Team VISTA

Methodology

A. Overview

Our ultimate goal is a working prototype: a physical lab and the software allowing students to remotely control the lab to further develop the concept of remote labs and provide more pedagogical insight into the efficacy of remote labs. Our project is particularly focused on engaging students in safe, controlled, and flexible lab environments that allow students to engage in experiential learning, make mistakes and build critical thinking and problem-solving skills. Our lab will also improve accessibility to these crucial learning experiences to disabled students, rural students, and students at schools with fewer financial resources.

To achieve these objectives, we will develop three interfaces: a first interface that allows students to view and control the lab, a second interface that connects the user with the laboratory hardware, and a third interface that executes the user's commands on the laboratory equipment. In developing the first interface, we will experiment with incorporating virtual reality (VR)/augmented reality (AR), haptic touch technology, and voice control of block code controls to create the most authentic experience possible while maximizing accessibility. To build the second interface, we plan on utilizing open-source software, particularly ____, which will be used to develop a communication backend that transmits input from the user and controller interfaces to the physical laboratory components that users will manipulate. Lastly, we plan on coding Arduinos and robotic arms to actuate the user's inputs on the actual laboratory equipment.

By providing an inclusive and equitable learning environment, we expect our project to not only enhance students' understanding of fundamental STEM concepts but also foster their interest in pursuing STEM careers. The impact of this project goes beyond the immediate, direct users. We aim to redefine the implementation and accessibility of educational STEM labs, ensuring that every student worldwide can benefit from experiential STEM education. Our solution will overcome resource limitations and accessibility barriers, making STEM education more inclusive, equitable, and scalable to students across the world.

B. The First Interface: User Interface

A multitude of technological components are required to develop such a remote lab that provides an accurate experience of conducting a scientific experiment within a lab. Of the various methods/components that should ideally be implemented in VISTA's project, two key concepts to consider are augmented and virtual reality. Augmented reality overlays digital content onto real-life environments and objects to enhance the real world with computer-generated perceptual information. Therefore, augmented reality can be a useful tool to "reproduc[e] hands-on labs (experiment and instruments) as much as possible to allow students to move seamlessly from AR representations of test equipment in a remote lab environment to actual test equipment in a real lab." (Odeh et al., 2012). Seamlessly transitioning between hands-on equipment and representations created through augmented reality is a significant step toward our final goal of developing the most realistic and engaging remote lab platform possible.

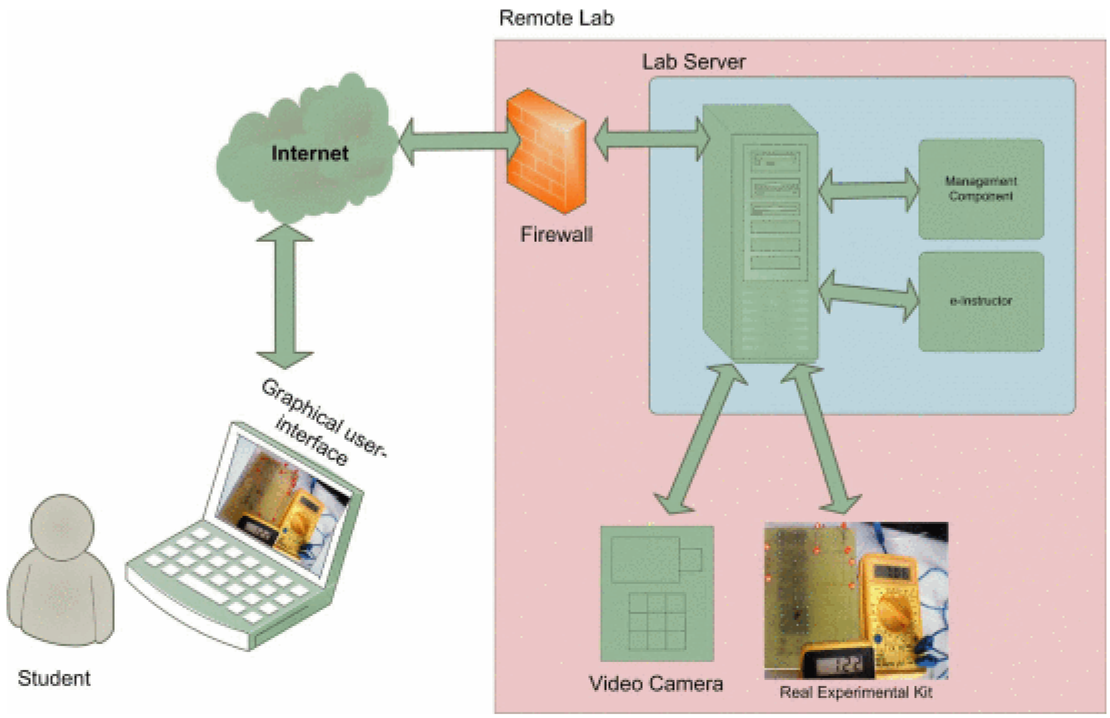

Figure 3. Distributed system architecture of the remote augmented reality engineering lab. (Odeh et al., 2012)

While augmented and virtual reality aren't identical, we can apply both to enhance the mixed-reality aspect of our remote lab. In virtual reality, the user experiences an entirely computer-generated environment independent of the real world. Considering our need for an interface with the ability to remotely conduct experiments with real equipment, relying solely on computer-generation isn't practical. However, augmented reality alone would require users, along with the equipment, to be present locally. Thus, to create a mixed-reality based interface applicable to our system, we must combine AR/VR technology (~$700) with a live video from the lab, necessitating a 360 degree camera (~$300) and software to transmit this live feed to to provide accurate enjoyable and engaging education experience with real representations, demonstrations, etc. (Odeh, et al., 2012). A quality audio transmitter and receiver ($) will also be critical to the system, so the user can both see and hear the equipment they are working with.

Furthermore, tactile gloves/equipment (~$250 for gloves) will also contribute to creating an authentic laboratory experience by recreating as many sensory aspects as possible.

- In order to control the equipment of the lab, the user will be able to control the lab equipment using a
  - In order to run powerful Virtual Reality simulations and/or Augmented Reality programs, a computer with a powerful processor, like an Intel i9-13900K (~$450), is needed to ensure our platform can smoothly run virtual/augmented reality programs for extensive periods of time while maintaining efficient data handling and quality performance. Therefore, possible choices for the computer and processor combinations include the M3 Pro processor of a MacBook Pro.
  -
- Simulink/matlab code control
  - https://ieeexplore.ieee.org/abstract/document/6140966
  - Working with these types of practices, users need to have MATLAB and Simulink software installed in order to modify the downloaded Simulink file. When a practice with the user-defined controller is selected, a Web page is displayed. In Fig. 4, the Web page of practice with user-defined controller for thermal system is shown.
  - The users can freely modify the reference input signals in the subsystems and controller1, controller2, controller3 for each electro-pneumatic actuator by means of MATLAB and Simulink package software. However, they cannot modify the name or the number of input/output ports of the subsystems. In addition, users can change the sample time, execution time and others important simulation parameters.
  - Availability: Web-based learning systems should be available 24 hours a day. This implies that the system should have self-protection rules for accomplishing this requirement. All the experiments must be equipped with hardware and software devices to prevent damage to the components and the people working in the laboratory.

- https://ieeexplore.ieee.org/abstract/document/5204073:
- The technologies and the programming languages generally used to connect the device and the local computer are often based on proprietary software, mainly Matlab with Simulink and LabView with data socket, with some software based on Visual Basic or Python [4], [5], [6], [7], [8], [9]
- It must be collaborative: Like in a multiconference, small groups of two to 20 participants (students and teachers) should be able to see/hear/talk to each other, supported by various collaboration tools (shared whiteboard, picture annotation, chat, etc.); moreover, they should see (all together) and remotely control (one at a time) the laboratory equipment.

Our lab won't be designed one-dimensionally, meaning basic control using a physical controller will not be the sole method available to the user. To maximize the accessibility of our prototype, another important component is voice control, which would involve a high-end microphone, (ex. Logitech Blue Yeti (~$100)) along with a suitable microcontroller, such as an Arduino Mega (~$150) to pick up the user's commands.

A critical component of the lab setup will likely be the video stream on the user interface, as illustrated in Figure Y. Several widely used programs could serve this purpose, including OBS Studio (Obsproject, 2024) and Owncast (Owncast, 2024). These programs offer similar features and functionality, with key distinctions in their coding languages and documentation support. OBS is written in C with extensive documentation, while Owncast is written in GO and has less comprehensive documentation.

In addition to the video stream, we are considering the potential benefit of object recognition software integrated into the video to highlight specific lab elements that might be challenging to identify visually. MMDetection, an open-source recognition tool coded in Python, could be a useful option for this feature (Huang, 2024).

Further aspects of this project that will need development include the user interface (UI) itself and the connections pipeline linking the UI to the lab equipment. The UI could potentially be developed as a web-based interface using tools like Chrome DevTools or any other HTML-compatible applications. Alternatively, or additionally, an application could also be developed using coding environments like Visual Studio Code or GitHub, aiming for compatibility with both computers and mobile devices.

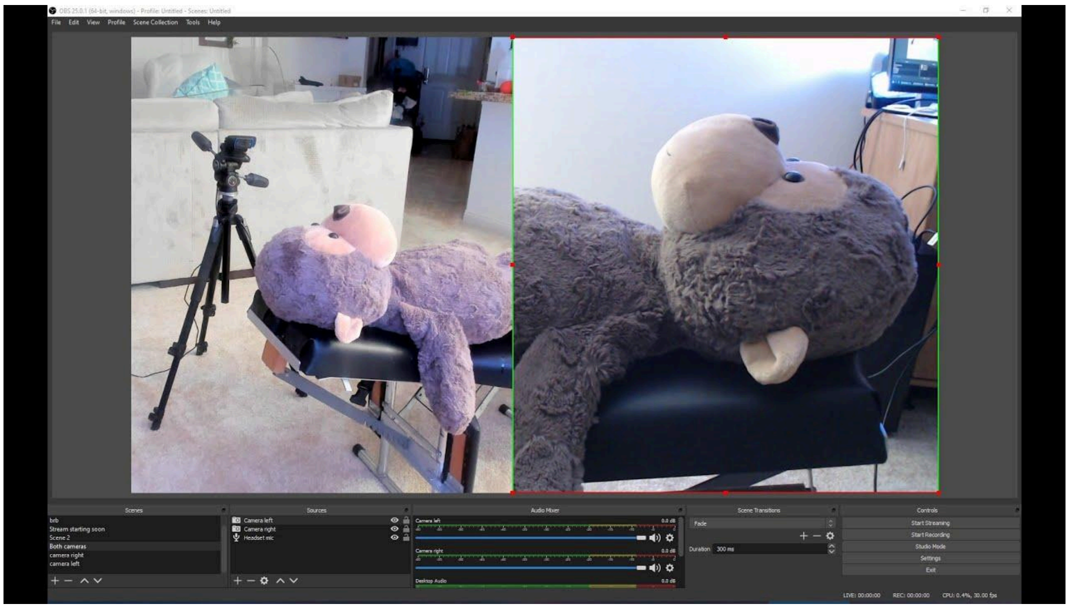

Figure Y. OBS Studio being used to broadcast live footage of a physical space.

C. The Second Interface: Connecting Interface

The connection between the laboratory components and the user interface must be able to transmit data both securely and fast enough to provide a real-time experience. For our security standard, the connecting interface should be data encrypted to prevent sensitive data such as location being intercepted by others. The connection should also prevent unauthorized access and connections.

In terms

https://ieeexplore.ieee.org/abstract/document/5204073

To connect the UI to the lab interface, Arduino Ethernet may provide a straightforward solution if we process with Arduino in the final design, as it supports remote connectivity. If an Arduino-based solution is not pursued, other internet connectivity options will be explored to ensure reliable interaction between interfaces.

D. The Third Interface: Lab Interface

Due to the number of technological components we need to implement in our project, multiple methods of control and connection are required. As far as physical connections, we will need a vast amount of wiring and electrical equipment such as resistors and capacitors to develop functioning circuits. For these circuits to function, a microcontroller is needed to allow software to manipulate the hardware components. Our equipment will be connected to our software with Arduino Mega(s). Arduino Megas have a large number of pins available compared to alternative microcontrollers, with the ability to connect to WIFI, making them increasingly flexible. This flexibility is necessary given the large number of mechanical devices that need to be controlled, including multiple motors for our robotic components, lights, cameras and sound equipment, etc. Although these components may each control different aspects of the system, having a single source manage them can simplify their interactions, as their code would reside in the same file. Given that some of these components may require more than 5V, which is the maximum power output of an Arduino Mega, a solid state relay may be necessary (~$15) as well as an external power source.

In terms of the physical system as a whole, where it is stored and how it is maintained is of utmost importance. For our current research, our remote lab will be stored in an electronics lab such that we can consistently test and modify it's components. However, our goal is to create a proof of concept that can eventually be applied to a large-scale operation. In that case, we would require a large room, or rooms, capable of hosting many remote labs. According to the developers of Practable, a group that operates remote labs out of the University of Edinburgh, automation is key in sustaining this large-scale facility (Reid 2022). While situations requiring in-person intervention are unavoidable, such as performing safety checks, minimizing the amount of staff is crucial in order to remain affordable.

When broken down into three components or "interfaces", much of the technology anticipated for remote labs is already available through open-source resources that allow for use and modification. For the interface controlling the lab equipment, two primary options we are exploring include an IR emitter (a device that uses infrared light to remotely control components) and a robotic arm. Both options have open-source software and resources available. For IR emitters, we are considering DIYRemote (Hannah, 2021) and Arduino-IRemote (ArminJo, 2024). As for the robotic arm, Annin Robotics (Annin, n.d.) provides open-source code and a customizable kit with most arm components, though some parts require 3D printing, which is feasible within UMD facilities. Figure X shows an example of a fully assembled Annin Robotics arm. An alternative robotic arm, the BCN3D MOVEO, may also be considered due to its lower cost, being fully 3D-printed (BCN3D, 2016).

Currently, we anticipate allocating funds towards an Annin Robotics arm, as it offers more extensive documentation and higher-quality components However, we may explore the BCN3D MOVEO if budget constraints arise.. IR emitters could potentially offer a simpler setup, as they are less complex to program and maintain than robotic arms, which might introduce additional failure points. Additionally, IR emitters are typically more cost-effective, priced at under $20 per emitter, compared to approximately $1,200 for a robotic arm from Annin Robotics. However, robotic arms could provide a more interactive and engaging experience for students, and we plan to gather further data on student preference to assess this potential benefit.

For now, the team anticipates experimenting with both IR emitters and robotic arms to determine which is most effective within the lab setup, with robotic arm usage contingent on available funding.   Between the two open-source code options for IR emitters, Arduino-IRemote (ArminJo, 2024) aligns more closely with our plan to incorporate Arduinos into our remote labs, making it a preliminary preference over DIYRemote, though this choice may be reassessed as the project progresses. as our choice for IR emitter code.

Figure X. Example of Annin Robotics arm

Requirements for Success

Our primary objective is to develop a functional prototype that meets our four accessibility standards:

1. Students' ability to meaningfully interact with physical lab equipment from a separate location, especially lab equipment they would usually not have access to
2. Disabled students' ability to meaningfully interact with such lab equipment
3. The remote lab system must showcase qualitative/quantitative proof of providing a greater educational experience than virtual laboratories
4. The cost of this remote laboratory must not exceed the cost of its physical equivalent

Accessibility standard #1 is the baseline for a successful remote laboratory and requires that:

- Our team develops hardware and software that controls an experiment without human assistance
- The user interface allows students to remotely control this lab equipment, while a simultaneous live feed provides a real-time visual representation of the experimental outcomes
- Labs too risky to perform in-person can be conducted via our platform

Accessibility #2 requires that certain features be added to our remote interface that account for a variety of disabilities. These features may include audio descriptions of the lab, varying colors, voice control, etc.

Standard #3 requires our team to evaluate the impact of our efforts. We will measure this impact by testing our prototype with students and evaluating their experience. This could be done, for example, on Maryland Day, where we will have an open table for students to come and test our prototype and involvement with local public schools.

Finally, standard #4 requires that our product be cost-efficient by making the system less costly than typical lab setups, ensuring wider accessibility for institutions with limited resources.

#1 is the baseline standard for any successful remote laboratory. By definition, an operational remote lab requires that:

- Our team develops hardware and software that controls an experiment without human assistance
- The user interface allows students to remotely control this lab equipment, while a simultaneous live feed provides a real-time visual representation of the experimental outcomes
- Labs too risky to perform in-person can be conducted via our platform

The basis of the software for our remote lab will be the user interface and video stream.

The final product should be functional in terms of mechanical and software errors, and the user interface should be intuitive and clear. The physical lab should (insert our standard for the physical lab here, like rate of it messing up and stuff). Both the connecting interface as well as the user interface will be thoroughly tested for software bugs (standard/procedure for that). The visual clarity of the user interface will have to be gauged through qualitative feedback from (surveys i guess, still not sure about things).

However, the resource-intensive nature of remote labs is a significant limitation. They often require complex setups, including sophisticated hardware and robust network infrastructures, which may make them difficult to scale for broader educational use (Potkonjak et al., 2016). Maintaining and upgrading this equipment can also be costly and labor-intensive, posing challenges for long-term sustainability.

Beyond engagement challenges, another common concern with remote labs is the lack of flexibility of teachers to select or personalize the labs they teach because designers and the free market determine the availability of virtual versions of specific labs (Xie, 2022). As such, for remote labs to become a handy tool to support online experimentation at all levels, they must address teachers' diverse needs by incorporating flexibility, interactivity, and open-endedness (Xie, 2022). As such,

The second standard involves features meant to improve accessibility for students who have disabilities that would inhibit them in use of a standard remote lab. The features would entail additional tools that supplement physical or mental help to make certain functions of our lab easier to use. Many students will be limited in their ability to control the remote lab machinery with the lab controller. Voice control is a feature we will implement to serve as an alternative method of using the lab machinery. (..........)

Feedback for this standard will be gathered through (yeah). Successful implementation of accessibility for students with disabilities should involve (9 out of 10 dentists recommend our virtual lab, or something)

Our third standard is predicated on the feedback we get from audience testing and examination. We will measure the effectiveness of our remote lab by having students test the product. To obtain ample feedback, we will have open tables for students to come and test our prototype on days when student traffic is at its highest, such as Maryland Day. In addition, we will also require feedback from academic institutions. Our primary method of obtaining this feedback will be working with our local public schools to find a time and place to have educators test our prototype. Student feedback is usually vague, so we must develop a quantifiable scale that will allow us to measure the results. The scale that we will use…… (discuss with the team).

will rely on feedback provided from tests and surveys. (go into procedure for different groups that use different labs, pre and post tests, surveys, etc, i guess we could talk about this as a team)

How to measure effectiveness

1) Student learning outcomes compared to students using virtual lab and in-person lab for the same experiment, using pre and post-examinations
2) How students rate their enjoyment of the lab compared to students using virtual lab and in-person lab for the same experiment
3) How often students "fail" before ultimately completing the lab compared to students using virtual lab and in-person lab for the same experiment
4) How many different multi experiential paths students experience compared to students using virtual lab and in-person lab for the same experiment
5) How many how quickly students actually use the same equipment in-person compared to students using virtual lab for the same experiment
6) How many students would like to participate in a another similar experiment compared to students using virtual lab and in-person lab for the same experiment
7) How long it takes to perform the experiment compared to students using virtual lab and in-person lab for the same experiment

8) How much does it cost for a user to do the experiment (total cost (capital, maintenance, etc) per student per experiment compared to students using virtual lab and in-person lab for the same experiment
9) How students rate their confidence with lab technique compared to students using virtual lab and in-person lab for the same experiment
10) How students rate their confidence with the learning content compared to students using virtual lab and in-person lab for the same experiment
11) User reviews on the website

Test individual components/interfaces

We will evaluate student feedback at each stage of the remote lab's development. This will allow us to avoid any backtracking regarding the construction and provide guidance for improving the remote lab.

Similarly, accessibility standard #4 will be tracked during each stage of the remote lab's development. As we add interfaces to the initial infrastructure, we will update the cost of the lab with the price of the materials utilized. The current cost of our remote lab will then be compared to those of the physical labs we have analyzed and researched. Regular comparisons between our remote lab and established physical labs throughout development will ensure that we abide by our affordability standards.

Conclusion:

Utilizing an HTML-based application, such as Chrome DevTools or OBS Studio, we will create a simple user interface that allows students to navigate the video stream of the physical hardware that they will be able to manipulate. Additionally, we will need to develop the interface that controls the physical machinery in the lab. The robotic arm will be built using a machinery kit created by Annin Robotics (Annin, n.d.), and the backend for the communication between the controller and arm will use Annin Robotics' open-source code as a template. Any additional material needed to build the robotic machinery will be manufactured using UMD's 3D printers.

According to a study done by the National Science Foundation, students with visual disabilities require audible commentary throughout the experiment (Moon 2012), a feature that Team VISTA can implement.

As such, Team VISTA could provide this failure by displaying live, accurate results that reflect student performance.

Hence, Team VISTA should collaborate with teachers to develop a plan enabling our system to eventually integrate seamlessly with various laboratory setups.

From 2009-2019, there were over 80,000 VISIR sessions and over 2,000,000 actions in those sessions performed by students (Garcia-Zubia et al., 2021). While VISIR represents elements similar to what Team VISTA seeks to achieve, some aspects and goals differ from those of this commercial remote lab. The first difference lies in the types of labs that VISTA aims to target; while VISIR works with circuits and electricity, which VISTA may also work with, VISTA aims to work with other types of labs that are more high school level, working with chemistry, physics, or electricity. Additionally, as previously noted, VISTA also seeks to work with dangerous labs that may spark interest in students to pursue, yet may be too risky to perform in-person or by the student. Secondly, VISTA aims to work with more physical components instead of just a matrix and an equipment server, VISTA wants to work with movement, playing around with ways to experiment with different types of formats, like robotic arms, wheels, and other components. The similarity grasped from this remote lab includes the elements that connect the parts to make them remote. VISTA seeks to develop a front-end interface that works with the student, find a way to get student input from one location to another, and finally return the output through a camera screen that the student may view. These are elements demonstrated by VISIR that VISTA may implement.

However, because of the elevated sophistication of netCIRCUITlabs, the main difference between netCIRCUITlabs and VISTA is the complication and types of labs. VISTA aims to work with lower-level students, expanding the types of labs instead of focusing only on circuits and electricity. Both VISIR and netCIRCUITlabs work primarily with experimentation, with different functions and their results. On the other hand, VISTA seeks to work more with education, hoping to develop interest in students and assist them in completing different types of labs. While there are apparent differences between the goals that VISTA has and those currency available, there are many similarities that may be implemented to reach the goals that VISTA sets out to achieve.

[1] Annin, C. (n.d.). Annin Robotics. https://www.anninrobotics.com/

[2] ArminJo. (2024). ArInfrared Remote Library for arduino. GitHub. https://github.com/Arduino-IRremote/Arduino-IRremote.

[3] BCN3D. (2016, July 28). BCN3D MOVEO: A fully open source 3D printed robot arm. BCN3D Technologies. https://www.bcn3d.com/bcn3d-moveo-the-future-of-learning-robotic-arm/

[4] Hannah, S. (2021). DIYRemote: Create your own universal remote control using HTML. GitHub. https://github.com/shannah/DIYRemote.

[5] Huang, H. (2024). Openmmlab detection toolbox and benchmark. GitHub. https://github.com/open-mmlab/mmdetection

[6] Obsproject. (2024). OBS Studio - free and open source software for live streaming and Screen Recording. GitHub. https://github.com/obsproject/obs-studio

[7] Odeh, S., Shanab, S. A., Anabtawi, M., & Hodrob, R. (2012). Remote augmented reality engineering labs. Proceedings of the 2012 IEEE Global Engineering Education Conference (EDUCON), 1-6. https://doi.org/10.1109/EDUCON.2012.6201162

[8] Owncast. (2024). Owncast. GitHub. https://github.com/owncast/owncast

[9] Reid, D., Burridge, J., Lowe, D., & Drysdale, T. (2022). Open-source remote laboratory experiments for controls engineering education. International Journal of Mechanical Engineering Education, 50(4), 828–848. https://doi.org/10.1177/03064190221081451