GENERATIVE VISUALIZATION – PROFESSIONAL STUDENT TRAINING

Viktor Matanski, Anton Iliev, Todorka Terzieva

Abstract. *The popularity of Generative Artificial Intelligence (GenAI) services sparked interest in applying these new tools to a variety of practical enterprise applications. New types of generative AI are continuously emerging and being used to generate images, video, audio, synthetic data, translation into different languages, etc. In this paper, we present the main highlights and results achieved in the training of students from computer science majors at FMI in the elective course "Introduction to Generative Visualization". Basic techniques from mathematics, physics, and fine arts that are used by artists, designers, and musicians in the creative industry are discussed. Some generative compositions created by the students in the Processing visual programming environment and printed on a 3D printer are presented.*

Key words: Generative Visualization, Generative AI, Generative Design, 3D Printing.

1. Introduction

Fundamental changes in the education system stem from the understanding of new educational values as well as the need for continuous learning based on innovative technologies. In everyday life one is faced with the need to solve concrete problems. Problem solving is inherent in every scientific field and academic discipline. However, each discipline is defined by the specific types of problems it addresses and the methodology it uses to address them.

Since 2013, the focus of curricular design has moved from what is taught (a knowledge model) to what is learned (a competency model). Computer Science Curricula Guidelines 2013 [1] contained 163 knowledge units grouped into 18 knowledge areas. Learning outcomes were identified for each knowledge unit. Distinction has been made between core topics that every computer science graduate must know and elective topics that were considered optional. Core topics were further divided into Tier 1 topics that were to be covered completely and Tier 2 topics, at least 80% of which had to be covered. Some early efforts to design a competency model of a curriculum were for Software Engineering and Information Technology [2]. The broader Computing Curricula CC2020 report proposed a competency model for various computing disciplines, including Computer Science, Information Systems, and Data Science. Competency models followed for Information Systems, Associate degree CyberSecurity and Data Science [3]. A knowledge model with its initial emphasis on content and a competency model with emphasis on outcomes are complementary views of the same learning continuum.

Other recent model undergraduate curricula for computer science include that of the All India Council for Technical Education [4] and the "101 plan" of the Ministry of Education in China [5]. Core subjects in the curriculum are subjects that every graduate student should know. Each syllabus is expected to cover all core topics. In addition to covering all the core topics, the syllabus is expected to cover a significant percentage of electives. Students may be expected to demonstrate proficiency in topics at various skill levels. Usually, Bloom's taxonomy is used to describe skill levels. The CS2023 knowledge model consists of 17 knowledge areas, listed in alphabetical order of their abbreviation: Artificial Intelligence (AI), Algorithmic Foundations (AL), Architecture and Organization (AR), Data Management (DM), Foundations of Programming Languages (FPL), Graphics and Interactive Techniques (GIT), Networking and Communication (NC), Operating Systems (OS), Parallel and Distributed Computing (PDC), Software Development Fundamentals (SDF), Software Engineering (SE), Security (SEC), Society, Ethics, and the Profession (SEP), Human-Computer Interaction (HCI), Mathematical and Statistical Foundations (MSF), Systems Fundamentals (SF) and Specialized Platform Development (SPD). In CS2023, core topics are categorized as either CS Core or KA Core (Knowledge Area). Elective topics are renamed Non-core.

In this paper, we present the main highlights and results achieved in the training of students from computer science majors at FMI in the elective course "Introduction to Generative Visualization". Basic techniques from mathematics, physics, and fine arts that are used by artists, designers, and musicians in the creative industry are discussed. Some generative compositions created by the students in the Processing visual programming environment and printed on a 3D printer are presented.

2. Major topics in the Curriculum

Societal practice requires that the general theory of computer science be adapted to different subject areas – business, finance, medicine, engineering automation, education, music, and others. By using the tools and methods of informatics, the development of alternative thinking abilities is stimulated, a variety of skills are formed, strategies are developed to search for solutions to both educational and practical problems, and the results of the decisions made are predicted based on the simulation of the objects, phenomena, processes, and relationships between them under study. The popularity of Generative Artificial Intelligence (GenAI) services sparked interest in applying these new tools to a variety of practical enterprise applications. New types of generative AI are continuously emerging and being used to generate images, video, audio, synthetic data, translation into different languages, etc [6].

The elective discipline "Introduction to generative visualization" taught in the Faculty of Mathematics and Informatics in the Plovdiv University "Paisii Hilendarski" gives a chance to the students to bring forth their creativity and imagination, by mixing main principles from the fields of mathematics, physics visual arts and generating both digital and physical creations.

In response to the huge rise in the usage of artificial intelligence for the creation of images and 3D models, the discipline emphasizes on the practical use of Processing (medium for visual programming) and the direct creation and further visualization of complex algorithms, mathematical functions and generative works in real time.

2.1. Main topics of the curriculum

The aim of the discipline is to introduce its students to the world of generative visualization [7], by encouraging them to explore different areas of science, using programming as a leading instrument. The discipline's curriculum is structured in such a way so it can empower the participants to further their imagination, to experiment freely and to use art to solve various issues. The main topics of the discipline are:

- Introduction to generative visualization – emphasis is placed on the creation of images, graphics and visual information mediums by utilizing algorithms and computational processes. The history of this contemporary art form is being further explored by analysis and programming of the works of Manolo Gamboa Namon [8], Michael Hansmeyer [9] and Katharina Brunner [10]. Different mathematical formulas and models, as well as randomness algorithms are used to create unique and often unpredictable work results.

- Systems built on randomness – randomness is used as a stepping stone for the creation, development and steering of processes and results. These systems find application in art, science, technologies and other fields, where randomness can add diversity, innovation and realism.
- Symmetry – fundamental properties of objects, systems and structures are explored and how their harmony, balance and repeating patterns are built. Symmetry is found in nature, art, mathematics, architecture. It is often linked to aesthetics and effectiveness, as symmetrical forms are accepted as beautiful and steadfast. Mirroral, rotational, translational, and group symmetry are examined.
- Particle systems – by using object-oriented principles, groups of models are created, which act as a "swarm". These systems are used in the film and gaming industries, architecture, 3D visualisations and other fields to study, simulate and dissect complex interactions of many small particles. They can replicate physical objects, natural phenomena and conceptual structures.
- Sound and image processing – different techniques for the manipulation, analysis and transformation of both images and sounds are demonstrated. The newly acquired data can further be used for the orchestration of particle systems [11, 12].
- Fractals – building on top of the symmetry, here we aim to explore the different properties of objects by decomposing them into self similar parts on the breach of endless complexity.
- 3D printing – as the last part of the course, this topic explores the mechanisms of FDM 3D printing. By this moment, the participants in the course should be able to freely use several different concepts of generative visualization and be able to compose a work that can then be 3D printed and interacted with [13].

By using Processing [14], students can easily manipulate their programming code and achieve instant visual effects, like organic growth, wave simulation or geometric tesselation. Changes in real time allow students to dynamically adapt the parameters of their programs, observing how small corrections can drastically change the final outcome. This way they have the ability to relive programming in a more intuitive, interactable and expressive process. This way students receive the opportunity to improve their programming abilities in an iterative way, by utilizing feedback in real time of instantly generated, visual results.

2.2. Acquired knowledge and expected results

After completing the course and successfully defending and presenting the course assignments, students will know:

- how to define and create works of generative visualization and art. They will have already viewed and recreated a sufficient number of works by proven authors, as well as be familiar with the works of their colleagues [15].
- how to identify, describe, measure, discuss and distinguish generative works and creative computing systems. Students will be able to:
- define and apply algorithms used in generative visualization, applying the following paradigms: randomness, noise, rule-based systems, cellular automata and artificial life. Each student receives a specific algorithm that they must develop and include in each of their works (Figure 1).
- design, implement and test generative systems from beginner to intermediate level. The learning process introduces students to a sufficiently deep level to be able to understand and distinguish generative principles in many different fields. Where is symmetry applied? How is it built? How, although different, multiple objects can belong to a single system of particles.

The course also introduces students to the potential of physical generative designs through 3D printing, providing a real-world application for their digital compositions (Figure 2). Translating their designs from digital to physical form requires students to consider practical considerations such as the properties of the material they are printing, its structure, and its scale. This experience emphasizes the value of interdisciplinary skills, helping students gain a fuller understanding of how generative AI can extend beyond screens and impact fields such as architecture, product design, and medicine and geography.

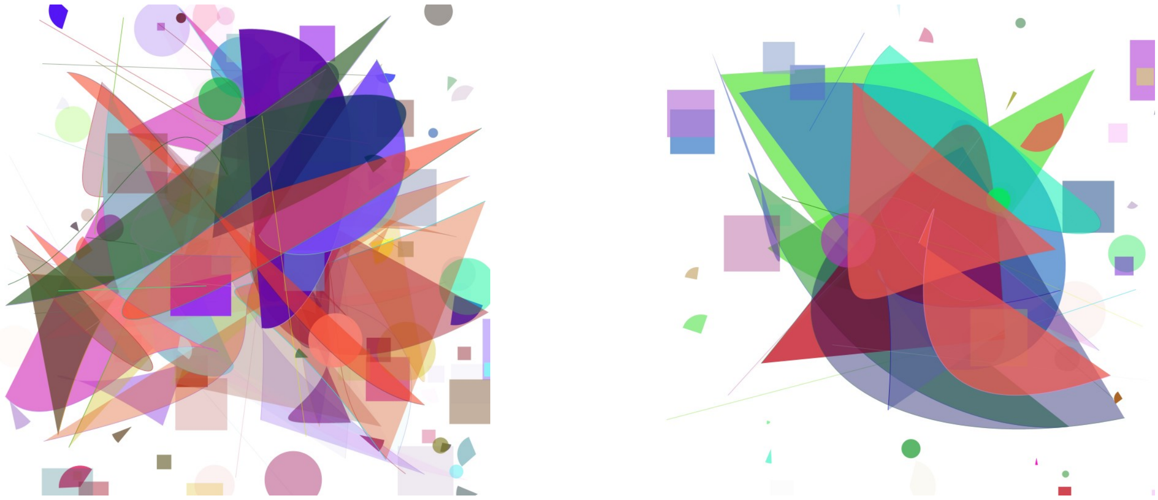

Figure 1. Generative composition "In the footsteps of Kandinsky"

The exam part of the course includes both individual assignments and collaborative group projects that encourage the creative process. Individual assignments allow students to delve into specific generative techniques and concepts independently, while group projects encourage collaboration, helping participants combine skills in mathematics, design and programming. By presenting their final projects to each other, the course creates an environment that mimics professional creative and technical situations, where direct feedback and teamwork are essential for achieving better results.

3. Student-created digital compositions and 3D projects

Key student projects illustrate the diversity of approaches encouraged by the course. For example, some students used fractal algorithms to generate tree-like patterns that evolved based on recursive rules, achieving organic visualizations that mimic the complexity of nature.

Others experimented with physics-based particle systems, creating dynamic, fluid visualizations that shift and interact in real time, simulating natural movements such as iridescent colors or waves. A geometric tessellation project inspired by Islamic art demonstrated the potential of generative AI for structured, symmetrical designs, producing complex 3D wall panels suitable for decorative purposes. These projects highlight how students combine technical rigor with personal creativity, gaining confidence in their ability to shape abstract algorithms into meaningful compositions.

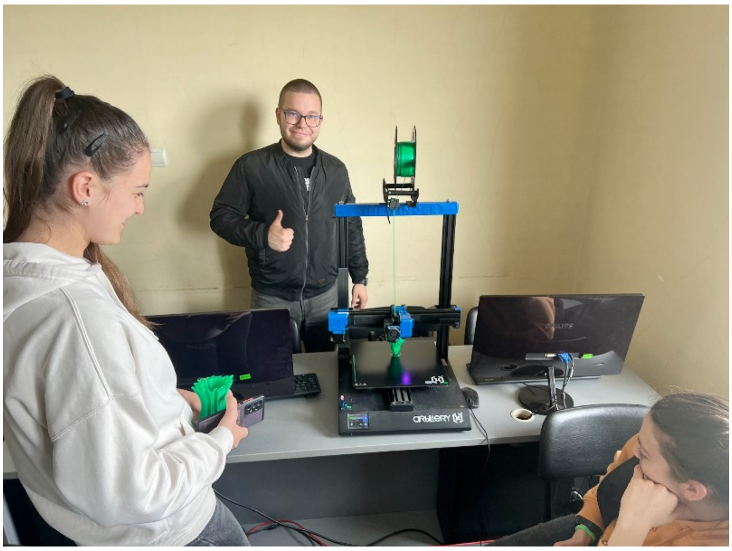

Figure 2. 3D printing generative models

Generative visualization requires multiple attempts to achieve desired effects, and participants are encouraged to view mistakes as a fundamental part of the learning process (Figure 2). Structured feedback sessions foster this mindset as students refine their designs based on input from peers and faculty, embracing the iterative nature of creative work. This focus on change and exploration builds a foundation of problem-solving skills and encourages the development of critical thinking.

By sequentially introducing different concepts, course participants gain confidence in their basic skills before moving on to more advanced applications, and by supporting each other through group work, they build a community of shared processes that enhance their understanding. Students created 3D printed bracelets, each with a unique generative design based on a faculty number (Figure 3).

4. Conclusion

In terms of educational impact, the course provides students with a diverse set of skills, combining technical programming skills with visual aesthetics and creative problem-solving. Many students report a shift in perspective, viewing programming as a means of creativity and expression rather than simply a technical discipline. The experience inspires them to consider career paths that merge technology with art, helping them envision roles in which generative visualization and generative artificial intelligence can play a transformative role.

Figure 3. Participants in the course

In conclusion, the course Introduction to Generative Visualization offers a progressive educational model that prepares students for an interdisciplinary and AI-driven future [16]. Through creative coding, hands-on experiments, and interdisciplinary projects, the course fosters students' innovation, adaptability, and artistic confidence, building the skills and competencies needed to integrate into today's digital and physical worlds.

Acknowledgments

This study is financed by the European Union-NextGenerationEU, through the National Recovery and Resilience Plan of the Republic of Bulgaria, project No. BG-RRP-2.004-0001-C01.

References

[1] ACM/IEEE-CS Joint Task Force on Computing Curricula. Computing Science Curricula 2013. New York, USA: ACM Press and IEEE Computer Society Press, 2013.

[2] M. Sabin, H. Alrumaih, J. Impagliazzo, B. Lunt, M. Zhang, B. By-

ers, W. Newhouse, W. Paterson, C. Tang, G. van der Veer, B. Viola, Information Technology Curricula 2017: Curriculum Guidelines for Baccalaureate Degree Programs in Information Technology. Association for Computing Machinery, New York, NY, USA, (2017).

[3] https://csed.acm.org/ (visited 29.11.2024).

[4] https://iiitd.ac.in/sites/default/files/docs/aicte/AICTE-CSE-Curriculum-Recommendations-July2022.pdf (visited 29.11.2024).

[5] Y. Liu, Q. Xiang, J. Chen, M. Zhang, J. Xu, Y. Luo, Undergraduate Computer Science Education in China, *ACM Inroads*, Vol. 15, 1, March 2024, 28–36.

[6] H. Sætra, Generative AI: Here to stay, but for good?, *Technology in Society*, Vol. 75, 2023, 102372, ISSN: 0160-791X, https://doi.org/10.1016/j.techsoc.2023.102372.

[7] A. Selçuk, Generative Art, *Geometric Patterns with Creative Coding*, 2023, 21–25, DOI: 10.1007/978-1-4842-9389-8_3.

[8] M. Namon, https://www.katevassgalerie.com/manolo-gamboa-naon (visited 29.11.2024).

[9] M. Hansmeyer, https://www.michael-hansmeyer.com/ (visited 29.11.2024).

[10] K. Brunner, https://www.katharinabrunner.com/ (visited 29.11.2024).

[11] V. Matanski, Generative Visualization Based on Sound, *Doctoral Conference in Mathematics and Informatics MIDOC 2015*, October 15–18, 2015, Sofia, Bulgaria, ISBN: 978-954-07-4186-4.

[12] A. Cmrk, N. Loknar, L. Mandić, GENERATIVE ART METHODS FOR IMAGE CREATION, *INTERNATIONAL SYMPOSIUM ON GRAPHIC ENGINEERING AND DESIGN*, 2024, DOI: 10.24867/GRID-2024-p36.

[13] F. Talaat, E. Hassan, Artificial Intelligence in 3D Printing, *Enabling Machine Learning Applications in Data Science*, Algorithms for Intelligent Systems, Springer, Singapore, 2021, 77–88, https://doi.org/10.1007/978-981-33-6129-4_6.

[14] Processing Foundation https://processing.org/ (visited 29.11.2024).

[15] A. Hertzmann, Generative Models for the Psychology of Art and Aesthetics, *Empirical Studies of the Arts*, Vol. 43, Issue 1, 2024, 23–43,

https://doi.org/10.1177/02762374241288696.

[16] M. Kuo, P. Lee, Generative and Adversarial: Art and the Prospects of AI, *October*, No. 189, 2024, 3–5, https://doi.org/10.1162/octo_a_00525.

Viktor Matanski[1], Anton Iliev[2], Todorka Terzieva[3],
[1,2,3] Paisii Hilendarski University of Plovdiv,
Faculty of Mathematics and Informatics,
236 Bulgaria Blvd., 4027 Plovdiv, Bulgaria
Corresponding authors: matanski@uni-plovdiv.bg