

Using the Three.js library to develop remote physical laboratory to investigate diffraction

Pavlo I. Chopyk¹, Vasyl P. Oleksiuk^{1,2} and Oleksandr P. Chukhrai¹

¹Ternopil Volodymyr Hnatiuk National Pedagogical University, 2 M. Kryvonosa Str., Ternopil, Ukraine

²Institute for Digitalisation of Education of the NAES of Ukraine, 9 M. Berlynskoho Str., Kyiv, 04060, Ukraine

Abstract

The article examines the process of designing and developing a virtual (remote) physics laboratory. The authors have identified the requirements of the laboratory for studying the diffraction effect (visibility, accessibility, flexibility and visualization). The ability to set additional working parameters such as measurement error, material types and performance conditions. No additional software is required to carry out laboratory work, only an Internet connection and a browser. The research analyzes the available digital tools for developing 3D models and ultimately selects the 3D library Three.js. The researchers then proceed to create a suitable model for representing laboratory objects and processing experimental data. The virtual laboratory enables students to measure the line widths of the diffraction pattern when using monochromatic light (red, blue, green). It also enables the solution of the inverse problem, allowing determination and verification of the wavelength. Additionally, students may calculate measurement errors through the virtual laboratory.

Keywords

remote physical experiments, JavaScript, Three.js, remote lab

1. Introduction

Experiments and laboratory work are an essential part of modern physics education in secondary schools, colleges and universities. They facilitate learners' understanding of physical principles. Laboratory assignments aid students in gaining a better understanding of theoretical concepts by enabling them to observe the operation of physics principles in the natural world. Additionally, practical laboratory sessions allow teachers to assess students' knowledge and understanding. Lab exercises also facilitate student acquisition of skills in data manipulation, processing, and analysis. These skills are valuable not only for the study of physics but also for a wide range of other fields in science and technology. Laboratory experiments promote critical thinking and problem-solving and have the potential to stimulate an interest in science. Furthermore, practical skills are refined through lab work, such as operating scientific apparatus and practical abilities with potential applications in the real world [1].

CS&SE@SW 2023: 6th Workshop for Young Scientists in Computer Science & Software Engineering, February 2, 2024, Kryvyi Rih, Ukraine

✉ chip.ukraine@gmail.com (P. I. Chopyk); oleksyuk@fizmat.tnpu.edu.ua (V. P. Oleksiuk); olexandrtera@gmail.com (O. P. Chukhrai)

🌐 <https://tnpu.edu.ua/faculty/fizmat/chopik.php> (P. I. Chopyk);

<https://tnpu.edu.ua/faculty/fizmat/oleksyuk-vasil-petrovich.php> (V. P. Oleksiuk)

🆔 0000-0001-6631-078X (P. I. Chopyk); 0000-0003-2206-8447 (V. P. Oleksiuk); 0009-0002-7075-3617 (O. P. Chukhrai)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

Currently, remote training is a common practice. Therefore, relevant software has been developed by many vendors and educators [2]. Although virtual laboratory work cannot always substitute traditional laboratory work. It has proven to be essential in certain circumstances [3, 4, 5]. The versatility offered by virtual labs allows students to work from any location, at any time [6]. Moreover, it eliminates the potential risks associated with handling dangerous or costly equipment, which could pose a threat to the safety and well-being of students. This facilitates the verification of results and the utilisation of modern technology. It is feasible to replicate experiments and substantiate findings within virtual laboratories. Researchers can experiment with concepts that are impractical or unfeasible within a conventional laboratory setting, such as exploring nuclear fission or the Big Bang theory.

The absence of practical experience and restricted engagement of students are drawbacks of virtual laboratories. Some concepts may have limited comprehension due to a lack of interaction with physical equipment. Furthermore, students may not adequately develop skills working with equipment. Therefore, it is advisable to incorporate a combined approach, combining both practical and virtual modelling. Virtual models aid students' readiness for laboratory experiments by facilitating comprehension and application of the material.

2. The model of a physical remote laboratory

To set up a remote physics laboratory, an educator can make use of readily available software or create their own. There are some free virtual laboratories such as [7, 8, 9, 10, 11, 12]

- PhET Interactive Simulations from the University of Colorado;
- Physics at School – HTML5-virtual environment ;
- Labster platform;
- CK-12 Interactive Physics for High School;
- Gizmos from Explore Learning.

All of them were created using HTML5, JavaScript, or Java technologies. Significant disadvantages that prevent the use of existing laboratories are the lack of monitoring of students' laboratory work, the inability to conduct measurements without displaying the finished results (in fact, simulations of physical processes are performed), and the perfection of the results, which does not allow for the assessment of measurement errors.

We decided to develop our service for running virtual physics labs. This study aims to systematise the design and development process of creating a remote physics laboratory for diffraction investigation.

Based on the analysis of studies by Holly et al. [13], Manabe et al. [14], Shepiliev et al. [15], Kiv et al. [16], Kanivets et al. [17] we have identified the following requirements for the service

- *Visibility*. The physical process should be fully visualised with the possibility of interactive control.
- *Accessibility*. No additional software is required to carry out laboratory work, only an Internet connection and a browser.
- *Flexibility*. The ability to set additional working parameters such as measurement error, material types and performance conditions.

- *Visualisation.* The appearance of virtual equipment should be similar to that of real laboratory equipment.

2.1. The frameworks, engines and libraries to developing physical remote laboratory

To establish remote laboratories, it is essential to carefully choose adequate development tools, which can include frameworks, engines, and 3D libraries. When considering the comparison of 3D graphics libraries, it is crucial to establish unambiguous standards to evaluate their aptitude and efficacy. The following are essential criteria to consider when evaluating 3D graphics libraries:

1. *Performance and rendering quality.* This criteria assesses the frame rate and rendering speed of intricate scenes, and the quality of graphics rendered, comprising lighting, shadows, and textures.
2. *Ease of use and learning curve.* The criteria are an assessment of the accessibility of tutorials, documentation, community support and ease of use of the library. It also includes the intuitiveness of the API and overall developer-friendliness.
3. *Flexibility and customization.* The assessment of this criterion involves determining the levels of ability to customize the rendering pipeline, effects and support for various 3D models, file formats, and asset management.
4. *Extensibility and modularity for integrating with other technologies.*
5. *Feature set and capabilities* are evaluated through the availability of essential 3D features and support for physics simulation, collisions, and particle systems. It also includes VR/AR support, if needed for the project.
6. *Community and ecosystem.* This criteria assesses the size, activity of the developer community, availability of third-party plugins, extensions, or add-ons, and frequency of updates and ongoing development.
7. *Cross-platform compatibility* provides for the definition the levels of support for various web browsers or platforms (desktop, mobile, VR devices), and the performance consistency across different devices and browsers.
8. *Integration and interoperability.* The criteria are an assessment of the compatibility with other popular frameworks, libraries, or technologies and ease of integration with existing workflows, tools, IDEs.
9. *Real-time collaboration and deployment* are evaluated through support for real-time collaboration among team members and smooth deployment and hosting options for web-based 3D applications.
10. *Project size and loading speed.* The criterion defines the impact of the library on the overall project size. It also includes booting time and efficiency in loading assets and scenes.
11. *License and open-source nature* is important in terms of material costs for project development. It involves assessing licensing terms (open-source, commercial, etc.), associated costs, and availability of a free or trial version for evaluation.
12. *Use cases and education adoption* This criteria assesses the relevance of the library to physics education tasks and examples of successful projects or applications built using

the library. The criterion is interdisciplinary. To assess it, there is a demand for training STEM education specialists who are competent in physics and able to develop and deploy virtual and augmented reality software [18].

There are some well-known libraries and frameworks that can be used to create 3D graphics and animations. These include Three.js, Babylon.js, A-Frame and Cannon.js. Based on the literature reviews by [19, 14, 20] and discussions with practising developers we have conducted a comparative analysis of these 3D graphics libraries. A 5-point scale was used to evaluate the criteria (5 points – maximum compliance of the framework with the criterion, 1 point – minimum compliance of the framework with the criterion). The results of analysis are given in table 1.

Table 1

Compare the capabilities of some libraries and frameworks that can be used to create 3D graphics and animation.

Criteria	Framework			
	Three.js	Babylon.js	A-Frame	Cannon.js
performance and rendering quality	4	5	3	2
ease of use and learning curve	4	4	5	4
flexibility and customization	5	4	3	2
feature set and capabilities	5	5	3	2
community and ecosystem	5	5	4	3
cross-platform compatibility	5	4	4	4
integration and interoperability	4	3	3	4
real-time collaboration and deployment	4	4	3	1
project size and loading speed	4	3	4	4
license and open-source nature	5	5	5	5
use cases and education adoption	4	3	5	3

So, we chose the Three.js library to develop a virtual physical laboratory to investigate the diffraction effect.

The process of designing our virtual laboratory to investigate diffraction predicted took place during such stages.

1. Formulation of the physical problem.
2. Selecting development tools and obtaining the required libraries.
3. The creation and design of a laboratory model.
4. Implementation and testing.

The diffraction grating's constant, d , the distance from the grating to the screen, L , and the diffraction order, m , are provided. Students are to measure the distance from the centre to the maximum l . From these two distances, they have to determine the angle β . Since the β angle is small, $tg(\beta) \approx sin(\beta) \approx \beta$ (figure 1.) We assume that a plane light wave is normally incident on the grating.

There are several steps in the process of conducting an experiment, whether in reality or virtually.

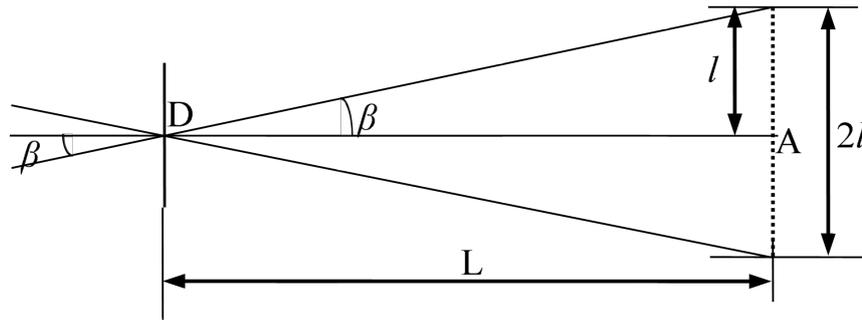


Figure 1: The physics problem of diffraction.

1. The diffraction spectrum on the screen should be captured accurately. Finally, the spectrum should be analysed.
2. Position the diffraction grating at a specific distance, L , from the screen.
3. Calculate the distance l , which represents the distance between the center of the line in the given order's spectrum and the zero spectrum, as illustrated in figure 1.
4. Employ formula (1) to calculate the wavelength λ .

$$\lambda = \frac{d \cdot l}{n \cdot L} \quad (1)$$

where d is the distance between the adjacent slits (grating period), n is the diffraction order (an integer).

5. Alter the distance between the screen and the grating repeat step 3.

2.2. Discussing the design and development of a remote laboratory

As mentioned above, we chose the Three.js library. Additionally, we employed Bootstrap, an open-source CSS framework designed for front-end web development. Therefore we have obtained the necessary libraries, such as three.js-master and bootstrap-dist and uploaded them to the web-server folder. The virtual laboratory model is shown in figure 2.

To actually be able to display diffraction with three.js, we need to create scene, camera and renderer. All objects in our virtual lab are placed in a `<div>` container on the canvas. The left menu contains the parameters of the laboratory objects such as switching on the lamp, activating light filters, moving the dynamic screen, changing the camera position, diffraction grating step, moving the pointer position (see figure 3).

There are a few different cameras in three.js. We used a PerspectiveCamera. It is designed to mimic the way the human eye sees [21]. Based on this mode, 3 camera instances have been created. They allow the student to see the phenomenon in three modes

1. *General view*. It shows a view of the equipment for studying diffraction in three-dimensional space.

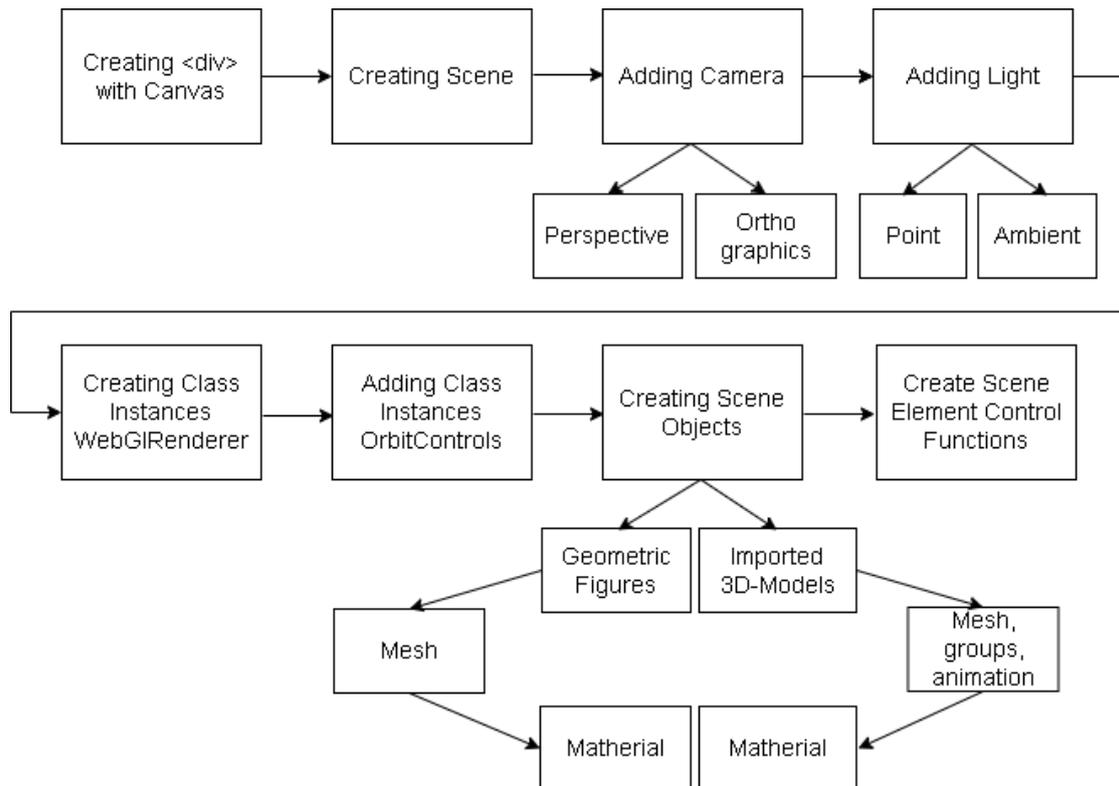


Figure 2: The model of virtual laboratory to investigate diffraction

2. *Observation mode.* It shows the diffraction pattern. This mode allows you to set the pointer.
3. *Distance measurement mode.* It provides rulers for measuring distance l from the formula 1.

The code 1 was used for the creation of a camera.

Listing 1: Camera creation example

```

camera=new THREE.PerspectiveCamera(45, webgl_width/webgl_height,
1, 10000);
camera.position.set(cameraPos1.px, cameraPos1.py, cameraPos1.pz);
camera.rotation.set(cameraPos1.rx, cameraPos1.ry, cameraPos1.rz);
camera.name = "Camera1";
scene.add(camera);

```

To create the lighting, we used 3D objects such as PointLight and AmbientLight. The first one is emitted from a single point in all directions. The second globally illuminates all objects in the scene equally [21]. PointLight is used for the lamp, and AmbientLight is the background lighting for the entire scene.

Then an instance of the WebGLRenderer class was created with parameters such as



Figure 3: The menu of virtual laboratory to investigate diffraction.

- alpha controls the default opacity value. Its value was set to zero;
- antialias is used to reset the internal WebGL state;
- domElement is a canvas where the renderer draws its output;
- setSize sets dimension of the scene;

The figure 3 displays the menu for the virtual laboratory designed for investigating diffraction.

In order to enable camera movement around a grating, an instance of the OrbitControls class was instantiated. The properties of this instance determine the maximum and minimum distance of the dynamic screen and the angle's limit of vertical rotation (see code 2).

Listing 2: An instance of the OrbitControls class creation example

```
controls = new OrbitControls(camera, renderer.domElement);
controls.minDistance = 2;
controls.maxDistance = 25;
controls.maxPolarAngle = Math.PI / 2;
controls.target.set(0, 4, 0);
```

The scene objects were created by using geometric shapes and importing 3D models. BoxGeometry was used to create the main objects in the scene. These include the table, ruler, dynamic screen, stand and diffraction grating. It is a geometry class for a rectangular box with a given width, height and depth. The view of these objects from camera 1 is shown in figure 4.

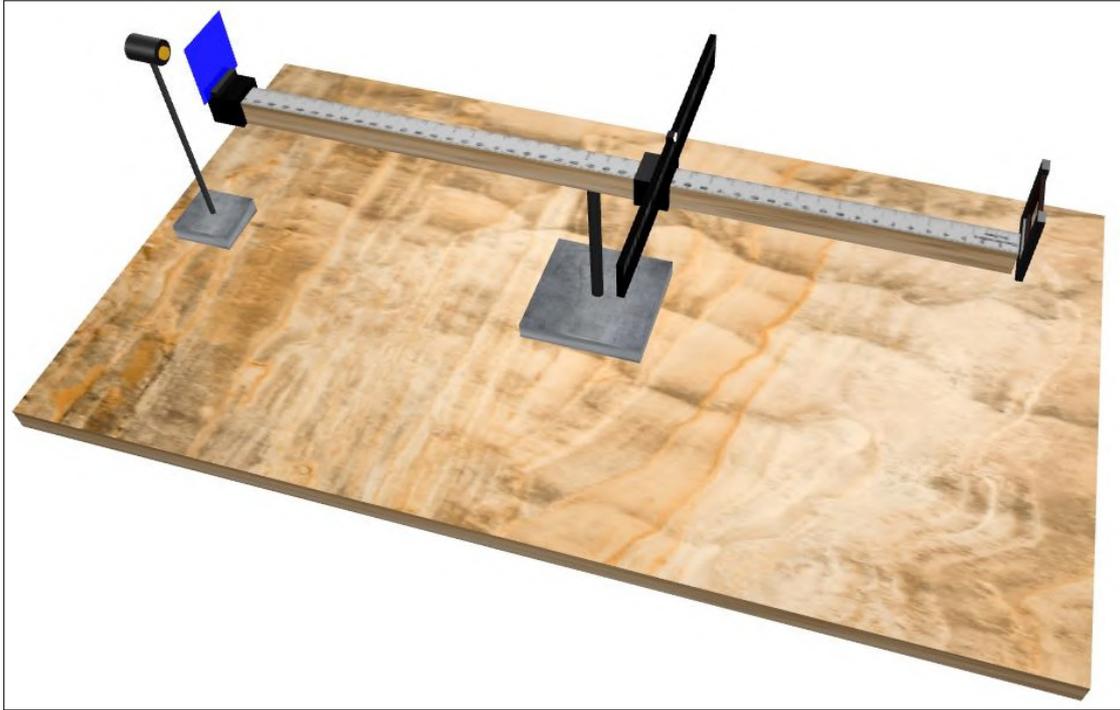


Figure 4: The view of the virtual laboratory from camera #1.

The materials from which these objects are made were created using the Mesh class. It represents triangular polygon mesh-based objects [21]. The instances of the MeshStandardMaterial class and its extensions MeshPhysicalMaterial and MeshPhongMaterial were used for creating some materials such as metal (black and silver), glass, plastic and table. For example, a diffraction grating in three dimensions was created using the code in listing 3.

Listing 3: The code for creating diffraction grating

```
var GratingGlassBox = new THREE.Mesh(new THREE.PlaneGeometry(0.7,
                                                                    0.6), materialGlass);
GratingGlassBox.name = 'GratingGlassBox';
var GratingLftBox = new THREE.Mesh(geometryGratingLftRghtBox,
                                    materialGratingPlastic);
var GratingRghtBox = new THREE.Mesh(geometryGratingLftRghtBox,
                                    materialGratingPlastic);
var GratingUpBox = new THREE.Mesh(geometryGratingUpDwnBox,
                                    materialGratingPlastic);
var GratingDwnBox = new THREE.Mesh(geometryGratingUpDwnBox,
                                    materialGratingPlastic);
```

Other laboratory items were produced similarly, including a colour filter, mobile pointer, and dynamic screen ruler. They allow students to change the diffraction effect and measure the

distance between the centre of the spectral line of a given order and the spectral line at zero order. To utilize the dynamic screen ruler, employ camera view #3 figure 5.

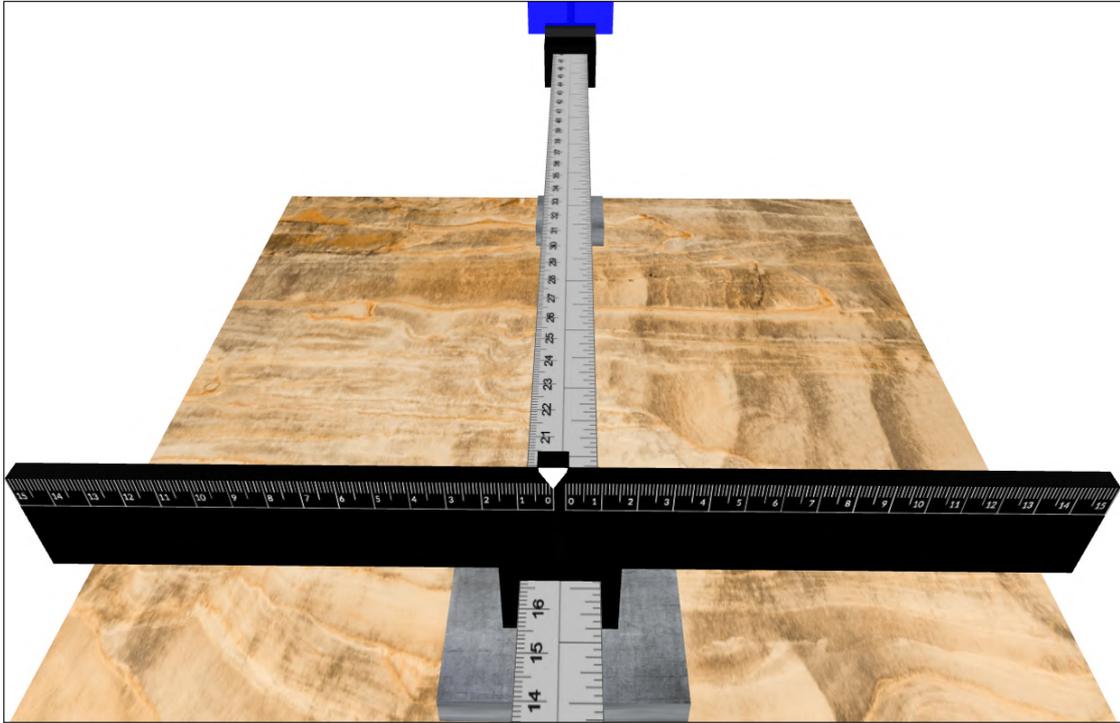


Figure 5: The view of the virtual laboratory from camera #3.

We used three groups to create complex objects such as a lamp and a tripod. They combined graphic primitives and imported images. The code for forming the lamp can be found in the listing 4.

Listing 4: The code for forming the lamp

```
{LampBase.position.set(0, 0, 0);
  LampVerticalCylinder1.position.set(0, 1.8, 0);
  LampHorizontalCylinder1.position.set(0, 3.58, 0);
  LampHorizontalCylinder1.rotation.x = 1.6;
  LampHorizontalCylinder1.rotation.y = 1.6;
  LampHorizontalColorCylinder.position.set(0, 3.572, 0.275);
  LampHorizontalColorCylinder.rotation.x = 1.6;
  GroupLamp.add(LampBase);
  GroupLamp.add(LampVerticalCylinder1);
  GroupLamp.add(LampHorizontalCylinder1);
  GroupLamp.add(LampHorizontalColorCylinder);
  GroupLamp.position.set(0, 1.15, -7);
}
```

The colour filter was created as a Mesh object with the values THREE.BoxGeometry and THREE.MeshStandardMaterial (see listing 5).

Listing 5: The code for forming blue filter

```
var FilterGlass = new THREE.Mesh(
  new THREE.BoxGeometry(1, 1, 0.01),
  new THREE.MeshStandardMaterial({ color: 0x0000ff,
    emissive: 0x0000ff,
    transparent: true,
    opacity: 0.9})
```

The calcPosSpectr() method computes the distance between the centre of the spectral line of a given order and the spectral line at zero order for the three colours (red, green, and blue) utilizing the formula (1). For example, the calculation for the first green line from the left is as follows (see listing 6)

Listing 6: The fragment of method for calculation the distance between the spectral lines

```
function calcPosSpectr()
{
  var elemLine = GroupDinamLine.getObjectByName
    ('DinamLineLeftBox');
  var width_dinam_line = (getWidth(elemLine)/150);
  var PR = ((currentPosDinamBox/1000) /
    currentGrattingSize) * 1000;
  var gl1 = spectr.rw * 1 * PR;
}
```

The first two lines of the calcPosSpectr() method convert matrix distances into screen coordinates. The constant 150 is the length of the ruler on the dynamic screen. In code (5), the variable PR is equal to $\frac{L}{d}$, spectr.gw is the width of the first green line.

Students are expected to perform their own calculations, including using cloud-based platforms such as CoCalc, Maxima, etc [22, 23, 24].

As a result, the diffraction effect can be observed in camera #2 (see figure 6). It also allows the student to measure the position and width of the spectral lines. In addition, boundary conditions are imposed on the positions of the ruler and pointer to ensure that they do not exceed the permitted limits. The displacement of the ruler is exactly 1mm. This was achieved by editing the texture of the ruler itself.

In the virtual lab, we made sure that the measurement result was not artificially tied to a particular value of physical quantities. This is why the diffraction pattern of the slit on the screen is slightly blurred. This makes it possible to obtain different values of distance during visual observation and to calculate the measurement error accordingly.

The real diffraction spectrum can be seen in figure 7. As you can see, the image is rather unclear and it is difficult to determine the position of the spectrum by looking through the diffraction grating. For this purpose, a pointer is provided in the virtual work and it is possible to switch to the view of camera #3 to measure distances more accurately.

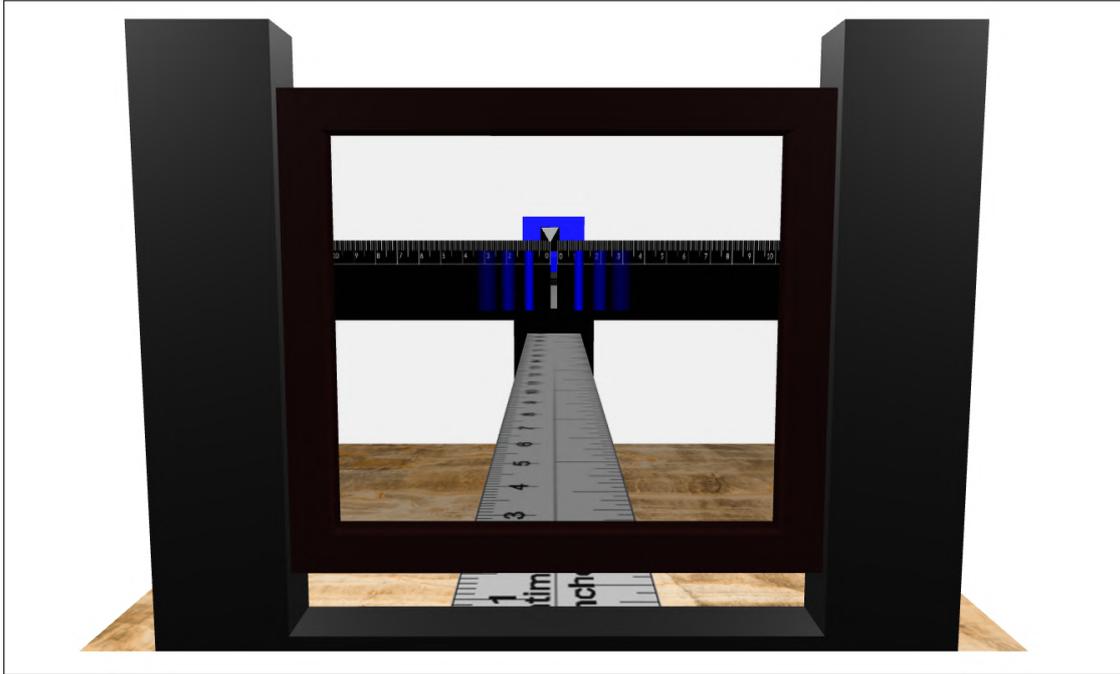


Figure 6: The view of the virtual laboratory from camera #2.

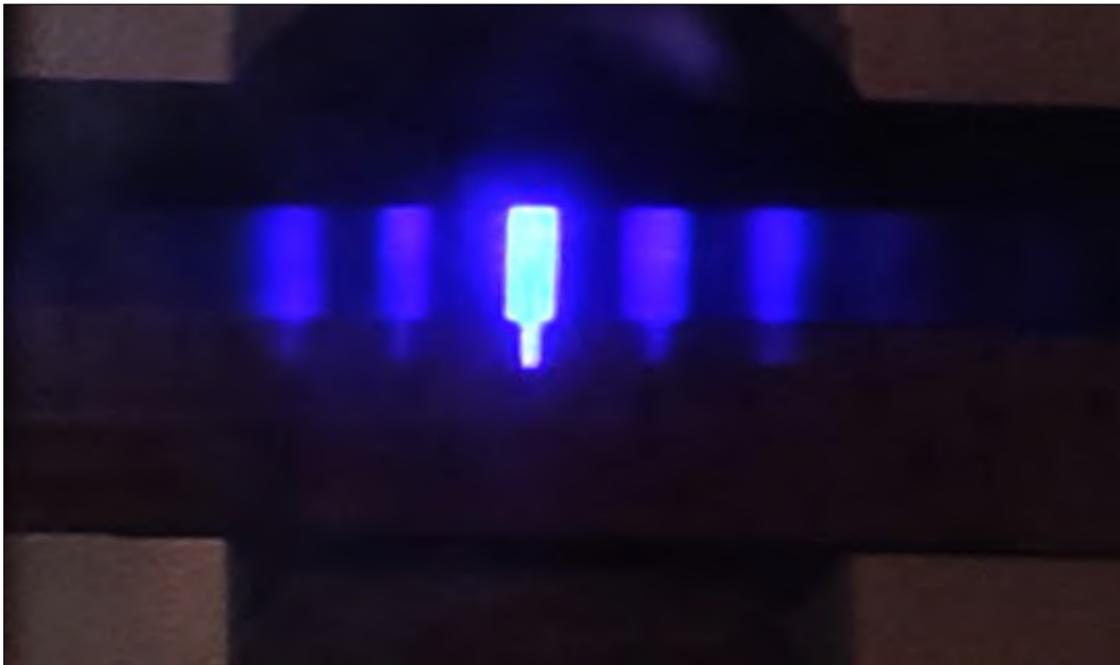


Figure 7: Real diffraction pattern.

We compare the results of the virtual and real experiments. When students used a blue light filter during virtual laboratory work, the following data were obtained: light wavelength $\lambda = 485 \pm 12$ nm with a relative measurement error of 2.3 %. During work on real equipment, the light wavelength was $\lambda = 472 \pm 25$ nm with a relative measurement error of 5.3 %. As we can see, the results within the margin of error coincide and correspond to the wavelength of blue light.

3. Conclusions

Developing virtual physics labs is a real problem. Modern digital tools such as frameworks and programming language libraries make this possible. Based on comparative analysis was selected one of these tools is Three.js. It is a JavaScript 3D library. In this research, the design and development of a virtual physics laboratory for studying the effects of diffraction using Three.js was carried out. The lab development process took place in several stages, such as creating scene, adding camera, creating main instances of classes (WebGLRenderer, OrbitControls), creating scene objects, and defining methods for calculating distances between spectral lines. The developed lab can be useful for blended learning in physics at colleges and universities. It can also be combined with real installations for the study of optics. The disadvantage of this virtual laboratory is the requirement of the Three.js library for modern hardware of the student's or teacher's device. This can be a limitation when working with older computers.

Three laboratory virtual works out of five, which are provided by the curriculum of the "Physics" course covering different sections (mechanics, molecular physics and thermodynamics), were developed using this method. In the future, it is planned to complete the development of the remaining works and combine them into a single integrated environment. This will allow students, in addition to performing work, to collect and process experimental data and prepare reports. We see prospects for further research, such as development of a version of this laboratory to provide student collaboration and its integration with modern learning management systems.

References

- [1] E. K. Faulconer, A. B. Gruss, A review to weigh the pros and cons of online, remote, and distance science laboratory experiences, *International Review of Research in Open and Distance Learning* 19 (2018) 155–168. doi:10.19173/irrodl.v19i2.3386.
- [2] Laboratory Experiences and Student Learning, in: S. R. Singer, M. L. Hilton, H. A. Schweingruber (Eds.), *America's Lab Report: Investigations in High School Science*, Washington, DC, 2006, pp. 75–115. doi:10.17226/11311.
- [3] A. Striuk, O. Rybalchenko, S. Bilashenko, Development and using of a virtual laboratory to study the graph algorithms for bachelors of software engineering, *CEUR Workshop Proceedings* 2732 (2020) 974–983. URL: <https://ceur-ws.org/Vol-2732/20200974.pdf>.
- [4] V. Oleksiuk, O. Spirin, The Experience of Using Cloud Labs in Teaching Linux Operating System, in: O. Ignatenko, V. Kharchenko, V. Kobets, H. Kravtsov, Y. Tarasich, V. Ermolayev,

- D. Esteban, V. Yakovyna, A. Spivakovsky (Eds.), ICTERI 2021 Workshops, Springer International Publishing, Cham, 2022, pp. 281–291. doi:10.1007/978-3-031-14841-5_18.
- [5] N. Balyk, V. Oleksiuk, Y. Vasylenko, G. Shmyger, Designing of Virtual Cloud Labs for the Learning Cisco CyberSecurity Operations Course, CEUR Workshop Proceedings 2393 (2019) 960–967. URL: https://ceur-ws.org/Vol-2393/paper_338.pdf.
- [6] P. P. Nechypurenko, S. O. Semerikov, O. Y. Pokhlietova, An augmented reality-based virtual chemistry laboratory to support educational and research activities of 11th grade students, Educational Dimension 8 (2023) 240–264. doi:10.31812/educdim.4446.
- [7] C. Chotimah, Festiyed, A meta-analysis of the effects of using PhET interactive simulations on student’s worksheets toward senior high school students learning result of physics, Journal of Physics: Conference Series 1481 (2020) 012093. doi:10.1088/1742-6596/1481/1/012093.
- [8] S. Siswoyo, D. Muliwati, Teaching high school physics using PhET interactive simulation, AIP Conference Proceedings 2331 (2021) 030003. doi:10.1063/5.0041657.
- [9] V. Vascak, Physics at school, 2023. URL: <https://vascak.cz/>.
- [10] S. K. Damekova, N. N. Shuyushbayeva, N. K. Tanasheva, G. S. Ahayeva, A. K. Kaliyeva, Additional education of schoolchildren in physics at the Children’s University, Bulletin of the University of Karaganda-Physics 1 (2021) 63–70. doi:10.31489/2021Ph1/63-70.
- [11] CK-12, CK-12 Interactive Physics for High School, 2023. URL: <https://flexbooks.ck12.org/cbook/ck-12-physics-flexbook-2.0/>.
- [12] ExploreLearning, Online simulations for physics, 2023. URL: <https://gizmos.explorelearning.com/physics-collection>.
- [13] M. Holly, J. Pirker, S. Resch, S. Brettschuh, C. Gütl, Designing VR Experiences – Expectations for Teaching and Learning in VR, Educational Technology and Society 24 (2021) 107–119. URL: <https://www.jstor.org/stable/27004935>.
- [14] S. Manabe, C. Mouri, H. Tenzou, T. Kasuga, K. Motoki, Development of an AR chart tool for understanding invisible physical phenomena, AIP Conference Proceedings 1733 (2016) 020001. doi:10.1063/1.4948819.
- [15] D. S. Shepiliev, S. O. Semerikov, Y. V. Yechkalo, V. V. Tkachuk, O. M. Markova, Y. O. Modlo, I. S. Mintii, M. M. Mintii, T. V. Selivanova, N. K. Maksyshko, T. A. Vakaliuk, V. V. Osadchyi, R. O. Tarasenko, S. M. Amelina, A. E. Kiv, Development of career guidance quests using WebAR, Journal of Physics: Conference Series 1840 (2021) 012028. doi:10.1088/1742-6596/1840/1/012028.
- [16] A. E. Kiv, V. V. Bilous, D. M. Bodnenko, D. V. Horbatovskyi, O. S. Lytvyn, V. V. Proshkin, The development and use of mobile app AR Physics in physics teaching at the university, CEUR Workshop Proceedings 2898 (2021) 197–212. URL: <https://ceur-ws.org/Vol-2898/paper11.pdf>.
- [17] O. V. Kanivets, I. M. Kanivets, T. M. Gorda, O. V. Gorbenko, A. O. Kelemesh, Using a mobile application to teach students to measure with a micrometer during remote laboratory work, CEUR Workshop Proceedings 3364 (2023) 87–107. URL: <https://ceur-ws.org/Vol-3364/paper08.pdf>.
- [18] S. O. Semerikov, M. M. Mintii, I. S. Mintii, Review of the course “Development of Virtual and Augmented Reality Software” for STEM teachers: Implementation results and improvement potentials, CEUR Workshop Proceedings 2898 (2021) 159–177. URL: <https://ceur-ws.org/>

Vol-2898/paper09.pdf.

- [19] D. S. Shepiliev, Y. O. Modlo, Y. V. Yechkalo, V. V. Tkachuk, M. M. Mintii, I. S. Mintii, O. M. Markova, T. V. Selivanova, O. M. Drashko, O. O. Kalinichenko, T. A. Vakaliuk, V. V. Osadchyi, S. O. Semerikov, WebAR development tools: An overview, CEUR Workshop Proceedings 2832 (2020) 84–93. URL: <https://ceur-ws.org/Vol-2832/paper12.pdf>.
- [20] T. A. Vakaliuk, S. I. Pochtoviuk, Analysis of tools for the development of augmented reality technologies, in: Proceedings of the 4th International Workshop on Augmented Reality in Education (AREdu 2021), volume 2898, 2021, pp. 119–130. URL: <https://ceur-ws.org/Vol-2898/paper06.pdf>.
- [21] R. Cabello, Three.js docs, 2023. URL: <https://threejs.org/docs/>.
- [22] P. V. Merzlykin, M. V. Marienko, S. V. Shokaliuk, CoCalc: an integrated environment for open science education in informatics and mathematics, CEUR Workshop Proceedings 3482 (2023) 39–53. URL: <https://ceur-ws.org/Vol-3482/paper025.pdf>.
- [23] S. Papadakis, A. E. Kiv, H. M. Kravtsov, V. V. Osadchyi, M. V. Marienko, O. P. Pinchuk, M. P. Shyshkina, O. M. Sokolyuk, I. S. Mintii, T. A. Vakaliuk, A. M. Striuk, S. O. Semerikov, Revolutionizing education: using computer simulation and cloud-based smart technology to facilitate successful open learning, CEUR Workshop Proceedings 3358 (2023) 1–18. URL: <https://ceur-ws.org/Vol-3358/paper00.pdf>.
- [24] M. Shyshkina, U. Kohut, M. Popel, The comparative analysis of the cloud-based learning components delivering access to mathematical software, CEUR Workshop Proceedings 2393 (2019) 93 – 104. URL: https://ceur-ws.org/Vol-2393/paper_241.pdf.