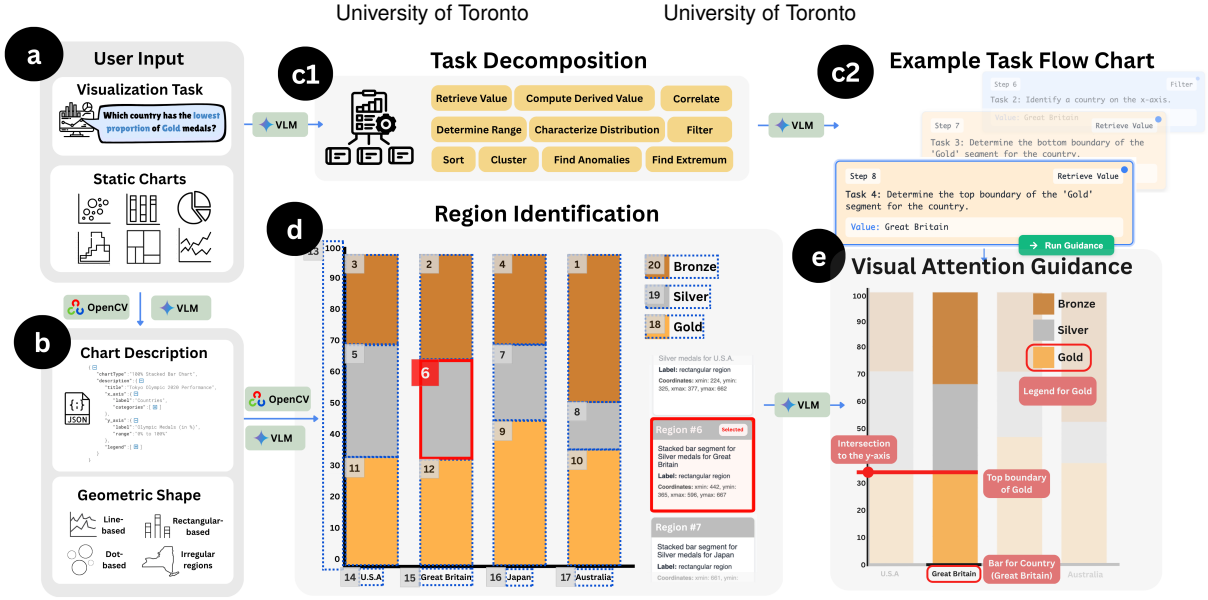

Figure 1: **A pipeline overview of ViStruct.** *(a)* User provides a visualization task description and static charts. *(b)* Behind the scenes, the system generates a detailed textual description of the chart and identifies its geometric elements. *(c1)* The high-level task is decomposed into a sequence of low-level component subtasks. *(c2)* An example shows how the original task is broken down into a step-by-step reasoning flow. *(d)* The system detects and labels chart regions based on geometric features and spatial relationships.*(e)* The chart is automatically annotated with task-relevant Areas of Interest (AOIs) to guide user attention.

Abstract

Data visualization tasks often require multi-step reasoning, and the interpretive strategies experts use—such as decomposing complex goals into smaller subtasks and selectively attending to key chart regions—are rarely made explicit. We developed ViStruct as an automated pipeline that simulates these expert behaviours by breaking high-level visual questions into structured analytic steps and highlighting semantically relevant chart areas. Leveraging large language and vision-language models, we evaluate the system on 45 tasks across 12 chart types and validate its outputs with trained visualization users, confirming its ability to produce interpretable and expert-aligned reasoning sequences.

Keywords: Data Visualization, Task Decomposition, Large Language Models(LLMs), Guidance System, Computer Vision

1 Introduction

Decomposing a complex task into smaller, manageable components is widely recognized as an effective strategy for problem-solving [5, 14, 18, 28]. In many domains, expert users perform such decomposition intuitively, applying structured strategies to reason through problems efficiently. However, these strategies are often implicit and difficult to observe, replicate, or teach [20]. Consequently, by simulating how experts break down complex tasks, we can externalize these internal reasoning patterns and make them accessible for analysis, instruction, or further automation [13, 27].

Data visualization tasks are a prime example of this need. They involve multiple cognitive stages and high-level goals [15], such as identifying trends or comparing proportions, which are typically achieved through a sequence of low-level perceptual and analytic operations [4]. Experts, defined as experienced users with demonstrated fluency in interpreting standard chart types, typically perform these reasoning steps fluidly and implicitly. Consequently, their structured reasoning remains hidden, making it challenging to study, explain, or teach [12, 26]. This highlights a significant gap in capturing and modelling expert reasoning processes; bridging this gap is crucial for developing more interpretable visualization systems and laying the foundations for educational tools.

Despite this need, generating expert-level reasoning for visualization analysis remains challenging. Task decomposition must be tailored to the structure and semantics of each chart, and each analytic step must be precisely linked to the relevant visual regions. Manually crafting these reasoning is infeasible at scale due to the wide variability in visualization goals, subtasks, and chart designs. Although AI-driven approaches have successfully decomposed complex tasks in other domains [24, 25], their potential for decomposing visualization-specific tasks and delivering precise visual attention guidance remains largely unexplored.

To bridge this gap, we introduce **ViStruct**, an automated pipeline that simulates expert-like visual reasoning through structured task decomposition and region-based visual attention. ViStruct leverages large-language (LLM) and vision-language models (VLM) to systematically **decompose visualization tasks**, dynamically **identify context-specific AOIs**, and effectively **produce actionable instructional sequences** tailored to the structure and semantics of the chart.

[*]e-mail: oliver@cs.toronto.edu
[†]e-mail: cnobre@cs.toronto.edu

To validate the scalability of the proposed technique, we applied it to 45 visualization tasks across 12 chart types. We evaluated our approach with 20 domain experts, confirming the expert-like guidance and transparency provided by ViStruct. The resulting technique is publicly available as an open-source interactive platform for researchers and practitioners, accessible here.

2 Related Work

Our work draws upon related efforts in AI-driven reasoning frameworks for task decomposition and in vision-language approaches for spatial and semantic understanding of data visualizations.

2.1 AI Assisted Task Decomposition

Previous research has investigated how AI can assist in task decomposition. Techniques like Chain-of-Thought prompting [29] and ReAct [31] help structure reasoning through sequential steps and action-based feedback. These methods are often embedded in systems such as Talk2Data [8] and LightVA [34] to support interactive visual analysis and analytic planning. Additionally, some approaches emphasize user control and trust by letting users refine or verify the decomposition through interactive steps [14], promoting transparency and interpretability.

In the context of data visualization, explainable AI frameworks [1] and workflow automation tools [3] follow a reactive paradigm, where users must explicitly request task decomposition or guidance. This reactive design places the burden of initiative and strategy on the user, which deviates from how experts naturally guide others through analysis. Rather than waiting for user prompts, ViStruct anticipates the visual analytic process by automatically breaking down the visualization task and highlighting relevant chart regions in a meaningful order, mirroring the reasoning that experts employ.

2.2 Region-Aware Processing in Data Visualization

Recent work reveals that while VLMs show some capability in interpreting chart structures and high-level relationships [10], they often lack consistency and robustness [19]. Many models struggle with understanding visual language and fail to capture relational information accurately, which is critical for interpreting charts [9]. Even advanced models still face challenges in reliably extracting meaningful visual relationships.

Newer approaches emphasize region-based understanding. MapReader [33] demonstrates how spatial visualizations benefit from region-level segmentation, while intermediate text representations [32] have been used to improve flowchart comprehension. With the help of OpenCV, the Chain-of-Region technique [16] proposes explicitly segmenting charts into interpretable regions (axes, bars, legends, etc) and combining this with VLMs enables precise coordinate-to-region mapping, which is crucial for tasks like identifying the correct values in bar or line charts.

ViStruct leverages region-based techniques by detecting semantically meaningful regions within charts and defining task-specific AOIs. Motivated by sequential visual cues (SVCs) [23], which guide attention through critical regions in order, ViStruct integrates region segmentation directly into VLMs, aligning visual cues with each step of the task decomposition.

3 Design Goals

In designing ViStruct, we aimed to simulate expert-like reasoning in visualization tasks, particularly how experts interpret charts step by step and focus on relevant visual regions. To derive these design goals, we extensively reviewed visualization and cognitive science literature and analyzed known distinctions in how experts and novices approach visual reasoning. Prior studies show that experts systematically attend to semantically meaningful regions, interpret visual encodings through structured reasoning sequences, and integrate spatial and semantic cues to guide their analysis [6, 17, 21, 22]. The design goals for ViStruct are:

G1: Semantic Region Understanding During the encoding stage of visualization comprehension, experts naturally identify and interpret distinct semantic regions of a chart (axes, data marks, and labels) to extract relevant visual information [6]. Effective encoding requires accurately understanding each region's roles within the visual structure [21]. Therefore, the system must detect these components accurately and assign labels meaningfully reflecting each component's function.

G2: Precise Coordinate Mapping In the decoding stage, the visual elements identified must be translated into their corresponding quantitative meanings. This process involves precisely mapping spatial features, such as the height of bars or the positions of data points, to numerical values through alignment with reference regions and axis scales [21]. The system must support this decoding step so that users can interpret the quantitative information embedded within the spatial arrangement of chart components.

G3: Structured Task Decomposition Experts often approach visualization tasks by intuitively breaking them down into smaller analytic steps without making this process explicit [4]. This decomposition happens internally and fluidly, informed by experience and familiarity with visual encoding strategies [2]. ViStruct aims to simulate this expert behaviour by externalizing the reasoning path: it generates a structured sequence of subtasks that mirrors the analytic flow an expert might follow.

G4: Supporting Diverse Data Encoding Charts use a wide range of symbols, layouts, and dimensional mappings to encode data; each may call for a different approach to interpretation [7]. To support diverse chart types and visualization tasks, the system must accurately classify chart components by integrating information from both textual descriptions and geometric shapes.

G5: Interpretability and Transparency Every stage of the reasoning process should be clearly explained to improve interpretability. This includes how regions are divided, what data is extracted, and why certain decisions are made. ViStruct enhances this process using visual attention guidance [11, 30], such as highlights and circled AOIs, to direct users toward task-relevant areas. These visual cues allow users to follow the model's interpretation path.

4 ViStruct

Considering the design requirements, we developed ViStruct as a prototype system to explore automated, expert-like reasoning of visualizations through chart decomposition and visual guidance. This section outlines the ViStruct pipeline and its integration, beginning with a user-facing scenario and then detailing each system component: chart characterization, task decomposition, region annotation, and attention-guided output. For each stage, we describe how it supports the design goals introduced in Section 3.

ViStruct is implemented as a prototype interactive platform using TypeScript and the Next.js framework, with a backend powered by OpenCV for region detection. The system is model-agnostic and adaptable to different vision-language models. In our initial experiments,We tested Gemini-2-Flash[1], GPT-4V[2], and Claude 3.7[3]. on structured chart inputs at each stage of the pipeline. We selected Gemini-2-Flash for integration due to its faster response time and more reliable performance in understanding visual elements.

4.1 Overview and Usage Scenario

ViStruct simulates how chart-literate experts approach visual reasoning tasks, especially those in visual literacy assessments such as VLAT[4]. These tasks are intended not to extract novel insights but to train users in structured interpretation strategies that experts implicitly use. ViStruct externalizes these strategies through interactive, step-by-step visual guidance.

[1] Gemini-2-Flash: https://deepmind.google/technologies/gemini/flash
[2] GPT-4V: https://openai.com/research/gpt-4v-system-card
[3] Claude 3.7: https://www.anthropic.com/news/claude-3-7-sonnet
[4] VLAT: https://bckwon.pythonanywhere.com/

Given a static chart and a user-defined visualization question, ViStruct processes the input through multiple stages (Fig.1). It begins by identifying the chart type and extracting structural elements (e.g., bars, axes, labels) using OpenCV and Gemini-2-Flash (Sec.4.2), supporting **G1: Semantic Region Understanding**. This structured representation is passed to an LLM-driven pipeline to decompose the question into low-level analytic subtasks (Sec.4.3), supporting **G3: Structured Task Decomposition**. Subtasks are organized into a coherent, editable flow (Sec.4.4), ensuring transparency (**G5**). Concurrently, precise visual regions associated with each subtask are identified and annotated (Sec. 4.5), enabling accurate coordinate mapping (**G2**). Users progress step-by-step with interactive visual cues highlighting these Areas of Interest (AOIs).

To demonstrate how ViStruct operates in practice, we present a scenario inspired by the VLAT questionnaire, where a user analyzes a chart of Olympic medals (see Fig. 1). The chart is a 100% stacked bar chart that displays each country's distribution of medals: gold, silver, and bronze. The user answers the question: *"Which country has the smallest proportion of gold medals?"* Although this question may seem straightforward, answering it visually involves several steps. The user must identify the gold medal segments, estimate their relative height in proportion to the total height of the bar, and make comparisons across different countries while disregarding other visually present but irrelevant data.

ViStruct automatically parses the chart into semantic regions such as bars, axes, and legends, supporting **G1**. Users can explore the chart interactively (e.g., hovering over a segment shows: *Gold medal count for USA*). The system decomposes the task into a structured flow (e.g., isolate gold segments, read segment heights, normalize by bar total, compare proportions), surfacing the kind of stepwise reasoning that experts typically perform internally, supporting **G3**. To assist decoding, the system draws boundary lines and reference overlays (Fig. 1e), enabling accurate interpretation of visual encodings (**G2**). Finally, each subtask is editable, and the whole reasoning sequence is transparent, supporting **G5**.

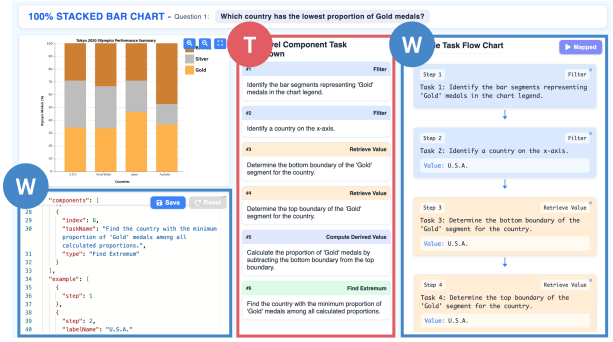

Figure 2: **(T) Task Decomposition:** The system presents a low-level component task breakdown derived from the user's high-level question, categorized by task type. **(W) Editable Workflow Interface:** Users can view and manipulate the task structure and customize the workflow, selecting variable instances and reordering steps.

4.2 Chart Characterization

ViStruct begins by characterizing the input chart to extract its structural and semantic layout as a foundation for downstream reasoning. Gemini-2-Flash organizes them into a structured JSON representation, including axis ranges, variable names, data groupings, and encoding types.

To improve the reliability of OpenCV-based region detection, graphical elements are categorized into four shape types: line-based, dot-based, rectangular, and irregular. This classification allows ViStruct to adapt region analysis methods to the chart's visual encoding scheme. Chart characterization supports **G4** by enabling ViStruct to generalize across diverse chart types and visual conventions.

4.3 Task Decomposition

ViStruct uses a multi-stage prompting framework with three sequential prompts to decompose high-level user tasks into low-level analytic subtasks. The LLM operates on a structured input that includes (1) the user's natural language query, (2) the JSON chart description (Sec. 4.2), and (3) a list of labelled chart regions with spatial metadata.

Decomposition is guided by a predefined taxonomy of ten low-level task types [4]. The first prompt produces a breakdown based on this taxonomy. The second refines each step to ensure it is grounded in specific, executable, and sufficiently detailed chart regions. If a step is too abstract, the LLM splits it into atomic components. The third prompt validates the structure, resolves step dependencies, and removes redundant subtasks (e.g., repeating the same operation across different chart elements).

This breakdown–refine–verify process ensures each step is precise, executable, and aligned with the task structure. It directly supports **G3** by enabling structured task decomposition and **G4** by adapting reasoning to diverse chart types and visual encodings.

4.4 Decomposition Flow Example

Due to visual analysis's compositional nature, there is often no correct way to sequence low-level operations. ViStruct presents one example workflow based on the decomposition results (Fig. 2), illustrating a valid approach. Recognizing that users may prefer different reasoning paths, the system allows them to modify the sequence, adjust parameters, or reorganize tasks to fit their analysis strategy better. The LLM then validates the revised flow to ensure logical consistency. If valid, both the workflow and the underlying decomposed tasks are updated accordingly.

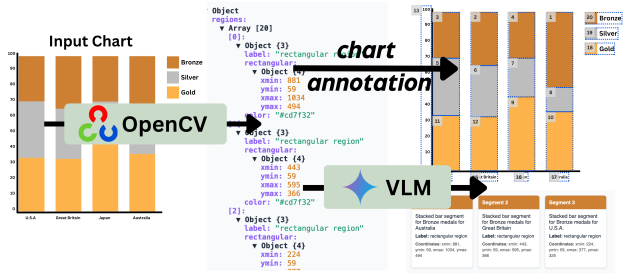

Figure 3: **Workflow of region identification**: The input chart is processed using OpenCV to identify distinct regions corresponding to chart segments. These regions are labeled by the system. VLM then identifies each annotated region with meaningful descriptions.

4.5 Region Identification

Region identification is a key step in the ViStruct pipeline, providing the precise coordinate-level data needed for visual guidance and accurate annotation. To highlight AOIs during task execution, the system must first detect all meaningful chart components, including axes, labels, data marks, and legends.

The process begins with OpenCV-based text detection (Fig. 3), which identifies individual characters and outputs their bounding boxes (`x-min`, `x-max`, `y-min`, `y-max`). Characters with similar x- or y-values are grouped into complete text elements, allowing the system to reconstruct axis labels, ticks, and legend entries.

For non-text elements, OpenCV's edge detection and colour segmentation identify visual regions such as bars, pie slices, or legend swatches based on shape boundaries and colour differences. In charts with continuous marks like lines or areas, the system uses identified axis ticks as reference points and interpolates pixel coordinates along the visual path to map data values.

This yields a complete set of region coordinates but without semantic meaning. To bridge this gap, the system overlays numbered

labels on an annotated chart image and sends coordinate data to Gemini-2-Flash, which returns natural language descriptions for each region (e.g., *"Gold medal for the USA"*). This transforms low-level visual segments into semantically meaningful components, directly supporting **G1** by identifying functional chart regions and **G2** by linking visual geometry to interpretable data values.

4.6 Visual Attention Guidance

After chart regions are semantically identified, ViStruct generates step-by-step visual guidance to help users focus on relevant elements and understand how to extract the correct information from the chart.

To generate this guidance, the system provides the LLM with structured input: (1) the current low-level subtask, (2) a JSON object detailing chart regions, their coordinates, region IDS, and semantic labels, and (3) task-specific metadata from stage 4.2. Each of the subtask types uses an individual prompt to ensure contextual accuracy and clarity.

Based on this input, the LLM generates a sequence of visual guidance steps tied to specific regions or reference points. Some, like bar segments or axis labels, are directly retrieved from the JSON, while others involve simple geometric reasoning. For instance, the system draws a horizontal line from the bar's edge to the y-axis to read a bar's top value, guiding the user's attention.

ViStruct creates an interpretable and grounded guidance flow by combining textual instructions with precise visual markers. This directly supports **G2** by helping users connect spatial features to quantitative meaning and **G5** by making each reasoning step transparent and visually traceable.

5 Evaluating the ViStruct Pipeline

We conducted two evaluations to assess ViStruct's effectiveness as an expert-like visual reasoning pipeline: a performance evaluation to test decomposition accuracy and scalability and an expert evaluation to understand user perceptions regarding clarity, usefulness, and expert-like behaviour.

5.1 Performance Evaluation

We tested ViStruct on 45 visualization tasks drawn from VLAT and Mini-VLAT, covering all 12 chart types (**G4**). Each task was executed five times, resulting in 225 trials. For each trial, we evaluated ViStruct's ability to identify semantic chart regions (**G1**), map visual features to data values (**G2**), and generate coherent task decompositions (**G3**). ViStruct produced correct outputs in 192 of the 225 trials (85.33%). It performed consistently on concrete tasks such as value lookup and filtering. In contrast, more abstract tasks such as correlation analysis in multidimensional charts (i.e., bubble charts) occasionally showed inconsistencies by selecting the wrong visual channels for analysis.

5.2 Expert Review

We conducted an expert evaluation to assess whether ViStruct demonstrates expert-like behaviour and provides valuable guidance. Each participant was assigned three charts and could select any task from the question bank. They evaluated ViStruct's decomposition, workflow logic, region identification, and visual guidance. Specifically, they rated ViStruct on: *(1)* usefulness for novice users, *(2)* accuracy of subtasks and AOIs, and *(3)* alignment with their reasoning. Participants also gave open-ended feedback and used a think-aloud protocol during tasks.

We recruited 20 participants (M=12, F=8), including 12 undergraduates who completed a data visualization course, 4 graduate researchers, and 4 industry analysts. All were screened for chart familiarity and reported an average expertise rating of *6.35/7*. Each session lasted 30 minutes, and participants received a $10 gift card.

Experts' Perceptions and Feedback. Participants rated ViStruct highly for guiding visual reasoning (*M = 6.14, SD = 1.38*); the accuracy of its decompositions and AOIs scored 5.93 (*SD = 1.58*), and its perceived expert-likeness was 5.97 (*SD = 1.53*).

Step-by-step guidance supported participants in organizing their reasoning. Several users noted that the combination of visual overlays and sequential explanations clarified how to approach complex charts. One participant remarked that "*visual overlays alongside explanatory text are a useful step-by-step guide... especially with bubble charts where there are more variables than people are used to*" (*p14*). This form of guided interaction helped participants build a mental model of the task structure (**G2, G5**).

Region annotations anchored users' attention to relevant chart elements. Participants appreciated the system's ability to highlight and label specific visual components (**G1**). Numbered regions were especially helpful for less experienced users, who found the annotations reduced confusion (**G5**). One participant commented that "*the mapping feature... helps visually guide them in each step by highlighting the regions to focus on*" (*p9*).

Experts interpreted the decomposition as instructional guidance aimed at novice users. While many agreed that the step-by-step breakdown resembled how they would teach a novice (**G3**), they did not feel the need to follow every step themselves. One participant observed that "*the steps were clear and resemble how a domain expert would interpret and guide someone*" (*p3*). In contrast, some others noted that "*for experienced people, some of the steps might be a bit too detailed, but someone who is new to these charts could find it very helpful*" (*p8*).

6 Limitations and Future Work

Varying Effectiveness of Guidance Across Task Types: Our evaluation showed that the AOI-based guidance works well for concrete tasks such as filtering or locating values, where visual cues are direct. However, AOIs alone are less effective for more abstract tasks (i.e., correlation). These tasks require integrating multiple elements, suggesting that richer cues, such as tooltips or side panels, may be more appropriate. Future work should explore customized guidance strategies through participatory design or user feedback.

Supporting Diverse Reasoning Paths and Scaffolding: While the ViStruct decomposition strategy is effective for many tasks, it may not always match how users reason about complex visualizations. Abstract tasks often permit multiple valid approaches, so a fixed breakdown can be limiting. Future work should enable flexible, user-driven reasoning by adding an interactive chatbot that refines task flows, clarifies goals, and adapts strategies to each chart. In addition, the system could benefit from providing contextual explanations that clarify the rationale behind each step, helping users understand not just what to do, but why certain visual reasoning strategies are effective.

Human-in-the-loop Opportunities and Alignment with Novice Intentions: A significant portion of ViStruct's failures in visual attention guidance stemmed from errors in chart region identification, particularly due to limitations in OpenCV-based detection (e.g., missed or fragmented visual elements). These failures impact the quality of AOI mapping and, by extension, the effectiveness of task decomposition. Rather than relying solely on automation, we see an opportunity to introduce a human-in-the-loop setting where interactive correction of detected regions could reduce such errors while simultaneously supporting learning. This form of productive friction [13] may help novices develop a deeper understanding of visualization literacy by engaging directly with the structural interpretation of charts.

7 Conclusion

In conclusion, we present **ViStruct**, an automated pipeline that simplifies visualization tasks by decomposing them into semantically meaningful subtasks, extracting structured region information, and providing step-by-step visual guidance. The prototyped system is deployed and can be accessed here. Our evaluation shows reliable performance across diverse static charts and tasks. We plan to enhance flexibility with context-aware guidance and extend the approach to interactive settings to better support insight extraction.

Acknowledgments

This work was conducted in accordance with Research Ethics Board (REB) Protocol #48124. We sincerely thank Runlong Ye and Matthew Verona for their thoughtful input and valuable feedback throughout the development of this work.

References

[1] G. Alicioglu and B. Sun. A survey of visual analytics for explainable artificial intelligence methods. *Computers & Graphics*, 102, 2022. doi: 10.1016/j.cag.2021.09.002 2

[2] A. Alsaiari, J. Aurisano, and A. E. Johnson. Evaluating Strategies of Exploratory Visual Data Analysis in Multi Device Environments. In A. Kerren, C. Garth, and G. E. Marai, eds., *EuroVis 2020 - Short Papers*. The Eurographics Association, 2020. doi: 10.2312/evs.20201054 2

[3] A. Alves, J. Moura Pires, M. Y. Santos, A. Almeida, and A. León. Ai-assisted analytics–an automated approach to data visualization. In *International Conference on Conceptual Modeling*, pp. 343–358. Springer, 2024. doi: doi.org/10.1007/978-3-031-75599-6_24 2

[4] R. Amar, J. Eagan, and J. Stasko. Low-level components of analytic activity in information visualization. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, INFOVIS '05, p. 15, 2005. doi: 10.1109/INFOVIS.2005.24 1, 2, 3

[5] E. Briakou, J. Luo, C. Cherry, and M. Freitag. Translating step-by-step: Decomposing the translation process for improved translation quality of long-form texts. In B. Haddow, T. Kocmi, P. Koehn, and C. Monz, eds., *Proceedings of the Ninth Conference on Machine Translation*, pp. 1301–1317, 2024. doi: 10.18653/v1/2024.wmt-1.123 1

[6] T. T. Brunyé, T. Drew, K. F. Kerr, H. Shucard, D. L. Weaver, and J. G. Elmore. Eye tracking reveals expertise-related differences in the time-course of medical image inspection and diagnosis. 7. doi: 10.1117/1.JMI.7.5.051203 2

[7] K. Börner, A. Bueckle, and M. Ginda. Data visualization literacy: Definitions, conceptual frameworks, exercises, and assessments. *Proceedings of the National Academy of Sciences*, 116(6):1857–1864, 2019. doi: 10.1073/pnas.1807180116 2

[8] Y. Guo, D. Shi, M. Guo, Y. Wu, N. Cao, and Q. Chen. Talk2data: A natural language interface for exploratory visual analysis via question decomposition. *ACM Trans. Interact. Intell. Syst.*, 14(2), article no. 8, 24 pages, Apr. 2024. doi: 10.1145/3643894 2

[9] Y. Hou, B. Gildereli, Y. Tu, and M. Sachan. Do vision-language models really understand visual language? *arXiv preprint arXiv:2410.00193*, 2024. doi: 10.48550/arXiv.2410.00193 2

[10] Z. Huang, Z. Zhang, Z. Zha, Y. Lu, and B. Guo. Relationvlm: Making large vision-language models understand visual relations. *CoRR*, abs/2403.12801, 2024. doi: 10.48550/ARXIV.2403.12801 2

[11] E. Jamet, M. Gavota, and C. Quaireau. Attention guiding in multimedia learning. 18(2):135–145. Place: Netherlands Publisher: Elsevier Science. doi: 10.1016/j.learninstruc.2007.01.011 2

[12] B. Karer, I. Scheler, H. Hagen, and H. Leitte. Conceptgraph: A formal model for interpretation and reasoning during visual analysis. *Computer Graphics Forum*, 2020. doi: 10.1111/cgf.13899 1

[13] M. Kazemitabaar, O. Huang, S. Suh, A. Z. Henley, and T. Grossman. Exploring the design space of cognitive engagement techniques with ai-generated code for enhanced learning. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*, IUI '25, 20 pages, p. 695–714, 2025. doi: 10.1145/3708359.3712104 1, 4

[14] M. Kazemitabaar, J. Williams, I. Drosos, T. Grossman, A. Z. Henley, C. Negreanu, and A. Sarkar. Improving steering and verification in ai-assisted data analysis with interactive task decomposition. In *Proceedings of the 37th ACM Symposium on User Interface Software and Technology*, 2024. doi: 10.1145/3654777.3676345 1, 2

[15] D. A. Keim. Information visualization and visual data mining. *IEEE transactions on Visualization and Computer Graphics*, 8, 2002. doi: 10.1109/2945.981847 1

[16] X. Li, Y. Sun, W. Cheng, Y. Zhu, and H. Chen. Chain-of-region: Visual language models need details for diagram analysis. In *The Thirteenth International Conference on Learning Representations*. 2

[17] E. Moerth, Z. Kostic, N. Gehlenborg, H. Pfister, J. Beyer, and C. Nobre. Beyond time and accuracy: Strategies in visual problem-solving. *CHI '25: Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, May 2025. doi: 10.1145/3706598.3714024 2

[18] B. B. Morrison, L. E. Margulieux, and M. Guzdial. Subgoals, context, and worked examples in learning computing problem solving. In *the 11th ACM International Computing Education Research Conference (ICER '15)*, 2015. doi: 10.1145/2787622.2787733 1

[19] S. Mukhopadhyay, A. Qidwai, A. Garimella, P. Ramu, V. Gupta, and D. Roth. Unraveling the truth: Do vlms really understand charts? a deep dive into consistency and robustness. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 16696–16717, 2024. doi: 10.48550/arXiv.2407.11229 2

[20] A. A. Nelms and M. Segura-Totten. Expert–novice comparison reveals pedagogical implications for students' analysis of primary literature. *CBE–Life Sciences Education*. doi: 10.1187/cbe.18-05-0077 1

[21] C. Nobre, K. Zhu, E. Mörth, H. Pfister, and J. Beyer. Reading between the pixels: Investigating the barriers to visualization literacy. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, 2024. doi: 10.1145/3613904.3642760 2

[22] M. Rezaie, M. Tory, and S. Carpendale. Struggles and Strategies in Understanding Information Visualizations . *IEEE Transactions on Visualization & Computer Graphics*, 30(06):3035–3048, June 2024. doi: 10.1109/TVCG.2024.3388560 2

[23] A. Schlieder, J. Rummel, P. Albers, and F. Sadlo. Sequential visual cues from gaze patterns: Reasoning assistance for bar charts. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, 2025. doi: 10.1145/3706598.3713352 2

[24] S. Suh, B. Min, S. Palani, and H. Xia. Sensecape: Enabling multilevel exploration and sensemaking with large language models. In *Proceedings of the 36th annual ACM symposium on user interface software and technology*, pp. 1–18, 2023. doi: 10.1145/3586183.3606756 1

[25] L. Tankelevitch, V. Kewenig, A. Simkute, A. E. Scott, A. Sarkar, A. Sellen, and S. Rintel. The metacognitive demands and opportunities of generative ai. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, article no. 680, 24 pages, 2024. doi: 10.1145/3613904.3642902 1

[26] T. W. Teo, Z. Q. Peh, T. W. Teo, and Z. Q. Peh. An exploratory study on eye-gaze patterns of experts and novices of science inference graph items. 3(3):205–229. doi: 10.3934/steme.2023013 1

[27] J. G. Tullis, R. L. Goldstone, and A. J. Hanson. Scheduling scaffolding: The extent and arrangement of assistance during training impacts test performance. *Journal of Motor Behavior*, 47(5):442–452, 2015. 1

[28] W. Wang, Y. Rao, R. Zhi, S. Marwan, G. Gao, and T. W. Price. Step tutor: Supporting students through step-by-step example-based feedback. In *Proceedings of the 2020 ACM Conference on Innovation and Technology in Computer Science Education*, ITiCSE '20, 2020. doi: 10.1145/3341525.3387411 1

[29] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022. doi: 10.5555/3600270.3602070 2

[30] H. Xie, F. Wang, Y. Hao, J. Chen, J. An, Y. Wang, and H. Liu. The more total cognitive load is reduced by cues, the better retention and transfer of multimedia learning: A meta-analysis and two meta-regression analyses. 12(8):e0183884. doi: 10.1371/journal.pone.0183884 2

[31] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations*, 2023. doi: 10.48550/arXiv.2210.03629 2

[32] J. Ye, A. Dash, W. Yin, and G. Wang. Beyond end-to-end VLMs: Leveraging intermediate text representations for superior flowchart understanding. In *Proceedings of the Human Language Technologies (Volume 1)*, Apr. 2025. doi: 10.18653/v1/2025.naacl-long.180 2

[33] Y. Zhang, W. Zhang, Z. Zeng, K. Jiang, J. Li, W. Min, W. Luo, Q. Guan, J. Lin, and W. Yu. Mapreader: a framework for learning a visual language model for map analysis. *International Journal of Geographical Information Science*, 2025. doi: 10.1145/3557919.3565812 2

[34] Y. Zhao, J. Wang, L. Xiang, X. Zhang, Z. Guo, C. Turkay, Y. Zhang, and S. Chen. Lightva: Lightweight visual analytics with llm agent-based task planning and execution. *IEEE Transactions on Visualization and Computer Graphics*, 2024. doi: 10.1109/TVCG.2024.3496112 2