

Research Article

Teaching Model Design of Computer Programming Courses for Digital Media Technology Students

Qianyi Zhan,^{1,2} Jing Wang ,^{1,2} Xiang Pan ,^{1,2} Yanrui Ding ,³ and Yuan Liu ^{1,2}

¹School of Artificial Intelligence and Computer Science, Jiangnan University, Wuxi, China

²Jiangsu Key Laboratory of Media Design and Software Technology, Wuxi, China

³School of Science, Jiangnan University, Wuxi, China

Correspondence should be addressed to Yanrui Ding; yr_ding@jiangnan.edu.cn

Received 13 December 2021; Revised 30 December 2021; Accepted 3 January 2022; Published 24 January 2022

Academic Editor: Xin Ning

Copyright © 2022 Qianyi Zhan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Digital media technology major is aimed at improving students' skill on technological aspects of multimedia and developing their artistic and creative thinking. Therefore, the fundamental programming courses, including C, C++, Java, and Python, should meet the demand of digital media technology. In this paper, we design the teaching model of programming courses, which consists of three parts: theoretical research on the deep integration of programming course cluster and digital media, teaching and learning method of programming courses led by "art-technology integration," and teaching quality control and evaluation system of programming course cluster.

1. Introduction

Digital media have assumed an important place in our society, giving rise to a booming industry and generating leaps in computer software and hardware development. The degree of digital media technology focuses on the technological aspects of multimedia in the context of computer science. Students of digital media technology not only need to master information science and digital technology but also need to have a unique artistic and creative thinking. Therefore, the teaching courses of digital media technology should take computer science as the skeleton and serve artistic creativity with deeply integration of technology and art.

The courses of digital media technology major mainly include programming, data structure, database, operating system, computer graphics, and computer vision. Among them, advanced language programming series courses, for example, C, C++, Java, and Python, are the fundamental courses and the "top priority" in the syllabus of digital media technology major. Therefore, how to combine programming courses with the professional characteristics of digital media technology, and developing a programming course system

suitable for the needs of digital media technology is the key for instruction design of programming series course.

Recent years, a collection of research from many different perspectives has been devoted to the programming teaching [1–5]. In [3], an introductory object-oriented programming course is designed according to results of cognitive science and educational psychology. [6] defined "extreme apprenticeship" which has a strong emphasis on guided programming exercises. [7] is aimed at improving the challenge that allows and requires the participants to be more active in their online learning of CS1 programming course. [8] discussed programming exercises tools provide automated feedback on student solutions. It performs a systematic literature review to find out what kind of feedback is provided, which techniques are used to generate the feedback, how adaptable the feedback is, and how these tools are evaluated. They made much progress on programming courses, while as far as we know, few work has focused on digital media major and addressed the teaching problem related to different programming courses and optimized them as a course cluster.

The instruction design of programming series courses for digital media students in different universities depends

on their use of programming, for example, image processing, game design, and animation and movie production. Therefore, the curriculum of programming courses is determined according to the cultivating objective, i.e., what kind of digital media technology students are cultivated.

In this paper, we propose the teaching model of programming courses for digital media major, which education goal is cultivating applied, compound and innovative talents with solid foundation, excellent quality, strong ability, and rich personality in media and related fields. Based on the above training objectives, the curriculum includes C, C++, Java, and Python into the programming course cluster and is aimed at deep integration of the courses with digital media major.

The teaching model consists of three parts: theoretical study of deep integration of basic programming course cluster and digital media, teaching method study of programming courses led by “art-technology integration,” and the teaching quality control and effect evaluation system of the programming courses. The paper is structured as follows. Section 2 provides a brief overview of the cultivating model framework. Sections 3–5 give an introduction to the three parts of the model: teaching theoretical study, teaching method design, and teaching quality control. The last section is the conclusion.

2. Overall Framework

As Figure 1 shows, the cultivating model proposed in this paper is carried out from three aspects: theoretical research on the deep integration of programming course cluster and digital media, teaching and learning method of programming courses led by “art-technology integration,” and teaching quality control and evaluation system of programming course cluster.

In the teaching theoretical study, we first analyze the core value of programming course cluster for the cultivation of digital media technology students. Then, based on cognition and learning theories, we proposed the approach for deep integration of the programming course cluster with digital media.

In the teaching and learning method design, the project-driven induction method is introduced in the teaching model to stimulate students’ interest and improve learning efficiency. With large-scale experiments or projects as the core, grouping and collaboration methods are used for students’ learning programming courses.

In the quality control part, we establish a strict management system to monitor the quality of teaching programming course clusters. Based on the system, we analyze and diagnose problems in teaching process, achieve the purpose of continuous improvement of teaching work, and enhance teaching quality.

The implementation scheme of the teaching model design for programming courses is shown in Figure 2. In the following three sections, we will introduce these three parts in detail.

3. Teaching Theoretical Study

In-depth understanding of the cultivation mode of digital media technology talents and the society demand of digital

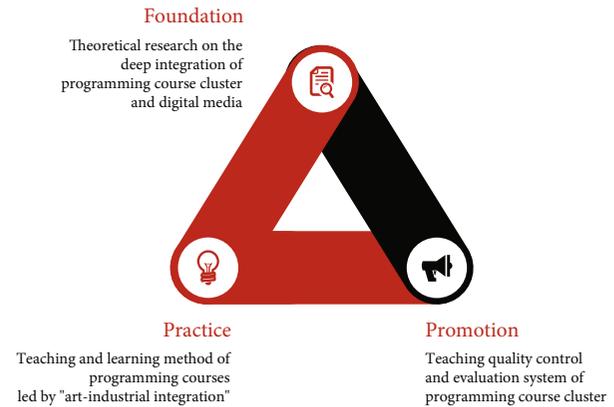


FIGURE 1: The framework of the cultivating model.

media technology talents is the basis for constructing the deep integration between the basic curriculum group of programming class and digital media. We first conduct the visits to relevant universities for research and in-depth analysis of the integration relationship between the programming courses involved in digital media technology and other courses. Then, we also make research with companies in the area of digital media such as entertainment games, interactive product innovation, internet products, e-commerce, animation design companies, digital audio-visual, and advertising media. It helps us to understand the development steps and essential features of different applications related to digital media, such as scene design, character image design, game programming, multimedia postprocessing, human-computer interaction, and creative design. Furthermore, we can analyze the demand of digital media technology and determine the depth and breadth of the curriculum of programming course cluster.

After the requirement analysis of the companies, based on the cultivating objectives of digital media major, we further study the training program, syllabus, and teaching tasks. The relationships between programming course cluster and specialized courses such as image processing, human-computer interaction, interactive media, animation principle and design, animation dynamics and kinematics, digital display, three-dimensional modeling, and game programming are well analyzed. The requirements of these higher level courses are clarified in this stage, and then, a curriculum architecture with deep integration of programming class basic course cluster and digital media is established (Figure 3).

As Figure 3 shown, programming course cluster, including C, C++, Java, and Python, is the foundation of digital media students. C is a common foundation for science and engineering majors, as it can be used to write working system programs and as a design language for applications. The C++ is ideal for processing image graphics and streaming files in digital media because of its direct manipulation of memory addresses and the high speed of compiling. The cross-platform nature and high security make Java an important programming language for game development and system design. Python, a popular programming

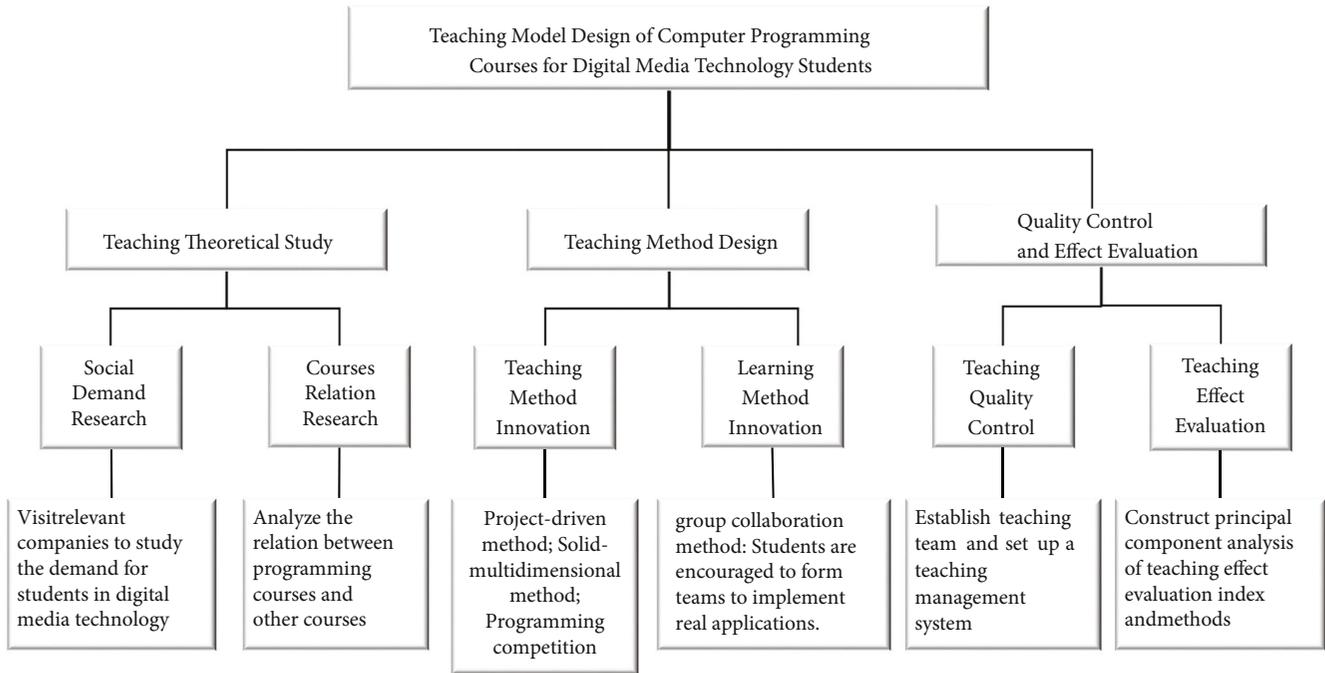


FIGURE 2: The implementation scheme of the teaching model design for programming courses.

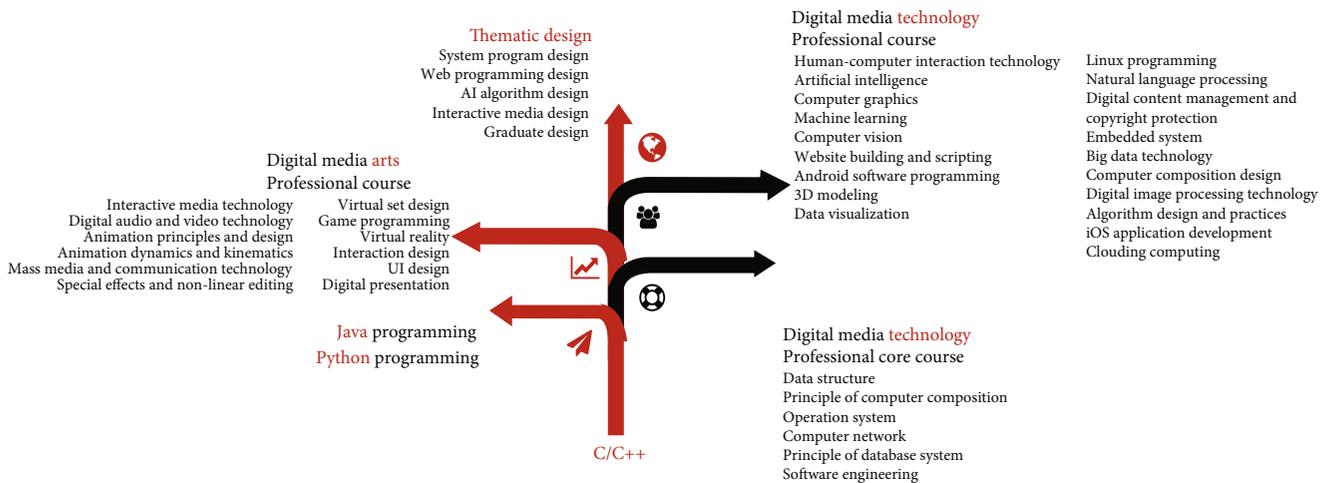


FIGURE 3: The curriculum architecture with deep integration of programming class basic course cluster and digital media.

language today, encapsulates many functional packages needed for virtual reality development and is suitable for the design and development of VR works in digital media. It is worth mentioning that Java and Python are glue languages which can be extended and embedded in combination with other languages such as C/C++.

4. Teaching Method Design

The curriculum design of programming course cluster ensures that the teaching content is useful for digital media students, while in the teaching process, there is a great connection between the method of teaching and the effect. Therefore, in this section, we focus on the method design of teaching programming course cluster.

For whatever programming language in the cluster, C, C++, Java, or Python, students are required to master not only the language grammar and basic algorithms, but also the ideas, approach, and skill of programming. Therefore, it is necessary to improve programming skills and consolidate knowledge through experiments and practices. Most of programming courses are introductory, but this traditional teaching methods are difficult to attract students. We propose the project-driven method, solid-multidimensional method, and programming competition to enhance the teaching effect.

4.1. Project-Driven Method. For certain points in programming, teachers identify a series of research-related projects and design teaching activities. The teacher and students

work together on a complete project to consolidate the language grammar and understand the connections among knowledge points. The teacher plays a leading role in the implementation of the project, while students are the core of the project, allowing them to apply knowledge and experience to solve practical problems and realize applications in real scenario. From the project, students are motivated by achievement and improve their self-learning ability.

4.2. Solid-Multidimensional Method. We also utilize the solid-multidimensional method in the teaching of programming courses. The method integrates situational teaching, heuristic teaching, and interest-oriented teaching. Firstly, it emphasizes the diversity of teaching forms. The introductory class in the classroom, experiment, after-school assignment, and flexible tasks are integrated, equally important and inseparable. Secondly, it focuses on the diversity of teaching mode. The introductory class teaches the knowledge structure of the programming course, and accomplishing experiments helps students find out the connection among knowledge points. Last, the diversity also reflects on the interaction forms between teacher and students. Besides the traditional face to face, teachers and students nowadays can utilize email, social media, online video meeting, and other applications to learn and communicate.

4.3. Programming Competition. The programming competition is a useful way to check the comprehensive abilities of students. To achieve the goal of “art-technology integration,” the competition subject is proposed by both programming course teachers and media art teachers. Students are asked to solve the problem using any programming language within a specified period of time. Through the competition, we aim to promote students’ ability of problem analysis, paper research, algorithm design, programming, and teamwork.

4.4. Group Collaboration Learning. In addition to the innovation of teaching method, we also develop the student learning model for programming course cluster, called the group collaboration learning method. Students are encouraged to form same-grade teams and mixed-grade teams to design and implement real applications. During the process, students can improve the ability of solving problem. More importantly, they can assess themselves, learn from each other, complement each other, and conduct self-learning and improvement. For example, students are asked to design a 3D game, and different parts are implemented by different programming languages. The project is first divided into several parts: the image processing part, the interaction part, the animation design part, etc. These parts are assigned to one same-grade team and one mixed-grade team, respectively. By working together to complete the tasks, students learn more about the features of each programming language. The “collaborative group learning method” maximizes the depth of integration between the programming course cluster and digital media.

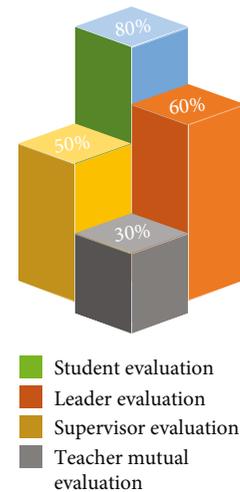


FIGURE 4: The evaluation system with four directions.

5. Teaching Quality Control and Effect Evaluation

In this section, we will introduce the teaching quality control and effect evaluation, which is the important part of the teaching model of programming courses.

5.1. Teaching Quality Control. Motivated by the cultivating goal of digital media students, we establish the teaching discussion group for the programming courses. According to the requirement of major development, the teaching team is constantly strengthened according to the principle of “training, improving, stabilizing, and introducing.” The teaching level of young teachers is constantly improved by various means, such as collective class preparation, learning from senior teachers, attending other’s lectures, and participation in teaching research and reform. Teachers of programming course are encouraged to undertake and participate in research projects and study from teachers’ digital media art. We introduce programming-related contents of projects into the classroom, internship, practical training, and graduation design sessions to promote teaching.

Moreover, we also set up a strict teaching management system. In order to guarantee the teaching quality, college, the leader of programming course cluster, and the course teacher form a three-level management system. Each level has clear tasks, and all levels are well coordinated. The teaching discussion group formulates the construction content and development goals of the programming course cluster. The leader of group decomposes the tasks and implements them to each course teacher, while the college is mainly responsible for inspecting and supervising the construction of the curriculum group.

5.2. Teaching Effect Evaluation. We also build a teaching quality evaluation system based on the innovative teaching mode proposed in this paper including project-driven teaching method, solid-multidimensional method, programming competition, and group cooperation method. The evaluation system consists of four directions (shown in Figure 4):

student evaluation, leader evaluation, supervisor evaluation, and teacher mutual evaluation. Student evaluation is the core of the overall system, and other three directions can be increased or decreased according to the real situation of the teaching quality evaluation work. We design the questionnaires for four directions, and corresponding evaluation weights are given according to the specific questions.

The correlation matrix of principal component analysis is formed from different course teaching quality evaluation indicators, and different samples are ranked according to the product of linear combination and contribution of different principal components determined after principal component analysis for teaching effect evaluation.

6. Conclusion

In this paper, we propose the teaching model of computer programming courses for digital media technology students. The model has three parts: theoretical study of deep integration of basic programming course cluster and digital media, teaching method study of programming courses led by “art-technology integration,” and the teaching quality control and effect evaluation system of the programming courses. We have utilized this model on the teaching process in our university and received good response from both students and teachers. In the future, we will modify the model according to the teaching quality.

Data Availability

The dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work is supported by the NSFC 61902152.

References

- [1] C. Watson, F. W. B. Li, and J. L. Godwin, “Predicting performance in an introductory programming course by logging and analyzing student programming behavior,” in *2013 IEEE 13th international conference on advanced learning technologies*, pp. 319–323, Beijing, China, 2013.
- [2] C. McDowell, L. Werner, H. Bullock, and J. Fernald, “The effects of pairprogramming on performance in an introductory programming course,” in *Proceedings of the 33rd SIGCSE technical symposium on Computer science education*, pp. 38–42, Cincinnati Kentucky, 2002.
- [3] M. E. Caspersen and J. Bennedsen, “Instructional design of a programming course: a learning theoretic approach,” in *Proceedings of the third international workshop on Computing education research*, pp. 111–122, Atlanta Georgia USA, 2007.
- [4] R. Rajaravivarma, “A games-based approach for teaching the introductory programming course,” *ACM SIGCSE Bulletin*, vol. 37, no. 4, pp. 98–102, 2005.
- [5] D. Wei and A. N. Burrows, “Tracking students’ performance to assess correlations among computer science programming series courses,” *Journal of Computing Sciences in Colleges*, vol. 32, no. 1, pp. 9–16, 2016.
- [6] A. Vihavainen, M. Paksula, and M. Luukkainen, “Extreme apprenticeship method in teaching programming for beginners,” in *Proceedings of the 42nd ACM technical symposium on Computer science education*, vol. 93, Dallas TX USA, 2011.
- [7] A. Vihavainen, M. Luukkainen, and J. Kurhila, “Multi-faceted support for mooc in programming,” in *Proceedings of the 13th annual conference on Information technology education*, pp. 171–176, Calgary Alberta Canada, 2012.
- [8] H. Keuning, J. Jeurung, and B. Heeren, “Towards a systematic review of automated feedback generation for programming exercises,” in *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education*, pp. 41–46, Arequipa Peru, 2016.