ParamCurve - Immersive Learning for Parametric Curves

Abstract: In higher education mathematical curves are often taught using traditional materials like paper, boards and slides. While these classes are often supported by digital teaching concepts, they are often limited to passive visualizations like videos or images. A new, more interactive approach is made possible through virtual reality. This medium offers many opportunities for new learning techniques that promote understanding beyond mathematical formulas and increase spatial imagining. In this project, we use the analog and digital assets of an existing mathematics class to develop an immersive learning application for parametric curves.

1 Introduction

The learning process of mathematical concepts in university classes is often a hard and dry process based on symbols, formulas and abstraction. While a subset of students will always struggle with this approach, the underlying knowledge remains important for applications in scientific and industrial contexts. Very often the facilitation of insights and deeper understanding is supported by non-interactive visualizations like images and animations. Early in the 1990s virtual reality was already being denoted as a paradigm shift in the educational space [WB92]. In the following years and decades, concrete learning effects have been associated with effective and targeted use of virtual reality [MN11, MGC+14]. For many years immersive learning applications for schools and universities have been actively developed and used [LGA+21]. Learning environments profit from the immersion of the user. This is due to the ability to construct environments that allow multiple levels of cognitive sensors through real-time and a natural interaction.

A first theoretical model of important factors that contribute to the development of successful immersive learning applications was presented by Salzman [SDLC99]. Further elaboration was created by Lee et al. [LWF10]. These factors create a collection of metrics that aid the development and evaluation of immersive learning applications. Virtual reality aims to obfuscate the context of the outer world. The user exists in a virtual environment that allows for different kinds of interaction [RMC+15].

In order to incorporate such interactive experiences in a class, educational concepts are needed. One such concept is microlearning, an emerging sub-category of learning that divides complex knowledge structures into small, single units of knowledge called nuggets [HD19]. This allows educators to pick different approaches for each nugget, i.e. using a virtual reality

application to communicate the knowledge of a single nugget if the content of the nuggets profits from such an interactive environment. Targeting only singular nuggets is vastly preferable when compared to attempts of forcing whole curricula into a single technology.

2 An immersive learning application for a mathematics class

Harnessing the potential of this type of immersive experiences for the purpose of learning requires applications that correctly map learning goals of individual classes and make them accessible to the target audience. In this poster we present *ParamCurve* our prototype of a virtual reality application that aims to visualize parametric curves to facilitate insights and deeper understanding of the concepts presented in a mathematics class in a computer science program. Gray [ASG06] uses Mathematica, we decided to use Python as programming language. The existing set of learning materials based on LaTeX, Python and Markdown is the basis for *ParamCurve*.

The Python module contains a lot of 2D and 3D curves and algorithms like arc length computation or arc length parametrization. In the exercises, students actively change this code basis. They not only code in Python but also build new and interactive Jupyter Notebooks. We expanded this Python module to be able to export the computed data in an JSON file containing curve and algorithmic data. This way we can map changes made by us or the students immediately in our immersive learning application. We also export system configurations that allow for scene configuration in the immersive application. Upon starting our virtual reality application, the imported curves are rendered in the virtual environment. They can be viewed in different ways and allow for simple interaction. The Python code and the source for *ParamCurve* is publicly available as a GitHub repository.

Students have access to this repository and the immersive learning application. We use the virtual reality application in the classroom with a set of all-in-one Head-Mounted Displays and in our lab on HTC Vive Pro. We also provide releases of the application for different platforms like Google Cardboard, HTC Focus Plus, Oculus Quest or OpenVR-based HMDs, so students can use the exported data for study at home, preparing exercises or the exam.

3 ParamCurve

We implemented *ParamCurve* using the Unity Game Engine and the Vive Input Utility [Viv22]. With VIU we can support a wide range of hardware settings in our lab, the classroom and for homework. Like other immersive learning applications built in our lab we decided to use a virtual seminar room to ease the transition between the real and virtual environment. Figure 1 shows an isometric view of this room. The scale is derived from a small classroom on our campus.

Based on values in the initialization file, elements of the room are dis-/enabled. We use this configuration to present examples from the class and exercises, where students have to
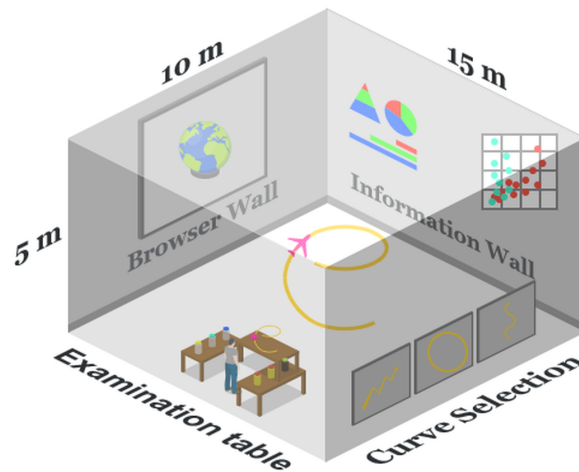
Figure 1:   Isometric view of the virtual room

examine given curves or a set a curves and their parameter functions. Curve data is imported as a single curve or a set of curves that can be navigated through to select individual curves for visualization. A single curve is rendered once in the middle of the room, and a second time in smaller scale on an examination table. The display view simply visualizes the curve in the virtual environment. This examination table shown in figure 2 resembles the visualization in a Jupyter notebook or on paper.
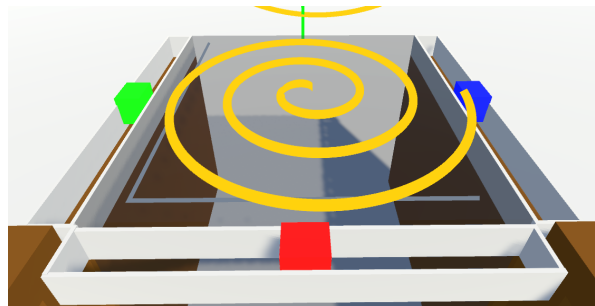

Figure 2:

The run view in figure 3 adds small airplane models to the curve that can be triggered to fly along the curve, moving and rotating based on Frenet frames. This view is used in class to highlight the concepts of velocity, accelaration, arc length and local coordinate systems on a parametric curve. The exported data always contains an arc length parametrization of the visualized curves. Once a run is triggered, both planes move along the curve, differing in speed and orientation due to their parametrization. This way students get a hands-on experience of mathematical concepts like arc length parametrization or ease-in-ease-out animation.

The walls of the room allow for further interaction. The browser wall displays Jupyter notebooks or other webpages containing additional information related to the current curve
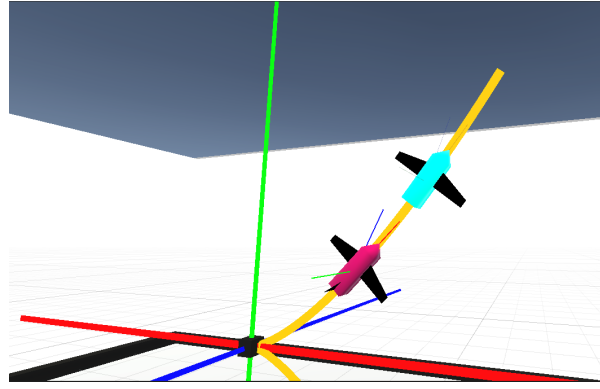
Figure 3:   Different parametrizations in the run view

or exercise. On the information wall we render geometric data values or plots that update during flights in the run view, again visualizing velocity, arc length and other curve attributes. A curve selection wall offers the opportunity to switch between specific curves directly.

4   Future work

The current prototype version is the result of a master thesis. More functionality will be added to the examination table, allowing for even more granular inspection of the visualizations. We are working on a different scene providing a first-person experience for the run view. Users will be able to switch from the room view to first-person and back with a set of scene transition techniques.

First small experiments in the 2021 winter term already showed the potential of our approach. Once restrictions related to the global pandemic have been lifted, we will do a thorough evaluation to improve the application and the learning arrangement. This evaluation will be based on the given immersive learning factors and hopefully evaluating a "joy of use" effect, leading to increased student engagement with the mathematical concepts.

This prototype is part of a three-part set of immersive learning applications related to different classes currently being taught on our campus. This includes an existing application related to ray tracing used in a computer graphics class and a application for the immersive visualization of scalar and vector fields, again in the mathematics class.

References

[ASG06]   E. Abbena, S. Salamon, and A. Gray. *Modern Differential Geometry of Curves and Surfaces with Mathematica, Third Edition*. Textbooks in Mathematics. Taylor & Francis, 2006.

[HD19]   Robin Horst and Ralf Dörner. Mining virtual reality nuggets: A pattern-based approach for creating virtual reality content based on microlearning methodology. 12 2019.

[LGA+21]   Lee, Mark, Georgieva, Maya, Alexander, Bryan, Craig, Emory, and Richter, Jonathan. State of xr and immersive learning 2021. Technical report, Immersive Learning Research Network, 2021.

[LWF10]   Elinda Lee, Kok Wong, and Chun Fung. How does desktop virtual reality enhance learning outcomes? a structural equation modeling approach. *Computers and Education*, 55:1424–1442, 12 2010.

[MGC+14]   Zahira Merchant, Ernest Goetz, Lauren Cifuentes, Wendy Keeney-Kennicutt, and Trina Davis. Effectiveness of virtual reality-based instruction on students learning outcomes in k-12 and higher education: A meta-analysis. *Computers and Education*, 70:29–40, 01 2014.

[MN11]   Tassos A. Mikropoulos and Antonis Natsis. Educational virtual environments: A ten-year review of empirical research (1999-2009). *Comput. Educ.*, 56(3):769–780, apr 2011.

[RMC+15]   Shuo Ren, Frederic Mckenzie, Sushil Chaturvedi, Ramamurthy Prabhakaran, Jaewan Yoon, Petros Katsioloudis, and Hector Garcia. Design and comparison of immersive interactive learning and instructional techniques for 3d virtual laboratories. *Presence: Teleoperators and Virtual Environments*, 24:93–112, 05 2015.

[SDLC99]   Marilyn C. Salzman, Chris Dede, R. Bowen Loftin, and Jim Chen. A Model for Understanding How Virtual Reality Aids Complex Conceptual Learning. *Presence: Teleoperators and Virtual Environments*, 8(3):293–316, 06 1999.

[Viv22]   ViveSoftware@htc.com. Vive input utility, 2022.

[WB92]   William Winn and William M. Bricken. Designing virtual worlds for use in mathematics education: the example of experimental algebra. *Educational Technology archive*, 32:12–19, 1992.