3D Visualization Methods
in Online Control Experimentation

Jakub Matišák and **Katarína Žáková**

Institute of Automotive Mechatronics
Faculty of Electrical Engineering and Information Technology
Slovak University of Technology in Bratislava
Ilkovičova 3, 812 19, Bratislava, Slovakia
(e-mail: jakub.matisak@stuba.sk, katarina.zakova@stuba.sk)

Abstract: Classical teaching of control theory that is based on mathematical models can sometimes be quite difficult for students. Not everybody is able to understand physical meaning of differential equations immediately. The use of various visualizations can facilitate the process of learning. The paper describes the system that presents animations of mechatronic systems to students. They can be run in two modes – in open loop to follow the dynamical behavior of the system without the use of any controller and in closed loop that considers the basic PID controller. The aim is to show students the effect of the individual components of the controller, i.e. the effect of P, I and D part. The visualization is realized in two forms: first, we use 3D animation that is available as the web application and then we decided to use increasingly popular augmented reality. This kind of simulation is available via smartphones and it is based on Google Services for AR platform, built for augmented reality experience. Both types of visualizations will be illustrated on the model of towercopter that we have in our laboratory also as a real plant. Using the presented educational environment allows students to achieve first experience with the system, before they want to use a real device in laboratory.

Keywords: control theory, PID, visualizations, towercopter, augmented reality, Scilab

1. INTRODUCTION

Purpose of universities is to educate people, to improve and to push boundaries of world understanding further. Also, institutions and companies need to educate their employees. Goal of each technical university must be to prepare future engineers for real life problems, how to deal with them and how to solve them. In higher education it can happen that it is quite difficult to understand meaning of mathematical models. Meaning of the specific subject matter can be hard to imagine and realize. For both, the student and the teacher it is better to explain issues on practical exercises. Useful way how to improve education is to visualize the whole process, before one gets to a real device. There are several reasons why to do that: *to understand the problem better, to save costs, to prevent injuries and to prevent damages.* This idea was also supported by (Pomerantz et al., 2019) and (Yip et al., 2019).

The paper is devoted to the control teaching based on 3D models visualisations in different simulation environments. Purpose of this approach is to bring new perspective to interactive teaching and to simplify the whole process of understanding the subject matter. Applications presented in the paper allow students to achieve first experience

with the device, before they want to use the real plant in laboratory.

2. RELATED WORK

Recently there has been a worldwide increase in various visualizations. It is mostly caused by the greater expansion of game industry and usage of the virtual reality. In education field there can be also found some interesting applications, i.e. (Raju et al., 2018). Model visualizations could be divided in two types:

2D models are used to describe certain types of images, such as technical diagrams, logos, mechanical systems, etc. There also exist tools for creating 2D models whereby they can offer developers additional predefined functionality regarding modeling of dynamical systems (Esquembre, 2004), (Saenz et al., 2015) or (Matišák and Žáková, 2018).

3D models are in some cases better than 2D models, because they are more explicit and understandable. 3D model created as an precise copy of the real plant can also be used as useful visualization in education. Some examples were already published in several conferences and journal papers. The tool referenced in (Esquembre, 2004) and (Saenz et al., 2015) does not only allow creating 2D representations but also 3D ones. Segway model based on real plant was presented in (Žáková and Hók, 2016). Model of electronic kart can be found in (Kučera et al., 2018) or model of planet systems e.g in (Huang et al.,

2019). The visualizations in 3D can also be used to reconstruct the image and to preserve cultural heritage (Park et al., 2014).

Studies confirm that interactive way of teaching supports the concentration. As it can be seen in (Cabero-Almenara et al., 2019) or in (Mayilyan, 2019) this approach helps students to achieve better results in exams. Papers (Álvarez et al., 2014) and (Chamba-Eras and Aguilar, 2017) show that students were satisfied with using augmented reality and were more successful in the subject.

However, the education is not the only field, where the visualization can be used. Companies change and upgrade technological processes during the time, so their employees need to learn new methods to control the devices. Therefore the use of augmented reality in industry is often demanded nowadays (Cardoso et al., 2013), (Paelke, 2014) or (Perla et al., 2016).

Based on presented studies visualizations reasonably have their place in education. Therefore our aim was to create a system for supporting teaching of control theory on 3D models.

3. SYSTEM ARCHITECTURE

We decided to build a new set of applications that enables to control 3D visualisation of real systems and to verify functionality of basic control algorithms (PID, state space controller, etc.).

3.1 Application Requirements

To achieve the maximal impact of the application it needs to fulfil several requirements. It should

- have minimal hardware requirements, possibility of widespread adoption among users and easy access to application,
- make the realistic view of the mechatronic device:
  · using web browser
  · using augmented reality
- accept user-defined parameters,
- enable running of the model
  · in open loop to follow the dynamical behaviour of the system,
  · in closed loop that considers e.g. the basic PID controller.

3.2 Technology Overview

To fulfil the mentioned requirements, the key point is to choose the right technology and development tools for the implementation.

For the web-based modules the PHP programming language and usage of the frameworks was chosen. Choose of the right framework can be crucial for developing, so we have considered few aspects: *performance, design pattern (Model View Controller architecture), database support, community support and documentation support.*

After reviewing possible frameworks, our attention was devoted to two frameworks: *Laravel and Symfony.* Both ones are large frameworks that meet our requirements.

Laravel is nowadays probably the most popular framework for the development of the full-stack applications and it is easy to learn. It comes with the long-time support (LTS) and the support of the community (Laaziri et al., 2019). It is based on Model-View-Controller architecture, which helps in improving the performance, allows better documentation and has multiple built-in functionalities (Shah, 2018).
Symfony is considered the most stable PHP framework and similar to Laravel it is also supported by an extended community and LTS.
Considering advantages and performance of both frameworks we decided not to use the same solution for the whole application. The detailed use of each framework is explained in 3.3.

Since the augmented reality (AR) is not generally supported in web-oriented platforms, the decision was to use the 3D development engine. Game engine is a development software environment for game designers to create games more quickly. These days there exists a lot of game engines. Most of them are commercial, but few can be used free for non-commercial purposes, such as Godot, Unity or Unreal Engine. All the mentioned engines have huge community of users. However, Godot does not provide support for augmented reality, so it was not appropriate for the AR development. Both Unity and Unreal engine are compatible with augmented reality platform and they are cross-platform. Based on comparison (Christopoulou and Xinogalos, 2017) Unity was chosen as engine for developing of our application, because it has better performance on mobile devices, better user experience and more realistic graphics.

Two leading development platforms for creating applications in augmented reality are Apple ARKit and Google ARCore. In our project ARCore was chosen, because Android is the most widespread smartphone operating system. According to data from September 2019 given by StatCounter (a company dealing with Website analytics) Android OS is used by 76.24% smartphone users, iOS by 22.48% and other OS by 1,28% users (StatCounter, 2019).

ARCore platform is characterised by three main principles defined by Google:

- *Motion tracking*
  To observe feature points in the room the camera in the smartphone and IMU (Inertial measurement unit) sensor data are used. ARCore takes both the position and orientation of the phone when it moves meanwhile virtual objects remain placed accurately in the specific position.
- *Environmental understanding*
  To place object on horizontal surfaces (common procedure for AR objects) it makes detection using the same feature points as for motion tracking.
- *Light estimation*
  To make the appearance of the object more realistic, ARCore platform observes the ambient environment light. In this way developers can light the virtual object in a more authentic way, because the light matches more with the surrounding environment (Amadeo, 2017).

Since our aim was not just to make visualisation of some object but also to control its output, some simulation engine is also needed. To facilitate the development the complex simulation environment is running in the background. It was possible to choose commercial Matlab or LabView or free alternatives such as Scilab or OpenModelica. All of them enable to model and simulate behavior of linear and nonlinear systems and the advantage is that all of them have graphical editor for building block schemes. We decided to use open source Scilab (with Xcos graphical editor) because it is very similar to commonly used Matlab (Simulink) and it does not require a license and therefore everyone has access to it.

Finally, let us summarize all technologies for building the final application. The complete development will use

- PHP frameworks - Laravel and Symfony,
- development engine - Unity,
- augmented reality platform - Google ARCore,
- Scilab/Xcos simulation environment.

3.3 System Architecture

The application is using modular design. The whole system is divided into several components. All these modules work as independently created parts and can be used in different systems in the future. The biggest advantage of modular design is that each module can be reprogrammed separately.

In Fig.1 the system architecture of the created modular system is shown. From the user's perspective there are two client applications. The user can interact with both of them. The visualisation of the system is done:

- using web application,
- using Google ARCore as a platform for the smartphone application.

The worldwide use of smartphones allows people to use online applications wherever and whenever it is necessary. The main reason to develop web-based application is still increasing diffusion of Internet connection and the growing number of its users. According to the survey from June 2019 presented in (Miniwatts Marketing Group, 2019) Internet is used by 58,8% of world population. Following this fact, it is good to prepare the online application because it is quite comfortable to use it since it does not require any installation. The user needs just a web browser. On the other hand, the screen is limited since it is flat. For more realistic visualizations it is possible to move to the augmented reality.

Web Application   The back-end side of the web application is developed using Laravel PHP framework. Fig.1 shows Laravel MVC implementation.

All information about the 3D model (such as PID controller parameters, additional experiment parameters, the control schema) are stored in MySQL relational database. It is open source, reliable, compatible with all major hosting providers, cost-effective, and easy to manage (Branson, 2017).

The View is rendered by Laravel framework by its own PHP templating engine called Blade. The JavaScript li-
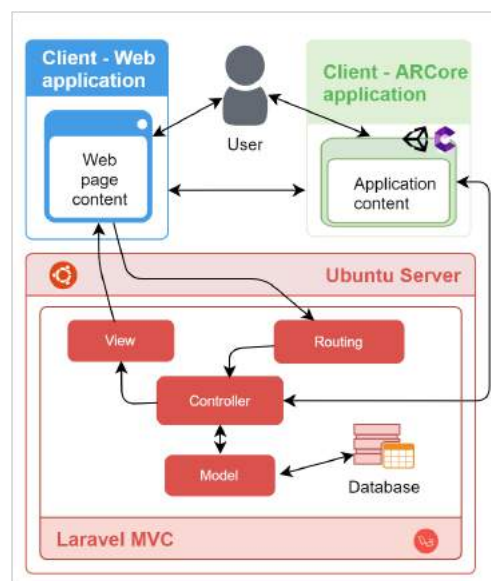

Fig. 1. Architecture of the web application and augmented reality module

brary Babylon.js is used for 3D visualization. It allows to render 3D model on the screen.

AR Application   The smartphone application is created in Unity, which uses plugin for Google ARCore development platform. The application allows user to see the model of the device in real environment. The application uses HTTP requests to Scilab module (Fig.2) and it also needs access to database that store necessary parameters.

Back-end for Client Applications   The Scilab module is running as a web service with its own application programming interface. It can be seen in Fig.2. The benefit of creating module as the API is that then it is accessible via Internet and reusable for further development.
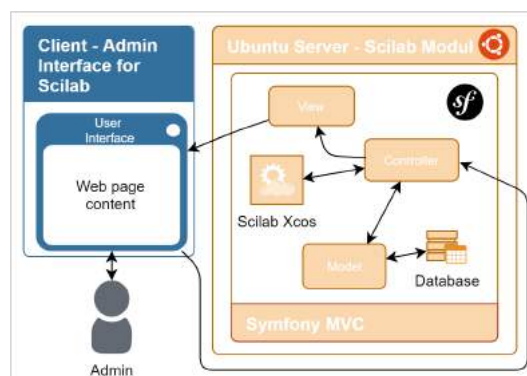

Fig. 2. Architecture of the computing module

The Scilab module was created using Symfony PHP framework. In comparison to Laravel the REST API performance of Symfony is much higher. Based on results in (InfoDroid, 2019) Symfony 4.2 was 71% more efficient than Laravel 5.8 in the accomplished test. The benchmark is represented by a REST API application. More about the test can be found in (Barboyon, 2019).

Since the Scilab module is mostly back-end application, which provides the API it is more efficient to use Symfony

as framework for its development (in difference to Laravel). Laravel is more efficient as framework for full-stack applications. On the other side the web module is mostly connection between front-end, back-end and the database in this case Laravel is a better option (Laaziri et al., 2019).

The architecture of the Scilab module is a little different (Fig.2) compared to architecture of web application (Fig.1). The module serves as the API for the Scilab/Xcos computing environment. The first difference is that the module is designed for more advanced users that have already some expertise in Xcos block schema creation. These users have the role of block scheme administrator (Fig.2). All block schemas (for each system model) are stored in the database.

The second difference consists in the fact that Symfony MVC is not using "routing", but all interactions are handled only by the "controller". The main purpose of the module is to handle and process HTTP requests. Requests contain information about the desired simulated block schema, parameters for the simulation and the API key, which is needed in the authorization process.

Controllers presented in Fig.1 and Fig.2 control the communication between Scilab module and web application module.

4. IMPLEMENTATION

The whole application is demonstrated on the visualisation of the real towercopter plant. The 3D model was created using Blender modeling tool, which is powerful software with open distribution and has big community of users. In Fig.3 the considered real device and created 3D model is shown.

Fig. 3. Real towercopter plant and the 3D model created in Blender rendered by babylon.js

After modelling the system, it was necessary to export it from Blender. The GLB format (Binary form of Graphics Library Transmission Format) is compatible export format for web and DAE format (Digital Asset Exchange) for the Unity.

The exported file with 3D model for web interface is then stored to the server folder and loaded to the web application by babylon.js library. This JavaScript library can work with objects and visualize them in browser.

The Unity uses scene, where the DAE format is imported. Using the combination with ARCore library, the object can be rendered on smartphone screen with real environment.

The Scilab module has Graphical User Interface (GUI) for easier uploading of block schema and defining all its default parameters. Each upload of an experiment is stored in database with its own ID. The module also provides API which allows to communicate with other applications.

In both version of application after selection of the experiment, the request is sent to the server and to the database. The database stores information about the experiment together with default initial conditions. These data are used for automatic loading of 3D model and generation of the form that enables interaction with the user. After starting the simulation, the HTTP request is sent (the GuzzleHttp was used) to Scilab module where it is processed in several steps:

- access validation using Google Client API,
- preparation of the Scilab script, which is based on the input data. It has access to the database with already uploaded block schemas. The block schema with all parameters is then sent to execution.
- script execution using Scilab console,
- parsing the result,
- validation of the result,
- HTTP response as JSON.

When the application gets the response, data from JSON are used for defining the movement of specific parts of the model. In the towercopter there is the moving platform and the propeller. For this purpose Babylon.js library uses *translate* function, and Unity uses the implementation of *Vector3* function.

The animation is completed with the plot that shows the dynamically changed position of the propeller.

5. EXPERIMENT

The application is based on the experimentation with the real towercopter plant (Matišák et al., 2019). It was identified by the system of the second order that can be described by the following transfer function

$$F(s) = \frac{2.6008}{0.4844s^2 + 1.392s + 1} \tag{1}$$

defined in the back-end of the application. The controlled output is the position of the propeller.

The experiment can run in open loop (for identification purposes) and also in closed loop. In this moment students can control the system by the basic PID controller (Fig.4) with the predefined default parameters K_p, K_i and K_d. Students should try to design their own controller with better perfomance. They can come out from the given model or to make their own identification running the experiment in open loop (without knowing the mathematical model in advance).

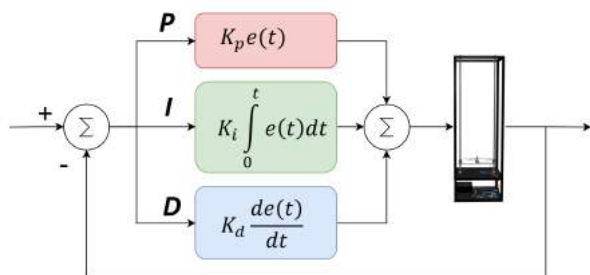GUI of the Scilab module is in Fig.5. It consists of three main parts:
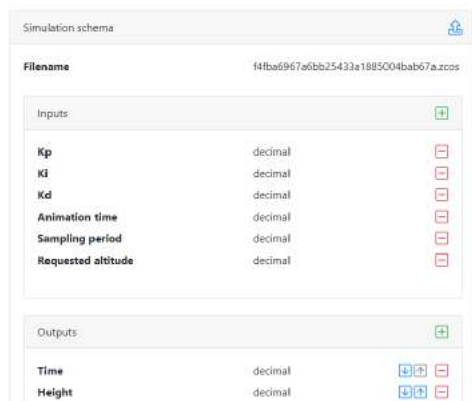
Fig. 4. PID controller schema

Fig. 5. GUI in Scilab module with necessary inputs and outputs

for the simulation

- *Filename*
  It is a file with Scilab/Xcos schema. The file uses zcos format, because it is the newest standard used in Scilab 6.0.2.
- *Inputs*
  Names of parameters are used in the Xcos block schema. K_p, K_i, K_d are individual components of the controller.
  In addition, it is necessary to define the animation time (i.e. length of the simulation), sampling period and the required altitude of the propeller.
- *Outputs*
  Time and height are the output variables. System uses these variables to map output from Scilab console to JSON format.

Visualization of towercopter 3D model in web application is shown in Fig.6 and in AR application is shown in Fig.7.

6. CONCLUSION

The paper describes set of two applications (Web application and AR application) that were developed as new tools for interactive teaching of Control Theory. They can also help students with better understanding of physical meaning of differential equations.

The developed applications illustrate the control of towercopter plant and can be found on *www.iolab.sk* subpages.. They were developed in the modular way because in similar way we plan to illustrate dynamical behaviour and to control other mechatronic experiments, too.
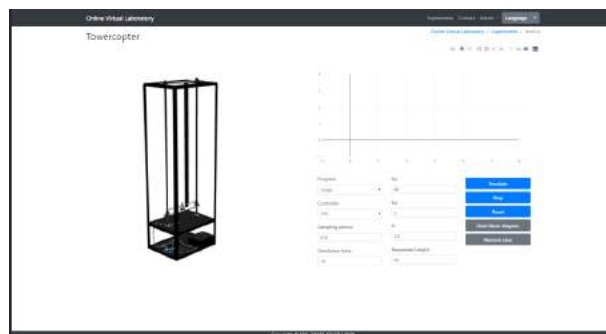

Fig. 6. Web application environment for controlling the towercopter 3D model

Fig. 7. Towercopter experiment – visualization via AR application

As a future work, authors would like to increase the number of predefined control structures not to use just PID controller. The best solution would be to offer users to set up also the own control algorithm. Another plan is to extend the API with more computing environments, such as Matlab/Simulink to create more complex tool.

REFERENCES

Amadeo, R. (2017). Google's ARCore brings augmented reality to millions of Android devices — Ars Technica. URL https://arstechnica.com/gadgets/2017/08/googles-arcore-brings-augmented-reality-to-millions-of-android-devices/.

Barboyon, S. (2019). REST API benchmark - benchmark-kit/restApi.md at master · phpbenchmarks/benchmark-kit · GitHub. URL https://github.com/phpbenchmarks/benchmark-kit/blob/master/documentation/framework/restApi.md.

Branson, T. (2017). The 5 Best Reasons to Choose MySQL - and its 5 Biggest Challenges - Dataconomy. URL https://dataconomy.com/2017/04/5-reasons-challenges-mysql/.

Cabero-Almenara, J., Fernández-Batanero, J.M., and Barroso-Osuna, J. (2019). Adoption of augmented reality technology by university students. *Heliyon*, 5(5), e01597. doi:10.1016/j.heliyon.2019.e01597.

Cardoso, A., Lamounier, E., Lima, G., Oliveira, L., Mattioli, L., Júnior, G., Silva, A., Nogueira, K., do Prado, P., and Newton, J. (2013). Vrcemig: A virtual reality system for real time control of electric substations. In *2013 IEEE Virtual Reality (VR)*, 165–166. doi:10.1109/VR.2013.6549414.

Chamba-Eras, L. and Aguilar, J. (2017). Augmented reality in a smart classroom—case study: Saci. *IEEE Revista Iberoamericana de Tecnologias del Aprendizaje*, 12(4), 165–172. doi:10.1109/RITA.2017.2776419.

Christopoulou, E. and Xinogalos, S. (2017). Overview and comparative analysis of game engines for desktop and mobile devices. *International Journal of Serious Games*, 4(4). doi:10.17083/ijsg.v4i4.194.

Esquembre, F. (2004). Easy java simulations: a software tool to create scientific simulations in java. *Computer Physics Communications*, 156(2), 199 – 204. doi:10.1016/S0010-4655(03)00440-5.

Huang, K., Ball, C., Francis, J., Ratan, R., Boumis, J., and Fordham, J. (2019). Augmented versus virtual reality in education: An exploratory study examining science knowledge retention when using augmented reality/virtual reality mobile applications. *Cyberpsychology, Behavior, and Social Networking*, 22(2), 105–110. doi:10.1089/cyber.2018.0150.

InfoDroid (2019). Comparator. URL http://www.phpbenchmarks.com/en/comparator?phpVersions={\&}groups={\&}components=laravel-5.8{~}symfony-4.2{\&}benchmark=rest-api.

Kučera, E., Haffner, O., and Leskovský, R. (2018). Interactive and virtual/mixed reality applications for mechatronics education developed in unity engine. In *2018 Cybernetics Informatics (K I)*, 1–5. doi:10.1109/CYBERI.2018.8337533.

Laaziri, M., Benmoussa, K., Khoulji, S., Larbi, K., and Yamami, A. (2019). A comparative study of laravel and symfony php frameworks. *International Journal of Electrical and Computer Engineering*, 9, 704–712. doi:10.11591/ijece.v9i1.pp704-712.

Matišák, J., Rosinová, D., Ťapák, P., and Žáková, K. (2019). PID Control of Towercopter System. In *16th IFAC Conference on Programmable Devices and Embedded Systems (PDeS)*, 288–292. Stará Lesná.

Matišák, J. and Žáková, K. (2018). Online tool for creating simple models of mechanical systems. In *2018 16th International Conference on Emerging eLearning Technologies and Applications (ICETA)*, 355–360. doi:10.1109/ICETA.2018.8572249.

Mayilyan, H. (2019). Augmented reality in education, ar globe project assessment in actual teaching-learning environment. *International Journal of Learning, Teaching and Educational Research*, 18, 1–14. doi:10.26803/ijlter.18.3.1.

Miniwatts Marketing Group (2019). Internet Growth Statistics 1995 to 2019 - the Global Village On-line. URL https://www.internetworldstats.com/emarketing.htm.

Paelke, V. (2014). Augmented reality in the smart factory: Supporting workers in an industry 4.0. environment. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*, 1–4. doi:10.1109/ETFA.2014.7005252.

Park, J., Muhammad, T., and Jae-hong, A. (2014). The 3d reconstruction and visualization of seokguram grotto world heritage site. In *2014 International Conference on Virtual Systems Multimedia (VSMM)*, 180–183. doi:10.1109/VSMM.2014.7136646.

Perla, R., Gupta, G., Hebbalaguppe, R., and Hassan, E. (2016). Inspectar: An augmented reality inspection framework for industry. In *2016 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, 355–356. doi:10.1109/ISMAR-Adjunct.2016.0119.

Pomerantz, M., Boettcher, A., Hans, M., Sandoval, M., and Wenzel, S. (2019). Multi-mission, interactive 3d visualization in a web browser for robotic system and space flight mission development and operations. In *2019 IEEE International Conference on Space Mission Challenges for Information Technology (SMC-IT)*, 17–23. doi:10.1109/SMC-IT.2019.00007.

Raju, K.C., Yugandhar, K., Bharathi, D.V.N., and Vegesna, N. (2018). 3d based modern education system using augmented reality. In *2018 IEEE 6th International Conference on MOOCs, Innovation and Technology in Education (MITE)*, 37–42. doi:10.1109/MITE.2018.8747078.

Saenz, J., Esquembre, F., Garcia, F.J., de la Torre, L., and Dormido, S. (2015). An architecture to use easy java-javascript simulations in new devices. *IFAC-PapersOnLine*, 48(29), 129 – 133. doi:10.1016/j.ifacol.2015.11.225. IFAC Workshop on Internet Based Control Education IBCE15.

Shah, J. (2018). Why Laravel is Consider as One Of The Best PHP Framework in 2018? URL https://medium.com/techcompose/why-laravel-is-consider-as-one-of-the-best-php-framework-in-2018-4f40def9c69c.

StatCounter (2019). Mobile Operating System Market Share Worldwide — StatCounter Global Stats. URL https://gs.statcounter.com/os-market-share/mobile/worldwide.

Yip, J., Wong, S.H., Yick, K.L., Chan, K., and Wong, K.H. (2019). Improving quality of teaching and learning in classes by using augmented reality video. *Computers & Education*, 128, 88 – 101. doi:10.1016/j.compedu.2018.09.014.

Álvarez, A., Javier, F., Parra, B., Beatriz, E., Tubio, M., and de Paula, F. (2014). From 2d to 3d: Teaching terrain representation in engineering studies through augmented reality: Comparative versus 3d pdf. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings*, 1–4. doi:10.1109/FIE.2014.7044193.

Žáková, K. and Hók, M. (2016). Interactive three dimensional presentation of segway laboratory model. In *2016 International Conference on Emerging eLearning Technologies and Applications (ICETA)*, 377–380. doi:10.1109/ICETA.2016.7802064.