Journal for Geometry and Graphics
Volume 28 (2024), No. 2, 213–227

Analysis of AI Prompts for Introductory Education in Generative Art

Kunio Kondo[1,2], Hiroyuki Nagayoshi[2], Makoto Nagata[2], Junichi Kanebako[2]

[1] *Tokyo University of Technology, Hachioji, Japan*
kondo@stf.teu.ac.jp

[2] *Kobe Design Unviersity, Kobe, Japan*
nagayoshi-h@kobe-du.ac.jp, nagata-m@kobe-du.ac.jp, kanebako-j@kobe-du.ac.jp

Abstract. Generative Art is creating CG images by designing and implementing image generation algorithms as programs. This method requires a solid understanding of algorithms and programming, which can be challenging for students and designers who do not have these skills. Therefore, our research proposes guidelines for writing prompts that support precise descriptions of image generation algorithms. The research involves three exercises: 1. Describing algorithms in natural language and generating and evaluating the resulting images, 2. Analyzing artworks created through Generative Art methods to derive algorithm descriptions, 3. Generating CG images using prompts. These exercises allow students to describe procedural steps in natural language, enabling AI to generate p5.js programs without programming expertise. Through analysis of the images produced and student feedback, the study identified key principles for designing effective prompts, essential considerations for program development, and guidelines for accurate CG image generation. While AI's interpretive capabilities improve, students' logical thinking and precise prompt-writing skills remain crucial for effective human-AI collaboration.

Key Words: generative art, introductory education, ChatGPT3.5, AI prompts, algorithms

MSC 2020: 68T01 (primary), 97G99

1 Introduction

Generative Art is a method of creating CG images by devising image generation procedures and implementing them as programs. This process requires the proposal of algorithms and the knowledge to implement them in programs. Therefore, for students and designers who lack related programming knowledge, creating CG images using Generative Art techniques is challenging.

This research proposes guidelines for writing prompts that enable more accurate descriptions of image generation algorithms based on the analysis of prompts and images created through three exercises. This study aims to make the creation of CG images in Generative Art more accessible using ChatGPT 3.5 and to clarify the effectiveness and challenges of this approach. The specific image generation procedure is as follows: As shown in Figure 1, the drawing procedure for CG drawing is input in natural language. Then, ChatGPT generates a p5.js program from the prompt. The program displays the image and a user evaluates whether it matches the intended image.

To perform this procedure better, we propose the following exercises for describing Generative Art algorithms in natural language: 1. Describe algorithms in natural language and draw and evaluate them based on human understanding of the algorithms. 2. Describing algorithms based on analyzing works using Generative Art techniques, and 3. Generating images using prompts. These exercises allow individuals to create CG images using generative AI without specific knowledge of programming languages, as they describe the procedural steps in natural language.

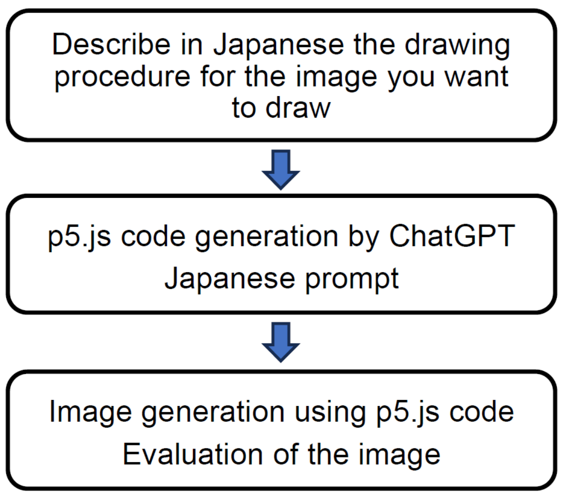

Figure 1: Drawing procedure using ChatGPT 3.5 [6].

These exercise has already been conducted at other universities, demonstrating its adaptability across different educational settings. Although its content is highly general, it can be easily adjusted to suit the specific objectives and needs of various teaching environments.

In this paper, Section 2 explains previous studies on education related to CG image creation. Section 3 proposes exercises for algorithm generation using natural language descriptions, defining their objectives, methods, and benefits. Section 4 describes the results of students' works using the three proposed methods, and Section 5 evaluates and discusses the contents of these exercises based on these examples.

2 Related Work

In this section, we discuss the current status and challenges of education in CG image creation, the acquisition of algorithms for image generation through Generative Art and mathematical modeling, communication between humans and generative AI in natural language descriptions of algorithms, and the role of generative AI in art and design education.

2.1 Algorithm Education in Generative Art

Teaching materials such as 'Shapes Created by Formulas' by Fuchigami [4] and Kondo et al. [7] propose acquiring mathematical knowledge and using formulas for various image expressions. However, to perform such image expressions, knowledge of formulas and programming is necessary, along with proposing algorithms to realize rough sketches of the desired images and programming based on them. Additionally, Oizumi [9] summarizes methods for reproducing CG works by the Computer Technique Group (CTG). They were investigating how past works were created leads to understanding their drawing algorithms. Analyzing past CG images created using Generative Art techniques, describing them in natural language, and creating CG images using programs generated by ChatGPT are the primary focus.

2.2 Information Transmission in Natural Language

This study proposes a procedure for describing algorithms in natural language and creating programs using generative AI like ChatGPT. Communication between humans and generative AI share common challenges with human-to-human communication. Clark [1] points out that people consider each other's background knowledge while conversing. While communicating, they consider public and private backgrounds such as society, culture, and the other person's preferences. Standard information and a shared foundation—mutual knowledge, common beliefs, and assumptions—are essential to facilitate smooth communication. We examine methods for conveying the necessary information for creating images through human communication based on educational outcomes.

2.3 Generative AI and Art, Art & Design Education

In this section, we discuss research related to generative AI and art [2, 3, 5, 8, 10–12], divided into three categories: (1) Generative AI and Art, and (2) Generative AI and CG/Art Education.

(1) Generative AI and Art

Epstein [2] analyzes the broad impact of generative AI on art, science, and the creative industries, considering its effects on society. Wyse [12] proposes mechanisms for generating new ideas and art pieces using deep learning neural networks, focusing on methodologies that promote art creation using generative AI and foster creative thinking. Additionally, it clarifies how deep learning technology can be applied and understood in the context of art and creative projects.

(2) Generative AI and CG/Art Education

Fathoni [3] discusses the methods and benefits of using generative AI in art and design education. McCormack [8] examines the applicability and effectiveness of complexity measures applied to multiple datasets and investigates how these measures relate to individual art pieces. Oppenlaender [10, 11] focuses on the techniques and applications of prompt engineering, discussing the classification of prompt modifiers and providing specific examples. This research considers how prompt engineering influences the creative process in AI art. Garvey [5] introduces a curriculum and practical applications of generative AI and art for university students, describing a useful educational program where students use AI tools to create art.

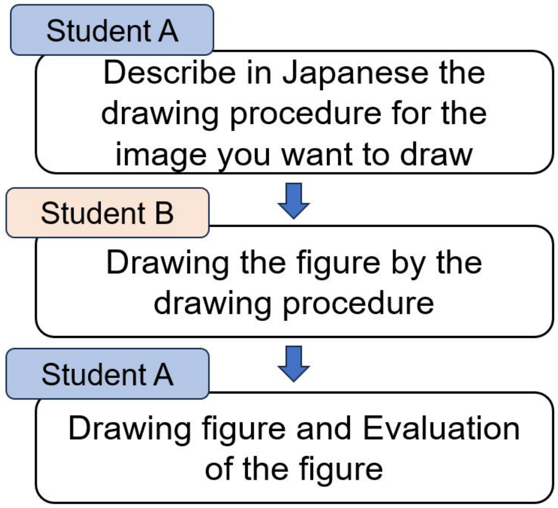

Figure 2: Drawing procedure by student pairs.

These related studies do not address the approach proposed in this paper, which involves "users summarizing logical drawing procedures into prompts, generative AI creating programs, and using these to produce CG images and the corresponding educational methods."

3 Exercises for Generating Algorithms in Natural Language

In this section, we discuss exercises related to generating Generative Art algorithms in natural language. We summarize the objectives, methods, and benefits of the three exercises proposed in this study. Furthermore, we explain the characteristics and necessity of each exercise.

3.1 Exercise 1: Describing Algorithms in Japanese

Objective: The goal is to describe the drawing procedure of original shapes conceived by each student and to draw according to the described procedures, thereby enhancing the ability to verbalize and understand shape drawing algorithms.

Method: As shown in Figure 2, two students pair up. Student A writes down the drawing procedure of a shape they have in mind in Japanese and gives the written instructions to the other student B. Student B reads the Japanese instructions, understands the drawing procedure, and follows it to draw the shape. Then, Student A evaluates whether the drawn shape matches the intended shape and assesses the written drawing procedure. Through this exercise, students also learn that drawing instructions can be interpreted differently, highlighting the importance of precise descriptions.

Benefits:
1. Even before taking computer programming courses, students can logically describe drawing procedures.
2. Students can learn to accurately describe drawing algorithms.
3. Students can learn to verbalize drawing procedures by reading and drawing instructions written by others (Figures 3 and 4).

3.2 Exercise 2: Analyzing Works Using Generative Art Methods

Objective: The goal is to understand drawing algorithms through the analysis of works.

1. Decide a point.
2. Draw a line segment of 0.5 cm upward.
3. Draw a line segment 1.5 times the length in the 90° direction.
4. Repeat this 10 times.

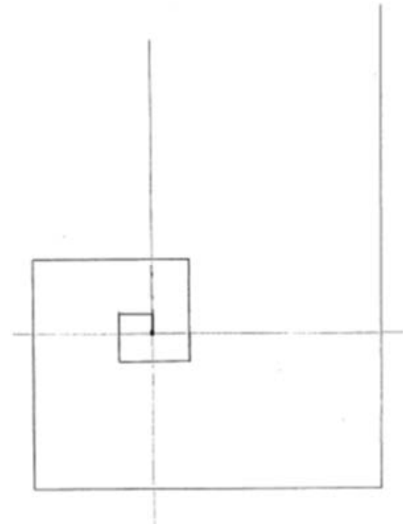

Figure 3: Text of drawing procedures and the result of drawing by student pairs.

1. Draw a square.
2. Draw a square with its diagonal as one side.
3. Repeat 5 times.

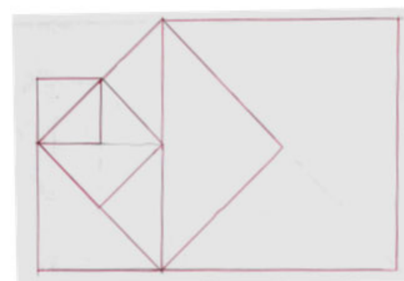

Figure 4: Text of drawing procedures and the result of drawing by student pairs.

Method: Using plane geometry, shape transformations, repetitions, and branches, summarize the drawing procedures of past works in natural language. Then, based on the described natural language, compile the algorithm descriptions. Input this text as a prompt into ChatGPT, generate a p5.js program, and display the image. Figure 5 shows an example of the images used for the analysis.

Benefits:
1. Through this exercise, students can learn about the algorithms used to create past works.
2. Students can clarify the drawing procedures and acquire a logical perspective by analyzing these works.

3.3 Exercise 3: Creating CG Images with ChatGPT

Objective: The goal is to visualize a shape and summarize the prompt needed to generate that shape. This method involves entering a Japanese-written algorithm into ChatGPT to create a p5.js program and drawing the shape instead of having a student draw the shape as in the previous section.

Method: First, compile the entire algorithm for the desired shape and input it into ChatGPT to generate a p5.js program and create the drawing. If the resulting drawing is not as intended, modify parts of the algorithm description or the program itself. Figure 6 shows the drawing procedure and examples from past students. Figure 7 shows the prompt compiled to generate the image in Figure 6.

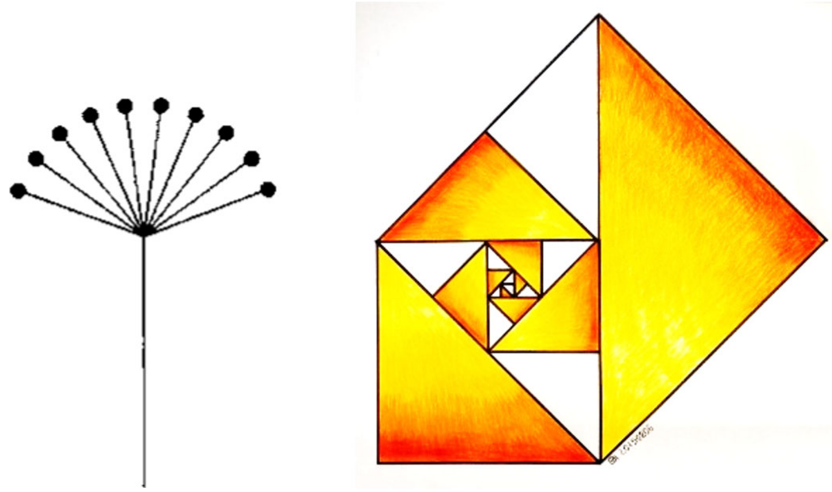

Figure 5: Example images for algorithm analysis.

Benefits:

1. Even before formal education in computer programming, students can logically describe drawing procedures through this exercise.
2. Students can learn how to write and use prompts for generative AI.

> 1. Draw a sector with radius r and an angle of 30°.
> 2. Rotate the sector from Step 1 by 15° and increase the radius by a factor of 1.1 (keeping the center at the vertex of the sector).
> 3. Repeat Step 2 twelve times.

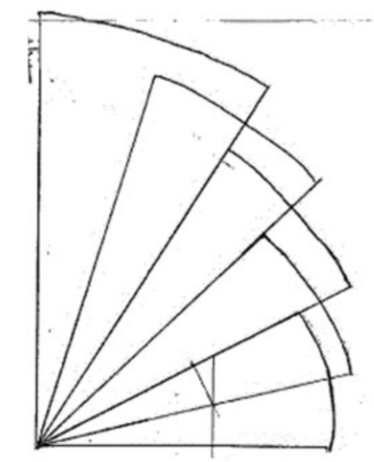

Figure 6: Drawing procedures, drawings by past students.

> Below is the description for creating a p5.js program to draw shapes. Based on this description, please create a recursive program.
> 1. Draw a sector with radius r and an angle of 30 degrees. Set r to 100.
> 2. Rotate the sector from Step 1 by 15 degrees and increase the radius by a factor of 1.1 (keeping the center of rotation at the vertex of the sector).
> 3. Repeat Step 2 twelve times.
> 4. On the 13th iteration, return to Step 1 and repeat Steps 1 to 3 once more.

Figure 7: Prompt based on the procedure in Figure 6.

Below is the description for creating a p5.js program to draw shapes. Based on this description, please create a recursive program.

1. Represent a line segment with a starting point p_1 and an ending point p_2.
2. Draw the line segment connecting p_1 and p_2, increase its length by a factor of 1.5, and set p_3 to the position obtained by rotating p_2 counterclockwise by 90 degrees around p_2.
3. Update p_1 to p_2 and update p_2 to p_3.
4. Use initial values $p_1 = (180, 220)$ and $p_2 = (180, 205)$, and repeat Steps 2 and 3 ten times.

Figure 8: Prompt based on the procedure in Figure 3.

4 Results of Image Creation in Natural Language

In Section 4, we present the results of image creation by students. We explain the results of each of the three exercises described in Section 3.

4.1 Example Results of Exercise 1: Writing and Drawing by Students

This section shows the results of two students, A and B, working in pairs. Each student wrote a drawing algorithm in bullet points, exchanged lists, read the instructions, and performed the drawing. They were instructed to refrain from asking each other questions if they encountered unclear points during the drawing process. Figures 9 and 10 show examples of the bullet-point instructions handwritten by student A and the shapes drawn by student B. From these exercise results, the following can be observed. When the bullet-point instructions are clear, the drawing student can easily visualize and draw the intended shape. However, when the instructions are unclear, the student cannot draw the intended shape. If humans cannot understand the instructions, neither can generative AI. Therefore, training to write more accurate and logical instructions and understanding algorithms is crucial.

4.2 Example Results of Exercise 2: Analysis of Trees and Geometric Shapes

In this section, we explain the exercise results of image generation based on the analysis of past CG images.

4.2.1 Analysis Results of Tree Drawing Procedures

Our examination of a tree from past works led us to summarize the drawing procedures into a prompt. The image we generated was a direct output of the p5.js program, a key tool in our process. Figure 11 presents the prompt and the resulting image.

4.2.2 Analysis Results of Rotational Shape Drawing Procedures

Figure 12 shows the reference image and the created image. The results of this exercise show that analyzing the given past shapes and considering how they were drawn leads to thinking

1. Draw a circle with a radius of 8 cm.
2. Divide the circle into 16 equal parts.
3. Number the points on the circle clockwise from 1 to 16.
4. Draw a circle with a radius of (0.5 × number) cm that is inscribed in the circle from Step 1 at each of the 16 points.

Figure 9(a): Drawing procedure by Student A.

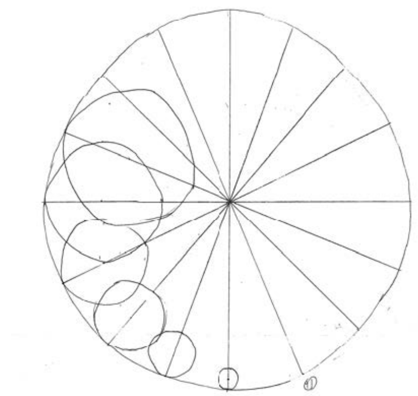

Figure 9(b): Result of Student B's hand-drawn drawing.

1. Draw a right-angled isosceles triangle.
2. Draw another right-angled isosceles triangle such that the longest side of the previous triangle becomes the shorter side of the new triangle. (Either of the shorter sides is fine, whichever looks better . . .)
3. Repeat Step 2. (Note: the meaning of this repetition)
4. Stop when it becomes too large.

Figure 10(a): Drawing procedure by Student A.

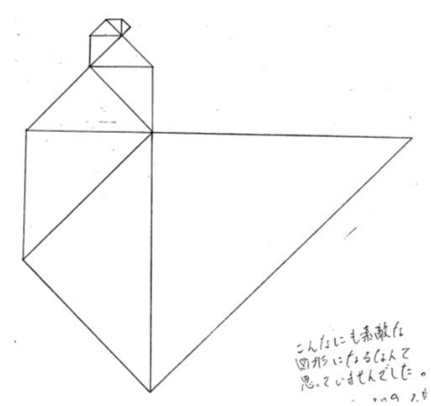

Figure 10(b): Result of Student B's hand-drawn drawing.

about drawing algorithms. This helps organize the use of geometric transformations and other techniques for drawing shapes.

Below is the text for creating a p5.js program to draw a shape. Based on this text, please create a program using recursion.

1. Draw a vertical line with a thickness of 1 in the center of the screen.
2. From the tip of that line, branch out into a fan shape with 7 lines at an angle of 20 degrees.
3. Repeat this process 3 times.

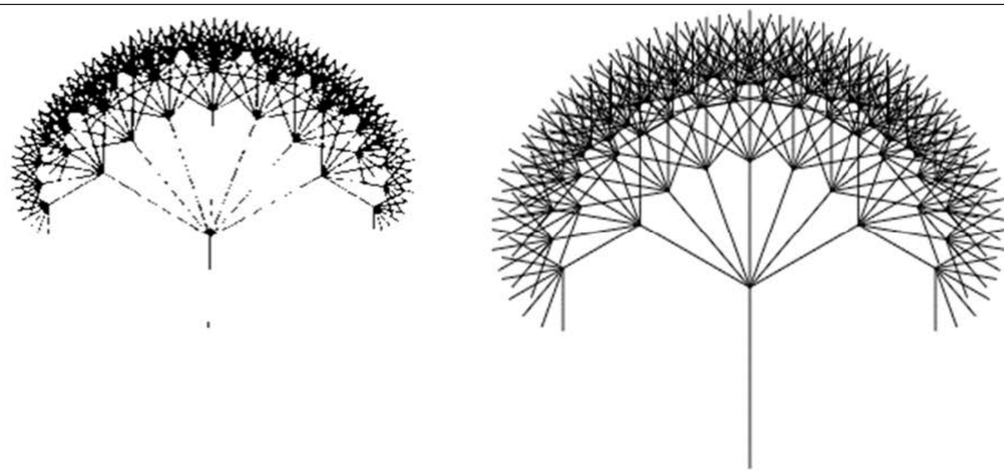

Figure 11: Prompt, analyzed image and drawing result by p5.js.

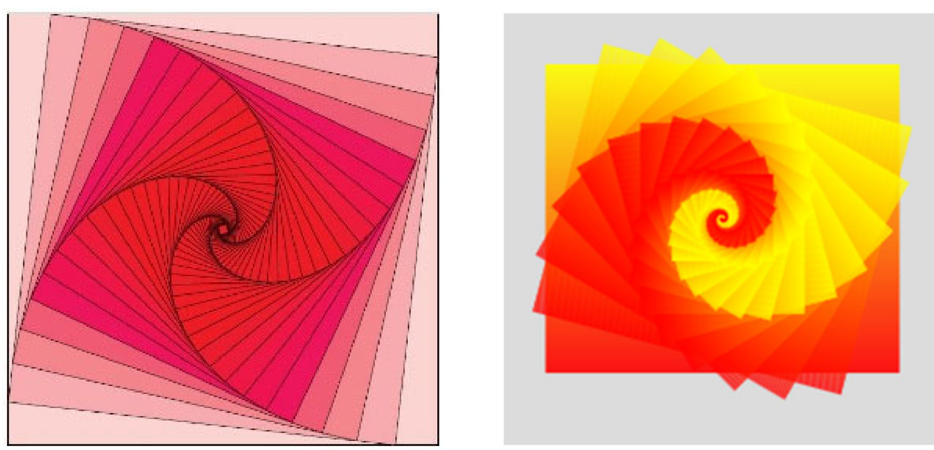

Figure 12: Reference image and drawing result by p5.js.

4.3 Example Results of Exercise 3: Creating CG Images with ChatGPT

Figure 13 and Figure 14 depict the outcomes of an image creation exercise, where students formulated prompts based on bulleted sentences and fed them into ChatGPT. These exercises highlight a key limitation of generative AI-while the students were successful in generating the desired drawing in a human drawing exercise, the AI often failed to produce the intended shape when the prompts were directly inputted. This discrepancy can be attributed to the difference in how human communication and generative AI process information. Human communication relies on tacit understanding, where the entire sentence is comprehended before the drawing process begins. In contrast, generative AI does not start with a holistic understanding of the sentence. As a result, if there is ambiguity in a single sentence, the AI may produce a figure that the creator did not intend.

Below is the text for creating a p5.js program to draw a shape.
1. Draw a large square.
2. Draw a circle that fits perfectly inside the square.
3. Draw the diagonals of the square by connecting each pair of opposite corners.
4. Using each vertex of the square as a center, draw a circle of the same size as the one inside the square. Ensure there is only one circle inside the square.

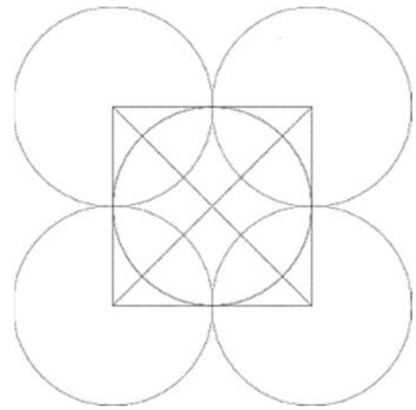

Figure 13: Prompt and drawing results with p5.js.

1. Draw an X using the diagonals.
2. From the intersection point of the diagonals, draw lines extending right 20 units, up 40 units, left 60 units, and down 80 units, increasing by 20 units each time.
3. Following this pattern, draw lines in the right, up, left, and down directions, each increasing by 20 units.

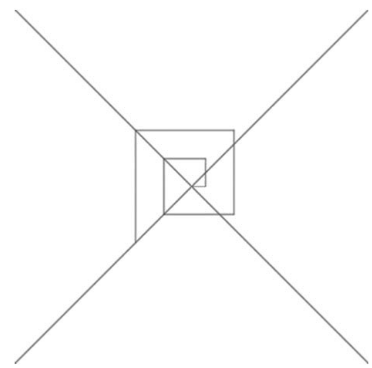

Figure 14: Prompt and drawing results with p5.js.

The CG image shown in Figure 15 is the final project of this course. After completing 14 lectures, this final project was created using ChatGPT. It can be said that this work demonstrates an improvement in logical thinking and programming skills, which were the objectives of this course.

This exercise encourages students to articulate their intentions in spoken language before writing the program, with ChatGPT assisting in translating these verbal descriptions into programming code. Furthermore, encouraging students to understand the code generated by AI can be an effective method for learning how to code, as it helps them grasp underlying programming concepts and logic.

5 Discussion

In this section, we analyze the CG images created through the three exercises described in Section 3, as well as the various comments received from the students who participated in these exercises. Based on this analysis, we propose key principles for creating effective prompts for CG algorithms, important considerations for developing programs using these prompts, and guidelines for writing prompts that facilitate accurate and efficient CG image generation.

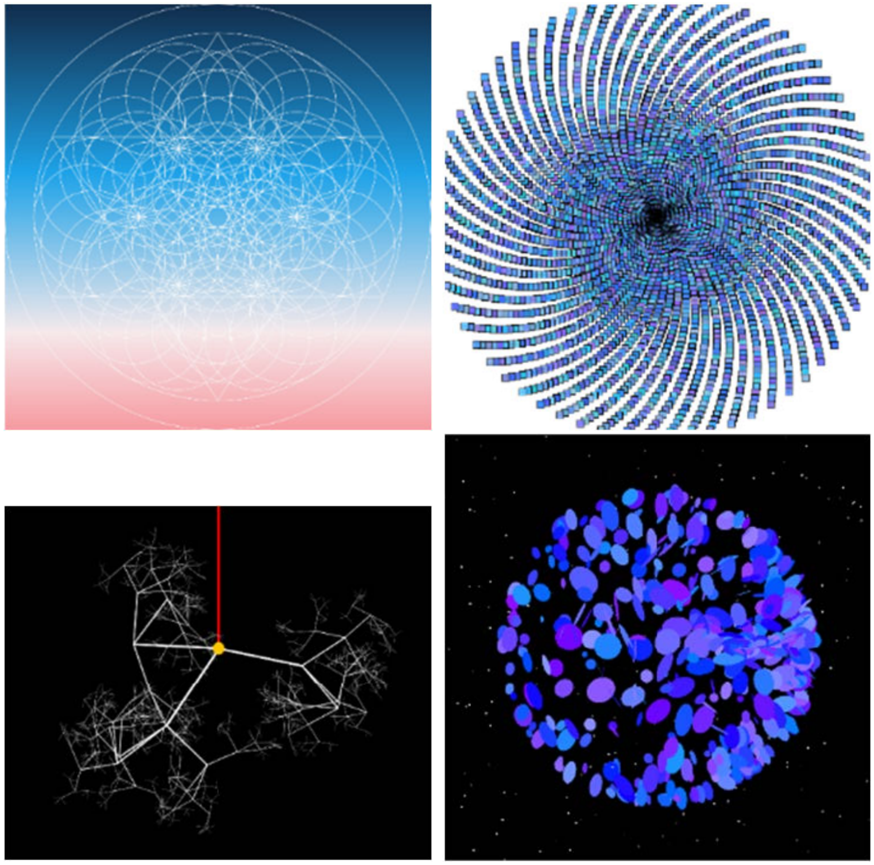

Figure 15: Students' work.

In this research, we use natural language prompts to describe algorithms and generate shapes. This approach allows students, even those without programming experience, to describe their intended shapes using natural language, enabling AI like ChatGPT to create the corresponding programs. While natural language can inherently contain ambiguities, this flexibility allows for new and creative expressions. To ensure that the generated images accurately reflect the students' intentions, it is essential to formulate precise prompts that effectively convey the desired drawing algorithms.

5.1 Note for Creating Programs using Prompts

1. About Text-Generation AI: Text-generation AI (language models) learn patterns from vast datasets to generate sentences. This data-driven approach can output sentences that reflect context to some extent. However, these outputs often remain superficial imitations, not based on the deep contextual understanding and experience humans have. The processing by generative AI is not necessarily sequential. It has a complex structure that processes inputs through multiple layers, but it does not achieve comprehensive understanding as humans do. On the other hand, human text understanding is a comprehensive approach that includes context. Humans can complement meaning even with incomplete information, drawing the intended shapes even with ambiguities.

2. Regarding Program Generation and Bugs: When a text-generation AI generates a program, it may contain bugs. Since generative AI generates programs from training data and has limitations in its ability to judge the accuracy of the generated program, it can be difficult to correct mistakes. Therefore, even if a person inputs the correct algorithm in text, the generative AI may output an incorrect program. This can be a significant issue, especially for beginners who lack the knowledge to detect program errors. However, in practice, students interact with ChatGPT multiple times to correct the program and achieve the intended image generation.

3. Regarding Inference and Completion: ChatGPT and similar generative AIs have the ability to infer and complete sentences from context. However, this is a probabilistic process, and the results are probabilistic and do not always achieve the intended outcome. When generative AI performs inference and completion, it is based on probabilistic reasoning derived from training data. This is generally referred to as "inference," but it is a mechanism different from human reasoning.

4. Complementary Functions Some generative AI models can search for external information and provide supplementary information. This can provide richer information. Understanding these four points is crucial for writing prompts for CG image creation.

5.2 Guidelines for Algorithmic Writing Prompts

When creating Generative Art algorithms in natural language, as proposed in Section 3, the following points should be noted:

1. Accurate Intent Description: This is the most crucial element in drawing. It is necessary to accurately convey what kind of CG image you want to generate and what processes you want to include in the p5.js code.

2. Logical Thinking and Justification: The generated program should be logical and consistent, especially in parts related to the code's structure.

3. Structured Writing: The requirements and instructions for the processing steps must be organized and structured in a way that clearly communicates the instructions for program generation.

4. Concise Writing: It is essential to avoid redundancy and convey the key points.

5. Providing Clear Context: It is important to provide specific requirements and conditions for CG image creation as context.

6. Appropriate Environment Setting Description: Providing information about the environment and requirements where the generated program will run allows for proper code design and execution.

7. Iteration and Verification: Verifying the program generated by ChatGPT and providing corrections or additional instructions as needed is crucial. Ensuring accurate understanding through repeated communication can be achieved.

5.3 Prompt Writing for Generative Art

There are significant differences between humans and generative AI in understanding coordinates and the placement of shapes for creating CG images. The following writing techniques are effective for ensuring that ChatGPT 3.5 accurately understands these concepts:

1. Specific and Clear Instructions: It is crucial to provide specific and clear instructions regarding coordinates and the placement of shapes. For example, explicitly specifying particular coordinates (e.g., x-coordinate 100, y-coordinate 200) and the size of shapes (e.g., 10 pixels tall, 20 pixels wide) as absolute positions can improve the accuracy of the generated image.

2. Use of Mathematical Expressions Utilizing mathematical expressions when describing coordinates and shape placement helps reduce misunderstandings. For instance, using mathematical expressions like "place a square at coordinates (100, 200)" makes it easier for AI to understand the intention accurately.

3. Detailed Shape Descriptions: It is essential to provide detailed descriptions of the shapes and forms used. For example, describing a circle with "place a circle at center coordinates $(50, 50)$ with a radius of 30 pixels" helps the AI accurately understand the shape to be drawn.

4. Provision of Contextual Information: Providing background information and requirements related to CG image creation is essential. For instance, specifying that "the top left of the screen is the origin, with the x-axis horizontal and the y-axis vertical" helps the AI accurately understand the coordinate system for drawing on a computer.

5.4 Contribution of our proposed Prompt Writing

In order to use generative AI prompts, we believe that clear description and expression of usage definitions in Japanese will become as important as programming education. Therefore, we believe that opportunities to relearn one's native language will become even more important.

1. Importance of Clear Descriptions: To effectively use generative AI, clear and detailed prompts are necessary. Ambiguous instructions cannot elicit accurate responses from AI, so users need the skill to convey what they want specifically.

2. Opportunities for Relearning Japanese: In prompt creation, accurate expression and vocabulary are important. This promotes relearning and deepening one's native language, which in turn is expected to improve overall language proficiency.

3. Similarities and Importance of Programming Education: Programming also requires the ability to give clear instructions and construct logic, and the same skills are needed for prompt creation. Therefore, just like programming education, it is valuable to include prompt creation skills in education.

4. Importance of our Proposal: This research is significant as it reaffirms the importance of language education amidst the growing use of generative AI. Improving language proficiency also contributes to the enhancement of other academic fields and professional skills.

For these reasons, the idea that education and research on creating generative AI prompts are important is valid and should be increasingly emphasized in the future.

These insights from the project are particularly valuable for nurturing creative artists. Our educational approach emphasizes that by following the prompt guidelines, students can develop the ability to clearly express their intentions through the artworks they create. This approach not only enables them to effectively communicate their ideas but also encourages the integration of emotional and cultural elements into their creations. As a result, students can gain a deeper understanding of how to convey meaning through visual art, which is essential for their growth as creative artists.

6 Conclusions

This research aims to propose guidelines for writing prompts that support more precise descriptions of image generation algorithms, derived from the analysis of prompts and images created through three exercises. Three exercise methods were proposed to achieve this aim: describing algorithms and drawing in natural language, analyzing artworks generated using Generative Art methods, and creating CG images using ChatGPT. Through this research, we sought to improve the ability to write effective prompts for introductory Generative Art education while evaluating the effectiveness and challenges of this approach. The images produced through these exercises, along with student feedback, were analyzed. Based on this analysis, the study identified fundamental principles for creating effective prompts for CG algorithms, essential considerations for developing programs using these prompts, and guidelines for writing prompts that facilitate accurate and efficient CG image generation. While AI's ability to interpret prompts may continue to improve, students' logical thinking skills and their capacity to craft precise prompts remain essential for successful collaboration between humans and AI. This research is part of an effort to continue refining the methods that enable students to effectively communicate their creative intentions through natural language, thus further bridging the gap between human creativity and AI capabilities. Future work will verify the effectiveness of these guidelines for natural language algorithm descriptions in creating Generative Art, using the findings of this analysis as a foundation.

Acknowledgements

We would like to express my deep gratitude to Professor Kimiya Fujisawa and Professor Mika Enomoto of Tokyo University of Technology for their valuable comments on human communication with artificial intelligence and program creation using ChatGPT. Many of the CG images included in this paper were created by first-year students at Kobe Design University. The examples of drawing by students were provided by students from Aichi University of the Arts and Kyushu University School of Design. We deeply appreciate their contributions. The copyright for these works remains with the students, and we have obtained their permission to include them in this paper.

References

[1] H. H. Clark and S. E. Brennan: *Grounding in communication.* In L. B. Resnick, J. M. Levine, and S. D. Teasley, eds., *Perspectives on socially shared cognition*, 127–149. APA Books, 1991. ISBN 1557981213. doi: 10.1037/10096-006.

[2] Z. Epstein, A. Hertzmann, L. Herman, R. Mahari, M. R. Frank, M. Groh, H. Schroeder, A. Smith, M. Akten, J. Fjeld, H. Farid, N. Leach, A. Pentland, and O. Russakovsky: *Art and the science of generative AI: A deeper dive*, 2023. arXiv: 2306.04141.

[3] A. F. C. A. Fathoni: *Leveraging Generative AI Solutions in Art and Design Education: Bridging Sustainable Creativity and Fostering Academic Integrity for Innovative Society.* In *E3S Web Conf. The 5th International Conference of Biospheric Harmony Advanced Research (ICOBAR 2023)*, vol. 426, 01102. 2023. doi: 10.1051/e3sconf/202342601102.

[4] K. Fuchigami: *CG Images Created by Mathematical Formulas.* Tokyo University of Technology, School of Media Science, Exercise Material, 2005. https://kondolab.org/education/course/math2014/.

[5] G. P. Garvey: *A University Curriculum Course for Undergraduates: Artificial Intelligence and Art.* In *SIGGRAPH Asia 2023 Educator's Forum*, SA '23, 1–2. ACM, 2023. doi: 10.1145/3610540.3627006.

[6] K. Kondo, H. Nagayoshi, M. Nagata, and J. Kanebako: *Introduction to CG Education Using ChatGPT in Department of Image Arts.* In *The 21st International Conference on Geometry and Graphics*, poster043. 2024.

[7] K. Kondo and H. Sato: *CG Images Created by Mathematical Formulas for Introductory CG Programming Education.* In *Proceedings of the 2008 Annual Conference of the Japan Society for Graphic Science*. 2008.

[8] J. McCormack and C. Cruz Gambardella: *Complexity and aesthetics in generative and evolutionary art.* Genetic Programming and Evolvable Machines **23**(4), 535–556, 2022. doi: 10.1007/s10710-022-09429-9.

[9] K. Oizumi: *The Creation of Computer Art: The Track and Thought of the Computer Technique Group (CTG) 1966–1969.* NTT Publishing, 2015.

[10] J. Oppenlaender: *Prompt Engineering for Text-Based Generative Art*, 2022. arXiv: 2204.13988v1.

[11] J. Oppenlaender, R. Linder, and J. Silvennoinen: *Prompting AI Art: An Investigation into the Creative Skill of Prompt Engineering*, 2023. arXiv: 2303.13534.

[12] L. Wyse: *Mechanisms of Artistic Creativity in Deep Learning Neural Networks*, 2019. arXiv: 1907.00321.

Received August 28, 2024; final form October 28, 2024.