Teaching portfolio

Uwe Hahne

July 27, 2026

1 Preface

This teaching portfolio is intended to provide an overview of my teaching activities to date and to convey what teaching means to me and what distinguishes my courses.

During my studies at Bauhaus University Weimar, I gained my first teaching experience in media informatics. This sparked my interest in continuing to work in teaching and also showed me the challenges that must be addressed. From summer 2006 to 2012, I have been employed at the Computer Graphics department at Technische Universität (TU) Berlin. There, for various courses, I have conducted tutorials and lectures, as well as organizationally managed practical sessions and helped shape their content. After finishing my PhD in 2012, I joined the SICK AG in Waldkirch, where I was responsible for the development of software for 3D snapshot cameras. In 2017, I became a group leader and received the opportunity to work as servant leader for experienced developers. I have come to recognize that supporting the growth of others is among the most rewarding aspects of my profession and represents a genuine passion of mine. In September 2021, I was appointed professor for Computer Vision in Media Applications at Hochschule Furtwangen (HFU). Since then, I have been able to teach a wide range of courses in media informatics and computer vision, as well as to supervise numerous theses.

In this teaching portfolio, I first outline my fundamental perspectives on teaching and my approaches to effective instruction. I then describe selected courses in detail and discuss the methods I employ. Finally, I address opportunities for enhancing teaching and offer an outlook on my future teaching activities.

2 Teaching Principles

Teaching in computer science includes not only the transfer of extensive specialist knowledge, but also the development of social skills. In programming, which is a core area of computer science, many work steps are carried out in teams, and even smaller projects require clearly defined interfaces between individual program components. This requires communication skills in dealing with colleagues in order to ensure successful completion. In teaching, this aspect of collaboration, which I consider very important, is often neglected. I try to counteract this and place great value on communication among participants in my courses as well.

Recently, AI tools have taken over many tasks in programming, so that students could simply let an AI agent solve their exercises. Therefore, I place great emphasis on the fact that students are able to explain their own solutions and defend them in discussions with other participants. Personally, I am ambivalent about the use of AI tools in teaching. On the one hand, I see great potential in them: there is no need to learn specific syntax of a language or all the parameters of a specific function in a extensive API. Also writing boilerplate code and write simple unit tests is done by an AI agent much faster. But on the other hand, I also see the danger that students will no longer learn to understand and apply programming methods independently. Also the moral aspect as AI models are obviously trained on biased and protected data leads to ethical concerns. Therefore, I try to find a balance between these two aspects in my courses.

A second principle that guides my teaching is the following: *A piece of work is only as good as its presentation.* In addition to internal communication, external presentation is another parameter in computer science that is often underestimated. Of course, conveying specialist knowledge and methods (in this case programming) is the essential part of teaching. However, students should be made aware as early

as possible that it is not enough to understand concepts well enough to solve various problems; the chosen solution strategy must also be defended and explained convincingly.

2.1 Content of Teaching

My courses can generally be divided into three categories:

A: Fundamentals of Computer Science
B: Computer Graphics/Computer Vision/Media technologies
C: Applied Computer Vision and Graphics in Seminars and Projects

In the first category A, I introduced first- and second-semester students to programming in a functional language and object-oriented programming through weekly tutorials at TU Berlin and later also with complete lectures and practical sessions at HFU.

The second category B includes courses that teach the fundamentals of computer graphics as well as computer vision. Here I designed and organized practical sessions. These consisted of programming assignments that had to be completed independently in small groups within a few weeks and then briefly presented to the instructor. In addition, theoretical questions had to be answered in writing. I presented the assignments in plenary sessions and was always available as a contact person. During small-group presentations, I could check understanding on the one hand, but above all clarify uncertainties and thereby support students in their learning.

In the third category C, I include project-based courses and seminars. Unlike lectures, projects begin with only one goal to be achieved. The path to that goal is to be developed independently by participants. This creates strong dynamics, and many different learning contents are taught in a participant-oriented way.

In Appendix A, Table 1 contains a complete list of all courses I was involved. It is indicated in each case which category the course belongs to.

3 Teaching Practice

3.1 Collaborative Work in Research Projects

My own studies in Weimar were project-oriented. Each semester, a research project was chosen and worked on intensively in a small group together with the professor. Relevant publications were sought out independently, read, then discussed and re-implemented in the group. In some cases, content was created that even led to scientific publications. This teaching approach was very inspiring and pleasant for me as a student, and at the same time extremely effective. I have passed on these very positive experiences in various projects at TU Berlin and HFU. I have tried to implement this alternative teaching format there as well. However, it proved difficult to carry out this mode of work in a single course when the overall degree program is structured differently. Nevertheless, in some projects I succeeded in creating a positive working atmosphere and in producing results that could later be published. Even though working very freely and independently was unusual for most students, feedback was generally very positive, and project participation often led to the writing of a thesis. Therefore, I see this teaching format as having the greatest potential for good teaching.

In the training course ‘Working with Problem-based Learning - Management of the Learning Process’ at the Center for Continuing Academic Education at TU Berlin, I found that this approach is very close to the Problem-Based Learning model developed at Aalborg university [Kolmos et al., 2004]. There, this teaching approach is implemented not at the level of individual courses, but as an overarching instructional concept at specific universities. This confirms my experiences at TU Berlin and HFU that if projects are organized as one of several courses and lectures, due to the many context changes the learning effect is minimized. From my experience as a Scrum Master, I therefore endorse the following rule: *one*

single project at a time. In order to avoid a loss of focus and hence working hours, in project management you try to minimize the number of context changes. Each change of context leads to some lost hours due to the time it takes to get back into the topic. In teaching, this means that a project should be worked on in a focused manner and not be interrupted by other projects or courses. This is realized in the program Games and Immersive Medis (GMB) at HFU, where students work in block periods of several weeks on a single course or project. We implemented this curriculum in accordance to EduScrum [Wijnands et al., 2020] after attending a training workshop by Willy Wijnand. After the first years, we gained both very positive experiences and feedback from students.

This approach appears to me to be the most effective way to teach students. In my opinion, this is the best way to learn as later at professional work, they will seldomly have the opportunity to work on a single project for several months. The university is usually the last place to learn focused and in depth about a single topic at a time and I will support students to gain this experience as much as possible.

3.2 Live Programming in classroom

As part of my work as coordinator of the practical sessions for the course “Methodical and Practical Foundations of Computer Science 2,” I introduced the use of projectors in seminar rooms for all tutorials - yes, it was not the standard equipment at TU Berlin in 2008. This makes it possible to demonstrate programming tasks interactively on a computer instead of developing source code only on the blackboard. This allowed us to discuss, together with students, how the code works and what results it produces directly. This eliminates the need for students to transfer blackboard notes to a computer at home and retrace them. Possible errors in the source code are identified immediately through direct checking and can then be handled right away, as is common in software development. This makes it possible, on the one hand, to avoid entirely that students continue working with faulty materials, and, on the other hand, students gain practical insight into how an experienced programmer works and can contribute their own experiences as well.

Nowadays, most students bring their own laptops to tutorials, so that they can follow the live programming directly on their own computers. This has proven to be very effective and is now standard in my courses. In order to create an optimal learning experience for all students, I try to ensure that every question is appreciated and answered. The slowest student should set the time of progress. I realized that this is not always easy, but it is important to me that no student is left behind. I also recognized that younger students are more used to type on a smartphone than on a keyboard. Hence, they need not just a little bit more time to find the right keys on a keyboard, they also have a mental context switch from understanding the coding task to typing the code. Therefore, I encourage students to learn typing before they learn to program. This is a very important skill that is often underestimated and neglected in teaching. In order to further reduce the mental load and context switches, I require German students to write all their code in English. Even the comments and all variable names must be in English. Otherwise students are using three languages (German, English and the programming language) at the same time.

3.3 Presentation Training

In seminars, students’ own presentations in front of other participants are very important to me. I offer individual support before and after the presentation. As described in the Section 2, I consider it indispensable to present good work accordingly. Here I support students as an experienced presenter in clarifying both subject-matter uncertainties and reflecting on content structure. After a student presentation, it is important to me that the presenter receives feedback. For this, I apply the classic feedback rules that encourage respectful and appreciative interaction.

4 Teaching Methods

4.1 Sandwich Principle

The sandwich principle according to Wahl [Wahl, 1991] is based on the fact that learners' attention span is relatively short compared to the duration of a double period; one assumes about 20 minutes. To support this, active and passive parts of the class are alternated. Passive and active refer to the learner, whose classic role as a passive listener is repeatedly turned into that of an active learner. Methods include small-group work, individual tasks, and open discussions. To complete the sandwich-principle model beyond the alternating *filling*, the teaching unit is enclosed by an introduction and a conclusion. The introduction provides an entry into the topic of the unit and gives an overview of what is to come. In the concluding part, the key points are summarized and recapitulated: what the take-home message is, and an outlook on the following units is provided. To implement this principle, I usually create some kind of screenplay for each leacture. You find an example of such a screenplay on my public course page for the course Digitale AV-Technik.

4.2 Embrace interaction

Tutorials, and especially group meetings in projects, are very well suited for conveying learning content through interaction between students. Here, subject-matter aspects are to be discussed and questioned so that learners, on the one hand, gain a deeper and more multilayered understanding of the material and, at the same time, learn to present and defend their views. Here, I aim to intervene solely in the role of moderator and only in cases of subject-matter uncertainty. In order to create an atmosphere in which students feel comfortable to express their opinions, I try to create a respectful and appreciative environment. I also try to ensure that all students are involved in the discussion and that no one is left out. In my experience, it is helpful to start the session with a small icebreaker, such as a short question or a small task that everyone has to answer. After that, the students evolve from passive listeners to active participants. In order to moderate such sessions, I benefit not only from my educational experience, but also from my experience as a Scrum Master. Here, I have learned to ask the right questions and to moderate discussions in a goal-oriented manner.

4.3 Agile principles

Working as a certified Scrum Master, I have learned to appreciate the agile principles and values of Scrum [Schwaber and Sutherland, 2020]. I have found that these ideas can be applied very well in teaching. In particular, the principles of self-organization and iterative development are very useful in project-based courses. In my courses, I present the learning goals as cards on a Kanban board. Before each session the goals are explained and placed in the *In progress* column. At the end, the cards are moved to *Done* if and only if the learning goal is clear to all participants and all questions are answered. See Figure 1 for an example of a Kanban board with the first learning goals of the course 3D Computer Vision at HFU.

When supervising theses, I rely on regular personal meetings with the student. A weekly rhythm seems optimal to ensure the independence of the work while at the same time avoiding development in technically incorrect directions. In a weekly meeting, the students answer the following four questions:

- How does it feel?
- What has been achieved since the last meeting?
- What is the next step?
- Do you need any support to achieve the next step?

These question follow the idea of the Daily Scrum [Schwaber and Sutherland, 2020] from the Scrum framework. The student is responsible for answering these questions, and I ask further questions to ensure understanding. This ensures that the student is always aware of the current status of the work and can reflect on it. In addition, I try to ensure that students are able to present their work in a structured way and can defend it in discussions with other participants. Many students tell me that this structure

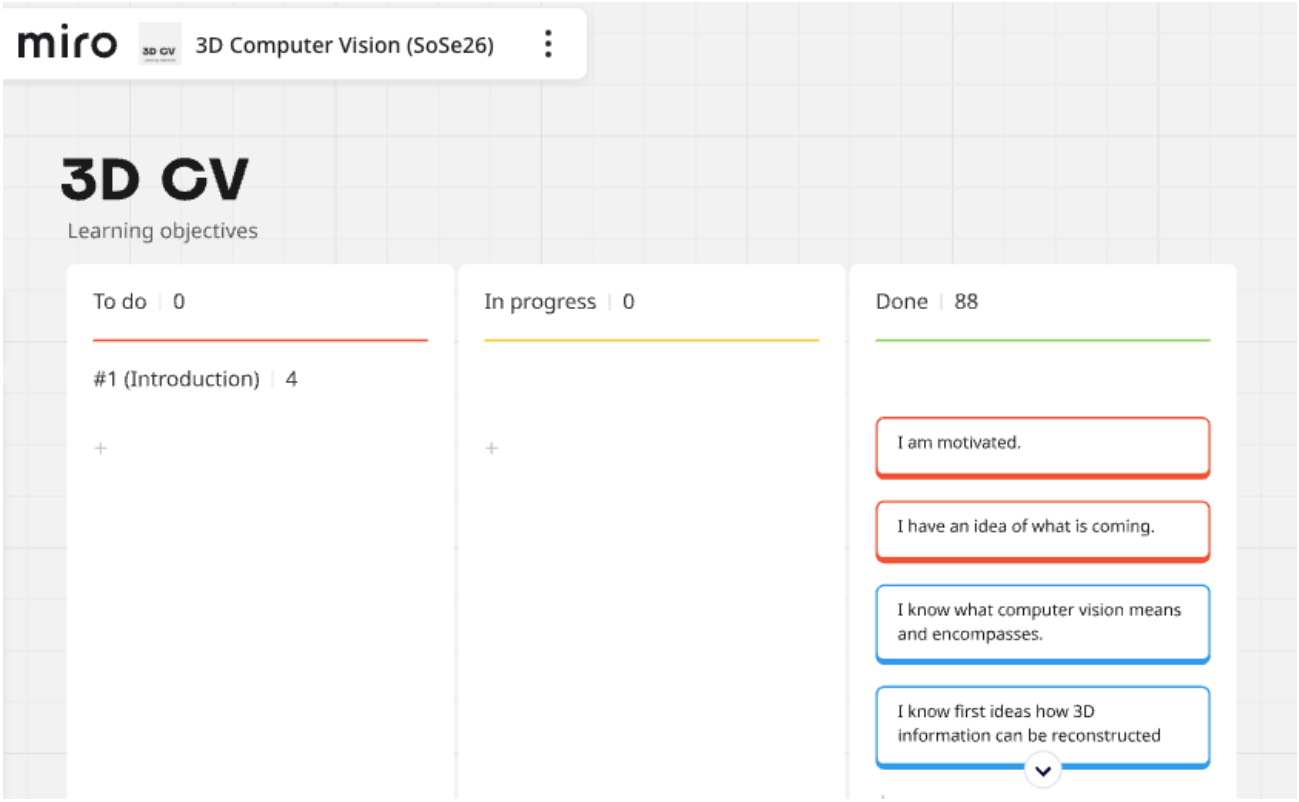

Figure 1: Screenshot of the Kanban board with the first learning goals of the course 3D Computer Vision at HFU.

helped them a lot to organize their work and to keep track of the progress. A complete list of all theses I have supervised is included in the Appendix C.

My teaching approach in general is based on the Scrum values of commitment, courage, focus, openness, and respect. I try to create an environment of trust in which students feel comfortable to express their opinions and to take responsibility for their own learning. I also try to ensure that students are able to work in a team and to learn from their peers. In my experience, this approach leads to a more effective and efficient learning process.

5 Enhancing Teaching

5.1 Muddiest Point Feedback

I regularly use opportunities to obtain feedback from students and, in seminars, I also encourage students to give each other feedback. A simple method has proven very effective: *the Muddiest Point*, which goes back to Frederick Mosteller [Mosteller, 1989]. Here, participants in a lecture are asked afterward which content point was least understandable. This question triggers learners to reflect on what they have heard and gives the teacher insight into which topic area is especially difficult to convey. Learning this is very important to me, because in my experience it is very difficult to reconstruct learners' difficulties once content has become internalized and self-evident for oneself, although this should of course be the learning goal. I can easily integrate the approach into my courses where I use the Kanban board mentioned above.

5.2 Further Training in Higher-Education Didactics

At the Center for Continuing Academic Education (ZEWK) at TU Berlin, a crash course called "Teaching for University's Best" on higher-education didactics was offered by Monika Rummler [Rummler, 2011]. With great interest, I have used various advanced training opportunities at ZEWK in order to repeatedly

reflect on my teaching and to learn about new methods and discoveries in the field of higher-education didactics. For this, I received a qualified higher-education didactics certificate. Furthermore, I myself have supported continuing-education offerings at TU Berlin as an instructor.

After becoming professor at HFU, I have continued to attend various workshops on higher-education didactics offered by the Geschäftsstelle der Studienkommission für Hochschuldidaktik an Hochschulen für Angewandte Wissenschaften in Baden-Württemberg (GHD).

5.3 Reflection

In my first years as a professor, I have sought to compare my own learning trajectory with those of my students. I have come to recognize that the way I learned Computer Science and its applications differs substantially from how students learn today. The sheer volume of learning materials now available can be overwhelming, and resources such as video lectures by 3Blue1Brown or platforms like Brilliant are tools I would have greatly valued during my own studies. However, it is hard to find a clear path through the jungle of information and I want to help students to find their own path and to develop their own ideas. More recently, generative AI tools play an important role in the learning process. Now, there are also an overwhelming number of AI-powered resources available. We, as educators, have to find a balance between using these tools and ensuring that students learn the fundamentals of a field and more important critical thinking skills. The ability to judge the quality and correctness of information is becoming increasingly important. The key to this is to create a learning environment of trust and respect, where students feel comfortable to ask questions, discuss and to express their opinions.

6 Conclusion

In conclusion, my approach to teaching is characterized by a commitment to student-centered learning and the continuous improvement of my didactic practices. Through the integration of various methods and principles, mainly from the ideas of agile development and Scrum [Schwaber and Sutherland, 2020], I aim to establish trust and respect in an engaging and effective learning environment. The ongoing pursuit of further training and professional development in higher-education didactics underscores my dedication to staying current with best practices and contributing meaningfully to the field of education.

References

[Kolmos et al., 2004] Kolmos, A., Fink, F., and Krogh, L. (2004). *The Aalborg PBL Model: progress, diversity and challenges*. Aalborg Universitetsforlag, Denmark.

[Mosteller, 1989] Mosteller, F. (1989). The 'Muddiest Point in the Lecture' as a feedback device. *On Teaching and Learning: The Journal of the Harvard-Danforth Center*, 3:10–21.

[Rummler, 2011] Rummler, M. (2011). *Crashkurs Hochschuldidaktik Grundlagen und Methoden guter Lehre*. Beltz Weiterbildung. Beltz Verlagsgruppe, Weinheim.

[Schwaber and Sutherland, 2020] Schwaber, K. and Sutherland, J. (2020). The scrum guide: The definitive guide to scrum: The rules of the game. https://scrumguides.org/scrum-guide.html. Last visit: 2026-07-15.

[Wahl, 1991] Wahl, D. (1991). *Erwachsenenbildung konkret. Mehrphasiges Dozententraining. Eine neue Form erwachsenendidaktischer Ausbildung von Referenten und Dozenten*. Neue Formen des Lernens im Betrieb. Deutscher Studien-Verl., Weinheim.

[Wijnands et al., 2020] Wijnands, W., Struijlaart, C., Fritsch, K., Ponchon, E., Bredikhina, E., van Mensvoort, B., Orbitowska-Fernandez, P., Postma, M., and Valente Hervier, X. (2020). The eduscrum guide.

Appendix A Course overview

This overview summarizes courses, seminars, and project-based teaching across different programs and terms at Hochschule Furtwangen (HFU) and Technische Universität (TU) Berlin. Each entry includes the affiliation, term, course title, target program, and a brief description of the course content or focus as well as an indicator of the course's category - which I defined above in Section 2.1. The courses cover a range of topics in computer science, with a particular emphasis on computer graphics, computer vision, and media technologies.

The following program abbreviations are used in Table 1 below to indicate the target audience of each course:

- MIM = Media Computer Science (M.Sc.)
- DIM = Interactive Media Design (M.A.)
- MIB = Media Computer Science (B.Sc.)
- OMB = Online Media (B.Sc.)
- GMB = Games and Immersive Media (B.A.)
- MKB = Media Design (B.A.)
- INB = Computer Science (B.Sc.)
- OI = Orientierungssemester Informatik

Recently, I started to build accompanying public web pages for some of my courses, as the internal OpenOLAT system is cumbersome to use. These pages provide additional resources, lecture notes, and project information. These pages can be accessed at the following URLs:

- Digitale AV-Technik: https://uhahne.github.io/DAVT/
- KI in Medienanwendungen: https://uhahne.github.io/GenAI/
- 3D Computer Vision: https://uhahne.github.io/3DCV/

Table 1: Teaching Courses

Affiliation	Term	Title	Program (Category)	Description
HFU	Summer 2026	Computer Vision and Deep Learning	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Summer 2026	Digitale AV-Technik	MIB (B)	Foundations of digital audio/video technology, from signals and systems to practical algorithms and information theory.
HFU	Summer 2026	KI in Medienanwendungen	MIB (C)	Seminar on generative AI in media applications.
HFU	Summer 2026	3D Computer Vision	MIM (B)	Master-level computer vision course focused on 3D reconstruction, including practical workshops.
HFU	Summer 2026	Entry project	GMB (C)	Introductory project course on agile game development with Unreal Engine Blueprints.
HFU	Summer 2026	CODE 3	GMB (B)	Block course on real-time computer graphics and the modern rendering pipeline in WebGL and Godot.
HFU	Summer 2026	STEM 2	GMB (A)	Block seminar on mathematical simulations inspired by Ten Minute Physics.
HFU	Summer 2026	SmartMirror	MIM (C)	Master students research project on building a human-size display that mirrors people as 3D avatars.
HFU	Winter 2025	Computer Vision and Deep Learning	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Winter 2025	Digitale AV-Technik	MIB (B)	Foundations of digital audio/video technology, including systems, signals, and information theory.

Table 1: Teaching Courses (continued)

Affiliation	Term	Title	Program (Category)	Description
HFU	Winter 2025	KI in Medienanwendungen	MIB (C)	Seminar on generative AI in media applications.
HFU	Winter 2025	STEM 3	GMB (B)	Block seminar on physical computing with single-board computers and interactive installations.
HFU	Winter 2025	Computational thinking	OI (A)	Experimental seminar inspired by MIT Computational Thinking and related approaches.
HFU	Summer 2025	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Summer 2025	Digitale AV-Technik	MIB (B)	Foundations of digital audio/video technology, including systems, signals, and information theory.
HFU	Summer 2025	Point&Click Mystery	MIB, MKB (C)	Student project (second term) for creating a point-and-click adventure game.
HFU	Summer 2025	3D Computer Vision	MIM (B)	Master-level computer vision course focused on 3D reconstruction with practical workshops.
HFU	Summer 2025	Entry project	GMB (C)	Introductory project course on agile game development with Unreal Engine Blueprints.
HFU	Summer 2025	CODE 3	GMB (B)	Block course on real-time computer graphics and modern rendering in WebGL and Godot.
HFU	Summer 2025	STEM 2	GMB (A)	Block seminar on mathematical simulations inspired by Ten Minute Physics.
HFU	Winter 2024	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Winter 2024	Digitale AV-Technik	MIB (B)	Foundations of digital audio/video technology, including systems, signals, and information theory.
HFU	Winter 2024	Point& Click Mystery	MIB, MKB (C)	Student project about creating a point-and-click adventure game.
HFU	Winter 2024	Mathematik und Simulation Seminar	MIB (A)	Java programming seminar accompanying the second-semester math and simulation course.
HFU	Winter 2024	Computational thinking	OI (A)	Experimental seminar inspired by MIT Computational Thinking and related approaches.
HFU	Winter 2024	Creative Coding	MKB (A)	Introductory coding course for design-oriented students using Scratch, p5.js, and ESP32.
HFU	Summer 2024	3D Computer Vision	MIM (B)	Master-level computer vision course focused on 3D reconstruction.
HFU	Summer 2024	Entry project	GMB (C)	Introductory project course on agile game development with Unreal Engine Blueprints.
HFU	Summer 2024	Free-viewpoint 3D video creation	MIM, DIM (C)	Master research project on free-viewpoint 3D video creation.
HFU	Summer 2024	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Summer 2024	Grundlagen Interaktiver Systeme	OMB, MIB (A)	Lecture on web development basics (HTML, CSS, JavaScript, databases).
HFU	Summer 2024	Mathematik und Simulation Seminar	MIB (A)	Java programming seminar accompanying the second-semester math and simulation course.
HFU	Summer 2024	Mathematische Grundlagen für Computergrafik und Gestaltung	MKB (A)	Math lecture for the media conception bachelor program (second semester).

Table 1: Teaching Courses (continued)

Affiliation	Term	Title	Program (Category)	Description
HFU	Summer 2024	STEM 1	GMB, MKB (A)	Math lecture and practical seminar for first-semester students, including p5.js implementations.
HFU	Winter 2023	Free-viewpoint 3D video creation	MIM, DIM (C)	Propaedeutic course preparing a student research project on Free3D.
HFU	Winter 2023	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Winter 2023	Grundlagen Interaktiver Systeme	OMB, MIB (A)	Course on web development basics (HTML, CSS, JavaScript, databases).
HFU	Winter 2023	Mathematik und Simulation Seminar	MIB (A)	Java programming seminar accompanying the second-semester math and simulation course.
HFU	Winter 2023	Mathematische Grundlagen für Computergrafik und Gestaltung	MKB (A)	Math lecture for the media conception bachelor program (second semester).
HFU	Winter 2023	Projection Mapping in the wild	MIB, MKB, OMB (C)	Continuation project on large-scale projection mapping, presented at Reservoir festival.
HFU	Winter 2023	Aktuelle Technologien im Bereich Digitaler Medien	MIB, OMB (C)	Seminar evaluating current technologies such as NeRF and 3D Gaussian Splatting.
HFU	Summer 2023	Grundlagen Interaktiver Systeme	OMB, MIB (A)	Course on web development basics; first course taught using eduScrum.
HFU	Summer 2023	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Summer 2023	Mathematik und Simulation Seminar	MIB (A)	Java programming seminar accompanying the second-semester math and simulation course.
HFU	Summer 2023	Mathematik und Simulation Praktikum	MIB (A)	Practical math exercises accompanying the math and simulation course.
HFU	Summer 2023	Echtzeit-Computergrafik	MIB (B)	Scrum-based real-time graphics course with WebGL/WebXR development in agile teams.
HFU	Summer 2023	3D Computer Vision	MIM (B)	Master-level computer vision course focused on 3D reconstruction.
HFU	Summer 2023	Projection Mapping in the wild	MIB, MKB, OMB (C)	Student project on projection mapping at Linachtalsperre with external partners.
HFU	Summer 2023	Aktuelle Technologien im Bereich Digitaler Medien	MIB, OMB (C)	Seminar evaluating current technologies such as NeRF, code generation, and diffusion models.
HFU	Winter 2022	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Winter 2022	Zukunft digitaler Medien	MIB, MIM (C)	Seminar plus guest lecture series on research topics and their future impact.
HFU	Winter 2022	Mathematik und Simulation Seminar	MIB (A)	Java programming seminar accompanying the second-semester math and simulation course.
HFU	Winter 2022	MINT Praktikum	MIB (A)	Math tutorial course.
HFU	Summer 2022	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Summer 2022	Echtzeit-Computergrafik	MIB (B)	Scrum-based real-time graphics course with WebGL/WebXR development in agile teams.
HFU	Summer 2022	Komplexe Datenstrukturen	MIB (A)	Introduction to graph theory and applications.
HFU	Summer 2022	Bildverarbeitung und Computergrafik	MIB (B)	Computer vision course with focus on 3D reconstruction.
HFU	Summer 2022	Mathematik und Simulation Seminar	MIB (A)	Java programming seminar accompanying the second-semester math and simulation course.
HFU	Summer 2022	MINT Praktikum	MIB (A)	Math tutorial course.
HFU	Summer 2022	VISMO Projekt	MIB, MKB (C)	Continuation project focused on implementation with THREE.js.

Table 1: Teaching Courses (continued)

Affiliation	Term	Title	Program (Category)	Description
HFU	Winter 2021	Grafische Datenverarbeitung	MIB (B)	Introduction to computer vision from image formation and features to deep learning.
HFU	Winter 2021	Echtzeit-Computergrafik	MIB (B)	Scrum-based real-time graphics course with WebGL/WebXR development in agile teams.
HFU	Winter 2021	MINT Praktikum	MIB (A)	Math tutorial course.
HFU	Winter 2021	VISMO Projekt	MIB, MKB (C)	Student project on visualizing structure-from-motion foundations in a 3D engine.
TU Berlin	Summer 2011	MPGI 2	INB (A)	Methodic and practical basics in computer science (part 2); tutorial organization and selected graph-theory lectures.
TU Berlin	Summer 2011	Processing Digital Camera Images	INB (C)	Seminar on modern camera image processing; support in organization and student talks.
TU Berlin	Winter 2010	Computer Graphics 1	INB (B)	Basic course along the rendering pipeline, including exercise support.
TU Berlin	Winter 2009	Processing Digital Camera Images	INB (C)	Seminar on modern camera image processing with support in organization and delivery.
TU Berlin	Summer 2009	Computational Photography	INB (C)	Project course in conjunction with the University of Arts (UdK) Berlin.
TU Berlin	Summer 2009	MPGI 2	INB (A)	Methodic and practical basics in computer science (part 2); led tutorial team.
TU Berlin	Winter 2008	Depth Imaging	INB (C)	Project course on combining time-of-flight depth imaging and color imaging.
TU Berlin	Summer 2008	MPGI 2	INB (A)	Methodic and practical basics in computer science (part 2); led tutorial team.
TU Berlin	Winter 2007	Multi-Touch Display	INB (C)	Student project co-organized with Marc Alexa.
TU Berlin	Winter 2007	MPGI 1	INB (A)	Methodic and practical basics in computer science; tutorial organization and execution.
TU Berlin	Summer 2007	Computer Graphics/Computer Vision	INB (B)	Basic course in CG and CV; exercises co-organized, including student panorama tool work.
TU Berlin	Winter 2006	Scanning real world objects without worries	INB (C)	Student project co-organized with Marc Alexa.

Appendix B Teaching Evaluation Results

In the following Table 2, selected student evaluation results are presented, highlighting feedback on course content, teaching methods, and overall satisfaction.

Table 2: Selected evaluation results

Course abbr	Program	Year	Evaluation comment	Overall results
CVDL	MIB	2026	'I consider this course to be one of the best lectures I have had the privilege of attending. For me personally, the combination of lecture slides, tutorials and a follow-up project is an excellent way to gain a thorough understanding of the subject matter. The workload is heavy, but given the quality of the course, I feel this is justified to some extent.'	1.57, (very good)
3DCV	MIM	2026	'The professor's enthusiasm for the subject matter of the course is evident in every lecture, and this has a very positive effect on one's own motivation to learn. Extensive queries regarding the slides were always addressed in detail and covered for everyone in the following lecture.'	1.56, (very good)
Entry project	GMB	2026	'Making a game is fun, The instructors where great, and I am excited for their instruction in the future. The helpers (student tutors) where great when they were available.'	'The instructor is engaged and interested?' 4.53/5 agreement.
CODE3	GMB	2026	'I enjoyed the parts where we had to solve a coding assignment based only on the theoretical explanation of what we were trying to do. The overview of the graphics pipeline was interesting and understandable.'	'The instructor is engaged and interested?' 4.93/5 agreement.
STEM 2	GMB	2026	'I think it makes a lot of sense that you're not completely on your own when programming, but rather that you work with the professor to develop the basic program. The programming assignments were also, for the most part, easy to solve and helpful for understanding the code.'	1.67, (very good)
GDV	MIB	2025	'I think the visualizations are very good, as are the explanatory videos, since you can watch them again at home to better understand the topic.'	1.75, (better than good)
DAVT	MIB	2025	'I really appreciate the casual and relaxed atmosphere in class, as well as the overall well-structured format of the lectures. The course materials are also very well organized on the dedicated website, which makes preparation, follow-up work, and studying much easier.'	1.67, (very good)
GIS	OMB, MIB	2024	'The script for this module is probably the best way to understand the subject matter and try things out for yourself. The atmosphere is always relaxed and friendly, and there's a lot to take away from the lectures. Above all, Mr. Hahne does an exceptional job teaching this course and has sparked a deep interest in programming within me.'	1.0, (very good)
Mathe	MKB	2024	'He really goes out of his way to teach us things. You can tell it's important to him that we understand. Sorry, bro, we're just not in the mood for math :('	3.29, (acceptable)
STEM1	GMB	2024	'You are very patient with us, even though we are often a bit unresponsive. You constantly try to understand what we find difficult.'	2.07, (good)
GDV	MIB	2023/24	'The atmosphere was very pleasant, and the exercises were designed to be interesting.'	1.67, (very good)

Appendix C Supervised Theses

In the following Table 3, a comprehensive list of supervised theses is provided, including details such as the year of supervision, author, title, affiliation, and type.

Table 3: List of supervised theses

Year	Author	Title	Affiliation	Program
Ongoing	Hepting, Marcel	KI-gestützte Produktbildgenerierung für den Einsatz im E-Commerce	HFU + Flowers&me	OMB
Ongoing	Fröhle, Pierre	Der Einfluss von Sounddesign in Horrorfilmen - Eine mathematische Untersuchung zur Erzeugung von Angst am Beispiel von der Exorzist.	HFU	MIB
2026	Dropic, Amra	Mathe erlebbar machen	HFU	MKB
2026	Hartmann, Franziska	Evaluating geometric foundation models in real-world applications	HFU	MIB
2026	Vogel, Valeria	Monokulare Tiefenschätzung bei Zeichnungen: Potenziale und Grenzen am Beispiel von DepthAnything	HFU	MIB
2026	Weber, Adrian	Digitale Barrierefreiheit messbar machen: Erstellung und Evaluation einer Prüfmethodik zur Sicherstellung der Barrierefreiheit in Softwareanwendungen	HFU + M&M Software GmbH	OMB
2026	Richter, Fabian	Die Untersuchung der Anwendbarkeit neuer, KI-gestützter DataMining-Ansätze für Patente und Patentstrategien im Bereich ausgewählter Schlüsseltechnologien der Elektrifizierung und Automatisierung im Automotivebereich.	HFU + DLR GmbH	MIB
2025	Hergert, Lea	Entwicklung eines KI-Chatbots zur spielerischen Sensibilisierung von Kindern für Online-Grooming	HFU + Inspire Technologies GmbH	MIB
2025	Specht, Enrique	Konzeption und prototypische Umsetzung für End-to-End-Tests in natürlicher Sprache	HFU + doubleSlash Net-Business GmbH	MIB
2025	Cleppien, Marie	Evaluating Multi-Agent Systems Writing for Creative Works Together with Humans	HFU	MIB
2025	Riedinger, Niklas	Einfluss der initialen Punktwolke auf die Optimierung für 3D-Gaussian-Splatting	HFU	MIB
2025	Göpfert, Ferdinand	User Focused Fire Evacuation Simulation for HoloLens	HFU, Kanagawa University (Japan)	DIM
2025	Brausch, Lukas	Panic behavior generation under emerged unexpected events with interactive avatars	HFU, Kanagawa University (Japan)	DIM
2025	Willmann, Lukas	Object recognition in radiance fields: A comparison of language embedding methods	HFU	MIM
2025	Fuchs, Ethan	Capturing a campus: A large-scale 3D Gaussian Splatting case study	HFU	MIB
2025	Zenker, Marit	Collision-Free Motion Planning in 3D Environments - A 3D Gaussian Splatting Approach	HFU	MIB
2025	Müller, Melanie	Maßstabsgetreues Projection Mapping von Grundrissen	HFU	MIB
2025	Kaserer, Patrick	Synthesized Sensor Data from Neural Radiance Fields	HFU + SICK AG, Waldkirch	MIM
2025	Sasanala, Aravind Sal	Algorithm Development for Feature Detection in 3D Polygon Mesh Models of Implants	HFU + stryker Leibinger GmbH & Co KG, Freiburg	Master
2024	Chemata, Rebecca	Generative KI zur internen Kommunikation von Unternehmensvisionen und -strategien	HFU + SICK AG, Reute	OMB
2024	Pönitzsch, Felix	ViperGPT: Automatisierte Softwareentwicklung mit Large Language und Computer Vision Models	HFU	MIB

Year	Author	Title	Affiliation	Program
2024	Nguyen, Baohan	Wasserspeier am Freiburger Münster: Entwicklung einer webbasierten Augmented Reality Anwendung zur Erhaltung des Kulturerbes und interaktiven Erkundung	HFU + Münster-bauverein Freiburg	MIB
2024	Widmann, Dominik	Implementation and evaluation of Neural Radiance Fields as an immersive technology in industry at technology start-up Dromni: An application study	HFU + DROMNI GmbH, München	OMB
2024	Punt, Jochem	Entwicklung eines AR-Produktkonfigurators und die Analyse des KI Nutzwerts	HFU + CAS Software AG, Karlsruhe	MIB
2024	Fischer, Marvin	Automatically switching between portrait and landscape formats	HFU	MIB
2024	Rubner, Nic	Design eines Software-Frameworks zur Integration von Azure Kinect-Daten in AR-Applikationen	HFU	MIB
2024	Junghans, Vincent	Remove the Dog: Textbasierte Bildbearbeitung	HFU	MIB
2023	Paredes Gonzalez, Moreli Andrea	Comparison of 3D Capture Techniques, with Application in the Design of a WebAR Catalog of the Gargoyles of the Freiburg Cathedral	HFU + Münster-bauverein Freiburg	DIM
2023	Velu, Balaji	Improvements in precision alignment of 3D digital counterparts in Mixed Reality	HFU + BMW group, München	Master
2023	Schleise, Sabine	Neural Radiance Fields in the context of the Industrial Metaverse	HFU + SICK AG, Waldkirch	MIM
2023	Reiche, Yannick	Erarbeitung eines On-Set Virtual Production Setups mit Projektoren für Low-Budget Filmproduktionen	HFU	MIB
2023	Reiprich, Alexander	Computer Vision als Werkzeug zur Steuerung von Videospielen - Analyse und Entwicklung eines Programms basierend auf Reinforcement Learning	HFU	MIB
2023	Kowatsch, Fabian	Implementing a real-time 3D viewer for Neural Radiance Fields	HFU	MIB
2023	Brausch, Lukas	KinectAR: Implementierung einer Live-Datenübertragung einer Kinect Azure-Punktwolke auf ein AR-fähiges Mobilgerät	HFU	MIB
2023	Pfeiffer, Sebastian	Creation and evaluation of a computer game using a teachable AI	HFU	MIB
2023	Münchinger, Maximilian	Die Zukunft des Fotografierens: Wie verändert künstliche Intelligenz die Fotografie?	HFU	MIB
2022	Kusch, Oliver	Entwicklung einer Augmented Reality App zur Visualisierung historischer Gebäude der Stadt Villingen	HFU	MIM
2022	Küllmer, Elisabeth	Entwicklung und Evaluation einer Deep Learning Beispielapplikation auf programmierbaren 3D-Kameras	HFU	MIB
2019	Fischer, Georg	Detection and Mitigation of Multipath Interference in 3D-Time-of-Flight Cameras	SICK AG + KIT (Karlsruhe)	Master
2019	Roth, Christoph	Ressourceneffiziente Personendetektion auf Basis von 3D Kameradaten für den Einsatz auf eingebetteten Systemen	SICK AG + KIT (Karlsruhe)	Master
2012	Miethe, Manuel	Integrating a Depth Sensor into an Interactive Tabletop	TU Berlin	Diploma
2010	Dippl, Stefan	Markerless Identification of Interactive Devices on Multi Touch Surfaces	TU Berlin	Diploma
2010	Stripf, Claudia	A Multi-Touch User Interface for Spatial Audio Scenes	TU Berlin	Term paper
2010	Chmiela, Stefan	Identifying Untagged Mobile Devices on Tabletops by Matching Camera and Acceleration Sensor Data	TU Berlin	Diploma
2010	Mircea Gigea, Raul	Augmentation of Real-Time Registered Surfaces using a Projector-3D Camera System	TU Berlin	Diploma
2010	Müllenhaupt, Martin	Evaluating time-of-flight sensors using an industrial robot	TU Berlin	Diploma

Year	Author	Title	Affiliation	Program
2009	Breitmeyer, Björn	Object recognition in a multitouch environment by combining Diffuse Illumination and Frustrated Total Internal Reflection	TU Berlin	Diploma
2009	Holzammer, Andreas	Combining Diffuse Illumination and Frustrated Total Internal Reflection for touch detection	TU Berlin	Diploma
2009	Elstner, Stefan	Combining Pen and Multi-Touch Displays for Focus+Context Interaction	TU Berlin	Diploma
2007	Mischler, Antoine	3D Reconstruction from Depth and Stereo Images for Augmented Reality Applications	TU Berlin	Diploma