Ref. Ares(2024)1818563 - 08/03/2024

XR2LEARN

D3.1 XR2LEARN BEACON APPLICATIONS

WP3 – XR Technology PUSH

February 2024

Funded by the European Union

Funded by the European Union under the Grant Agreement 101092851. Views and opinions expressed are however those of the athor(s) only and do not necessarily reflect those of the European Union or European Health and Digital Executive Agency (HADEA). Neither the European Union nor the granting authority can be held responsible for them.

Author	LS Group
Work Package	WP3
Delivery Date	08.03.2024
Due Date	29/02/2024
Classification	Demonstration, Public

Status of deliverable

Action/role	Name	Date (dd.mm.yyyy)
Submitted by	Antoine Lasnier (LS)	29.02.2024
Responsible (WP leader)	Antoine Lasnier (LS)	29.02.2024
Approved by (internal reviewer)	Ioannis Chatzigiannakis	01.03.2024

Author(s) contact information

Name	Organisation	E-mail
Antoine Lasnier	LS	Antoine.lasnier@ls-group.fr
Benjamin Capel	LS	Benjamin.capel@ls-group.fr

Document reviewers

Name	Organisation	E-mail	Comments
Ioannis Chatzigiannakis	CNIT	ichatz@diag.uniroma1.it	
Alessio Gugliotta	EITM	Alessio.gugliotta@eitmanufacturing.eu	

Document history

Date	Version	Author	Comments
09.12.2023	v0.1	LS	Initial version
30.01.2024	v0.2	LS / HOU / Maggioli	Adding up-to-date content for each Beacon App description paragraph
06.02.2024	v0.3	LS	Proofreading, adding some missing paragraphs, redacting Conclusion
08.02.2024	v0.4	LS	Font, layout homogenization (Work In Progress)
26/02/2024	v0.5	LS	Last content added (BA3 Instructions) + page layout and numbering rework.
29/02/2024	V0.6	LS	Address QA comments
08/03/2024	V1.0	LS	Final version

Disclaimer

Acknowledgment

This document is a deliverable of the XR2Learn project. This project has received funding from the European Union's Horizon Research and Innovation programme under grant agreement № 101092851.

Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Executive Agency, while neither the European Union nor the granting authority can be held responsible for any use that may be made of the information it contains.

Table of Contents

List of abbreviations 6
Executive Summary 7
1. Introduction 8
2. Beacon Application 1: Laser Cutting Machine 10
- 2.1 Description 10
- 2.2 State of the Art 11
- 2.3 Content 12
- 2.4 Architecture 13
  - 2.4.1 Project structure 13
  - 2.4.2 Avatar representation 14
  - 2.4.3 Training scenario and gamification 14
- 2.5 Instructions 16
  - 2.5.1 Prerequisites 16
  - 2.5.2 How to install / launch the Demo 16
  - 2.5.3 Controls 17

3. Beacon Application 2: V-Lab, Virtual Laboratory 18
- 3.1 Description 18
- 3.2 State of the Art 19
- 3.3 Content 20
- 3.4 Architecture 21
- 3.5 Instructions 23
  - 3.5.1 Installation 23
  - 3.5.2 Usage 24
  - 3.5.3 Microscopy process 25

4. Beacon Application 3: Industrial equipment training and safety 26
- 4.1 Description 26
- 4.2 State of the Art 26
- 4.3 Content 27
- 4.4 Architecture 28
  - 4.4.1 Synthetic Dataset Creation 29
  - 4.4.2 AI Model Creation 33
- 4.5 Instructions 35
  - 4.5.1 Dependencies 35
  - 4.5.2 Quick Start 35

5. Conclusion 41

LIST OF ABBREVIATIONS

BA	Beacon application
EdTech	Educational Technologies
KPI	Key performance indicator
SSL	Self-supervised learning
WP	Work package
XR	Extended reality
Partners' names and acronyms	
CNIT	CONSORZIO NAZIONALE INTERUNIVERSITARIO PER LE TELECOMUNICAZIONI
F6S	F6S NETWORK IRELAND LIMITED
MAG	MAGGIOLI SPA
LS	LIGHT AND SHADOWS
SYN	SYNELIXIS SOLUTIONS SA
SUPSI	SCUOLA UNIVERSITARIA PROFESSIONALE DELLA SVIZZERA ITALIANA
UM	UNIVERSITEIT MAASTRICHT
HOU	HELLENIC OPEN UNIVERSITY
EADTU	VERENIGING VAN EUROPEAN DISTANCE TEACHING UNIVERSITIES
EITM	EIT MANUFACTURING SOUTH SRL

EXECUTIVE SUMMARY

This deliverable, D3.1 XR2Learn Beacon Applications, represents the innovation and collaborative efforts of the first 14 months in developing three pioneering XR applications: laser cutting machine simulation, chemistry lab simulation, and machine learning training utilising XR datasets. Each application stands as a beacon of immersive learning, fostering the advancement of XR technologies within education. The three beacon applications provided by the XR2Learn consortium will later be enriched by other beacon applications developed by the XR2Learn community.

Note : as specified in the Description of work, *Deliverable 3.1 is a DEMO type deliverable. The demonstration of BA1, BA2 and BA3 have and will be performed during live demonstrations and events. Developments are also illustrated in videos. This optional report is an addition to the demonstrations and provides further information on each beacon application guiding the XR community toward best practices in VR application development and utilisation.*

1. INTRODUCTION

Embedded within Work Package 3 XR Technology Push, this deliverable describes the work carried out in Task 3.1 “XR2Learn Beacon applications” and revolves around the concept of Beacon applications, strategically crafted as immersive learning environments. These applications not only facilitate analysis and comprehension of best practices for VR application creation, but also serve as foundational tools for educational enhancement. Targeting a diverse audience including XR technology providers, designers, educators, developers, and decision-makers, XR2Learn strives to democratize the landscape of immersive learning, in particular through these beacon applications.

These pilot applications serve as valuable sandboxes for the XR community and are delivered as open-source projects hosted on the XR2Learn GitHub repository (https://github.com/orgs/XR2Learn/repositories) so that anyone can clone, analyse, modify, adapt and improve at will.

Through the delivery of three VR applications, along with their sources and documentation, we aim to provide a comprehensive toolkit for exploration and learning aimed at stimulating the interest of external stakeholders. This list of beacon applications is expected to possibly grow with contributions from FSTP projects.

This deliverable includes three sections, one for each beacon application, with a summary of their content, target audience, architecture, and functionalities. For each section, further elaboration on specific details, technical aspects, and user guidance are added to provide a comprehensive understanding and facilitate seamless utilization of the beacon applications.

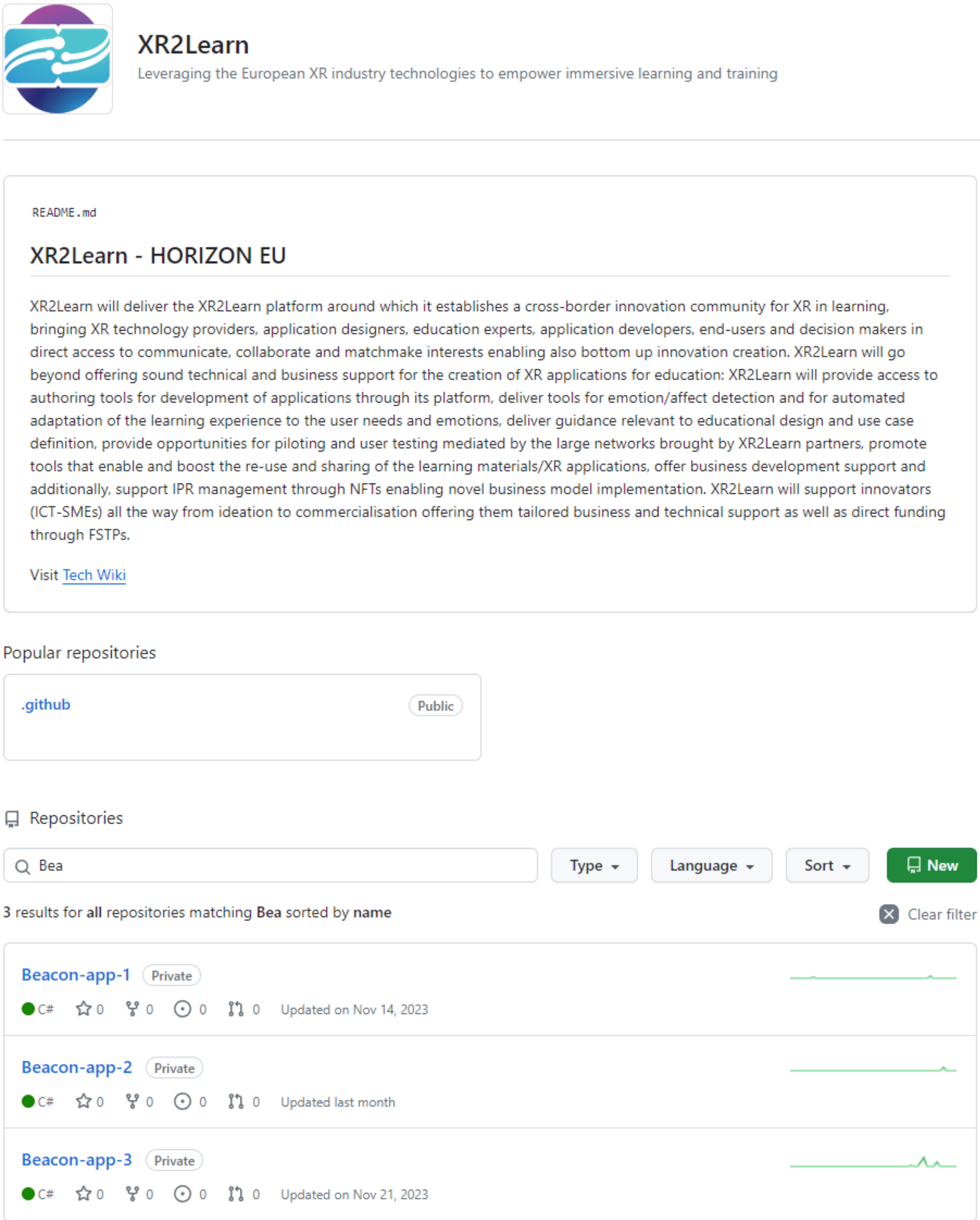


Figure 1: XR2Learn Github repository for Beacon Applications

2. BEACON APPLICATION 1: LASER CUTTING MACHINE

A. 2.1 DESCRIPTION

Tailored for engineering and manufacturing students, this application simulates real-world scenarios, allowing users to learn and practise a maintenance procedure on a laser cutting machine within a safe, virtual environment. The maintenance procedure consists of properly turning the machine off, removing mechanical parts (mirror, nozzle, plate, ...), cleaning individual elements, and mounting the machine back in a precise order.

Beacon application 1 video can be found here : XR2Learn Laser Cutting Machine (https://youtu.be/olxAlag3-S4?si=Mige01e9C5FYHTLE&t=137)

We used the enabler INTERACT to develop this Beacon Application on how to maintain a Trotec Speedy 400 laser cutting machine. The application illustrates the authoring process of immersive training applications using INTERACT from 3D data import to interaction configuration and scenario description. The source Unity project can serve as a powerful starting point to address other industrial scenarios, and authors can bring their own 3D data and use cases to modify or adapt this application.

The particular aspects of VR authoring for industrial use cases illustrated in this beacon application are :

- 3D asset optimization
- Realistic interaction with object
- Scenario creation with adjusting level of difficulty
- Visual effects

Figure 2: Illustration of the Trotec laser cutting machine for Beacon Application 1

B. 2.2 STATE OF THE ART

Various XR-based authoring tools for developing industrial training scenario have been proposed in the relevant literature to study how they can be used to assist in knowledge creation[1]. Initially authoring tools provided a low-level framework that required the author to provide code[2]. More recent tools are starting to follow a low-code approach that make high-fidelity VR prototyping easier[3]. Most of these authoring tools are generic and are used to develop proof-of-concept ITS with a main goal to introduce certain features and study how they are perceived by non-technical designers. For the case of industry 4.0 related VR applications, ARAUM[4], ARTA[5] and WAAT[6] are among the very few authoring tools available. These tools focus on how to provide guidance of the industrial procedures that need to be performed by the user.

In contrast to these approaches, Beacon application 1 and the authoring tool INTERACT allows to provide realistic behaviour of the different components that comprise a workstation and thus move beyond simply displaying the instructions. Different components such as the cables, frames and lenses can be properly handled within the XR environment thus providing an immersive learning experience.

Another differentiating aspect of this application is its commitment to user engagement and effective learning. The application incorporates elements of gamification, which are carefully aligned with pedagogical principles. The user is presented with multiple difficulty levels, which cater to different levels of experience and expertise. This strategy not only ensures that the content is appropriate for each user, but also provides a sense of progress and accomplishment as users advance through the levels. Furthermore, a scoring system is implemented at the end of each session. The scoring system is based on several factors, including time consumption,

[1] Narges Ashtari, Andrea Bunt, Joanna McGrenere, Michael Nebeling, and Parmit K. Chilana. 2020. Creating augmented and virtual reality applications: current practices, challenges, and opportunities. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems (CHI '20). Association for Computing Machinery, Honolulu, HI, USA, 1–13. isbn: 9781450367080. doi: 10.1145/3313831.3376722.

[2] Dieter Schmalstieg, Anton Fuhrmann, Gerd Hesina, Zsolt Szalavári, L. Miguel Encarnação, Michael Gervautz, and Werner Purgathofer. 2002. The Studierstube Augmented Reality Project. Presence: Teleoperators and Virtual Environments, 11, 1, (Feb. 2002), 33–54. eprint: https://direct.mit.edu/pvar/article-pdf /11/1/33/1623621/105474602317343640.pdf. doi: 10.1162/105474602317343640.

[3] Lei Zhang and Steve Oney. 2020. Flowmatic: an immersive authoring tool for creating interactive scenes in virtual reality. In Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology (UIST '20). Association for Computing Machinery, Virtual Event, USA, 342–353. isbn: 9781450375146. doi: 10.1145/3379337.3415824.

[4] John Ahmet Erkoyuncu, Iñigo Fernández del Amo, Michela Dalle Mura, Rajkumar Roy, and Gino Dini. 2017. Improving efficiency of industrial maintenance with context aware adaptive authoring in augmented reality. Cirp Annals, 66, 1, 465–468.

[5] Michele Gattullo, Giulia Wally Scurati, Michele Fiorentino, Antonio Emmanuele Uva, Francesco Ferrise, and Monica Bordegoni. 2019. Towards augmented reality manuals for industry 4.0: a methodology. Robotics and Computer-Integrated Manufacturing, 56, 276–286.

[6] Pierre Bégout, Thierry Duval, Sébastien Kubicki, Bertrand Charbonnier, and Emmanuel Bricard. 2020. Waat: a workstation ar authoring tool for industry 4.0. In Augmented Reality, Virtual Reality, and Computer Graphics. Lucio Tommaso De Paolis and Patrick Bourdot, (Eds.) Springer International Publishing, Cham, 304–320. isbn: 978-3-030-58468-9.

accuracy, and the use of hints or skip buttons. This comprehensive approach to scoring encourages users to optimise their performance while offering meaningful feedback.

C. 2.3 CONTENT

Figure 3: Picture of an apprentice learning in Virtual Reality

Beacon Application 1 scenario illustrates maintenance procedures of a Trotec Speedy 400 laser cutting machine. It involves several steps to ensure the machine is properly maintained and continues to operate efficiently. The process typically involves turning off the machine and unmounting various components, such as the mirror, lens, and nozzle. The working table and main enclosure are then vacuumed to remove any dust or debris. The plate is wiped with a sponge, and the various components are remounted and the machine is turned back on. Wiping the lens and nozzle with a fiber cloth including removal or particles is also simulated in the application. When the cleaning is complete, the components need to be remounted. Finally, the working table and main enclosure are then vacuumed to remove any dust or debris.

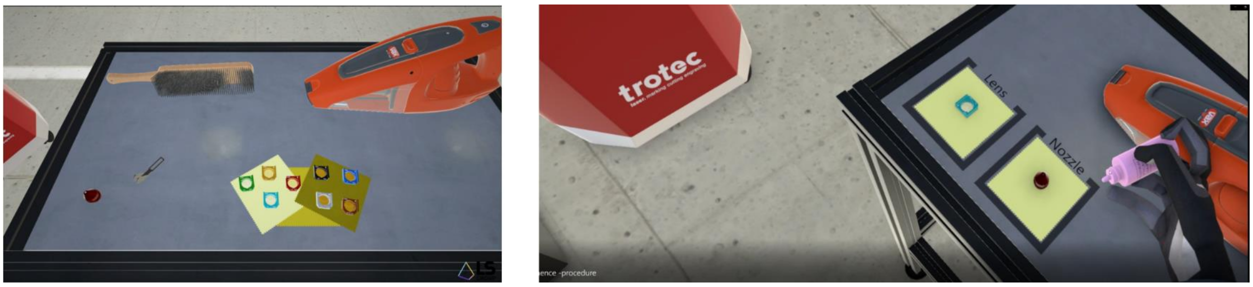


Figure 4: Unmounting and cleaning the machine components

This step-by-step procedure application addresses different levels of learning: from beginners to experienced workers who want to quickly learn how to operate a new machine model. To do so, the beacon app integrates a helping system that the user can trigger at any time during the scenario: to get a hint on how to proceed (display a tasklist, show which part should be moved, and where, or simply skip a step of the procedure).

Depending on the amount of help the user has used during the scenario, and the time he took to complete it, the user ultimately obtains a score in different evaluation categories: rapidity, safety, and autonomy.

He can then try to beat his own high score, or compare with other students. This learning loop is particularly engaging, and may even allow us to connect to LMS systems easily.

D. 2.4 ARCHITECTURE

This Beacon App has been built using Enabler 1: INTERACT. It is a typical example of how INTERACT can be used as a framework to create step-by-step immersive learning procedures from industrial 3D CAD data.

The resulting Unity project has been delivered in the XR2Learn public repository. The community can therefore analyze the project structure and use it as a sandbox for learning how to produce similar use cases. In particular, XR authors opening Beacon application 1 project can :

- Import new 3D models (native CAD file or 3D scan)
- Define Interactable components. An interactable component is an object requiring physical interaction in order to be manipulated and assembled in the scenario
- Configure part to be assembled (kinematics and dynamic behavior)
- Define part targets (keypoint) in the final assembly. These keypoints will also serve as helpers for the worker (displaying a ghost of the part destination)
- Create a step by step scenario corresponding to the work sequence by configuring the order of keypoints. The scenario acts as a state machine defining when assembly steps should be activated/deactivated
- Import extra media to provide further information on the training (pdf, images, videos, audio files)
- Set up assistance and feedback

2.4.1 Project structure

The Unity project is organised into a structured hierarchy for easy exploration, modification, and facilitates onboarding. In the Asset section, separated folders contain scripts, 3D assets, textures, scenes, and prefabs.

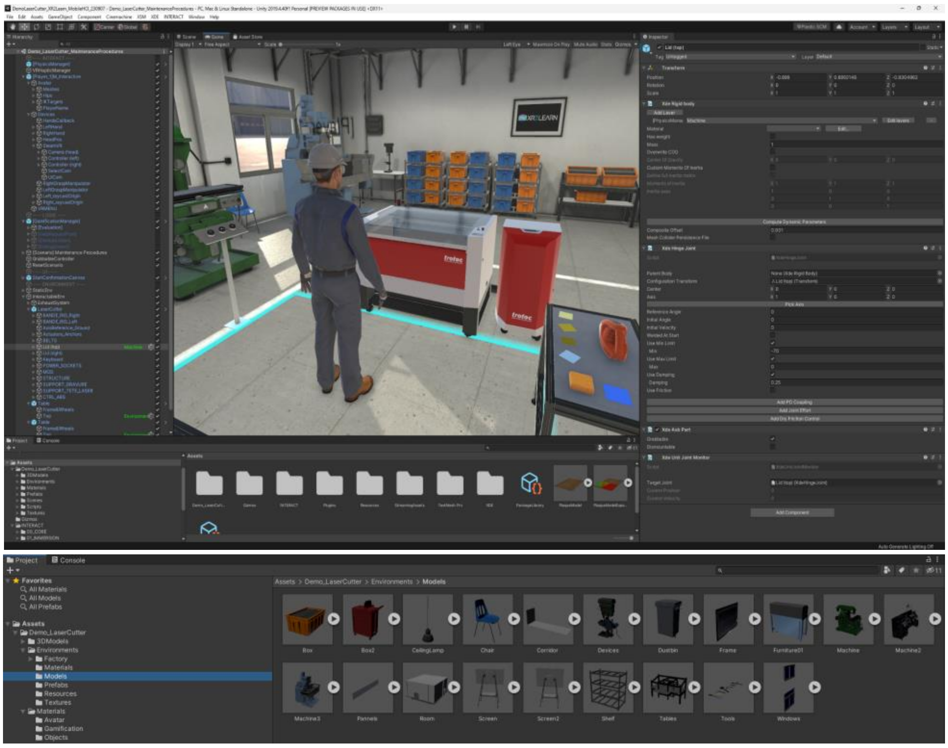


Figure 5: Overview of the Beacon Application 1 Unity project

When opening the source project, a particular attention should be given to the scene structure where physical objects (that can be interacted with) are materialised with a gear icon. Different physical behaviours are illustrated from free moving objects to constrained parts (ex : laser machine top lib with a revolute joint). Intrinsic part properties (mass, center of gravity, friction) are also displayed to fine tune object interaction and reach physical accuracy.

2.4.2 Avatar representation

The avatar's movement is computed from the trainee's tracking data. Basic tracking using only headset and controller's movement is provided through INTERACT. Such basic tracking does not provide the minimum information to reconstruct full upper body configuration. Temporal coherency, inverse kinematics and physical skeleton or biomechanical limitations are used to extrapolate at best the most probable avatar's configuration. Advanced versions with additional hardware requirements (Vive trackers) are available for more accurate full body tracking.

2.4.3 Training scenario and gamification

Using gamification techniques, Beacon application 1 is based on the application of "game design elements in a non-game context" to motivate participants in learning new tasks, improving the memorization of work sequences. INTERACT helps to create such assembly sequences by providing intuitive interfaces to the designer so that creating a VR training scenario does not require advanced skills.

In more details, INTERACT provides the Scenario Graph to create a hierarchy of steps that create an assembly sequence. The user introduces 3D objects and indicates how they are connected through Placing Steps. The user can encode rules to allow the learning scenario to unlock the next steps. For example, the assembly of a wheel starts only if the brake disk is in place AND the bolts have been properly screwed. Several options are available to describe your assembly process in the Scenario window, including time constraints that are required before proceeding to a subsequent step, or interaction with robots and actuators, etc. A scenario can also include Events, that is actions that are only triggered on specific conditions. For example to unweld or activate another part when a keypoint is reached.

Figure 6 provides an example of the scenario graph and the series of steps that make up the assembly process. The Scenario manager automatically handles the visual helpers (trajectories, ghost, instruction panel) when executing the scenario and the transition between steps is done when the part to place reaches its target.

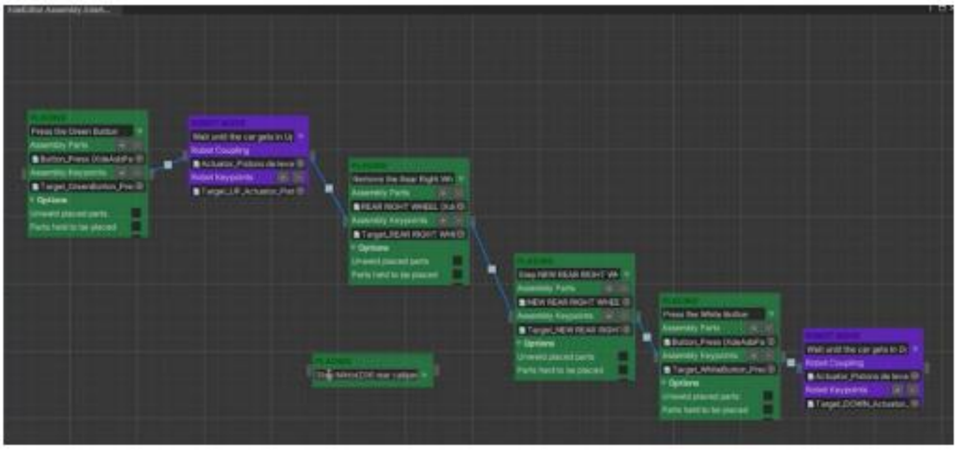


Figure 6: Scenario creation tool with node graph and visual scripting

If needed, the trainee can display visual cues and assistance highlighting the task to perform depending on what the scenario designer has configured for the scenario.

These visual cues can be activated or deactivated during the scenario execution and are composed of :

- A highlight of the active part (the one concerned by the active step)
- A ghost visualisation of the part target (where it should be positioned in the final assembly)
- A guideline showing the trajectory (linking the part current position to the target)

At the end of the scenario, a global score of the exercise is computed and displayed in a dashboard.

Figure 7: Score calculation and goal achievement panel

E. 2.5 INSTRUCTIONS

Below is some useful information on how to install and operate the beacon application.

2.5.1 Prerequisites

1. 2.5.1.1 HARDWARE

- A fairly recent VR-ready laptop or desktop computer running Windows 10/11
- A Meta Quest 2/3 VR headset and its Touch controllers
- (if any other SteamVR compatible headset is used, please disregard each step about Oculus / Meta specific installations)
- An Oculus Link cable (or any compatible USB-C cable with enough bandwidth)

2. 2.5.1.2 SOFTWARE

- Install SteamVR. An offline version (which doesn't require a Steam account) is proposed.
- Install the Oculus Application, and connect with your Oculus / Meta account. The same account should be used on the computer AND on the VR headset.

3. 2.5.1.3 ROOM SPACE

- if possible have a free space of 4x4 m². Less is possible, but less comfortable.

2.5.2 How to install / launch the Demo

- On the computer : launch the "Oculus" application. If not done already, log in with your Oculus / Meta account.
- Plug the Oculus Link cable to your Headset, then to the computer.

- In the VR headset, open the "Quick Settings" by clicking on the clock on the main menu bar
- Then, click on the "Quest Link / Rift" button, then "Start". After a short loading time, you should find yourself in a white virtual environment, with a different menu layout.
- (Back on the computer, if not done already : Unzip "DemoLaserCutter_XR2Learn.zip"
- Then launch Demo XR2Learn Laser.exe
  Put the VR headset back on, and take the Touch controllers. You should be able to see a virtual room with a laser cutting machine, and see the hands of your virtual avatar moving with the controllers.

2.5.3 Controls

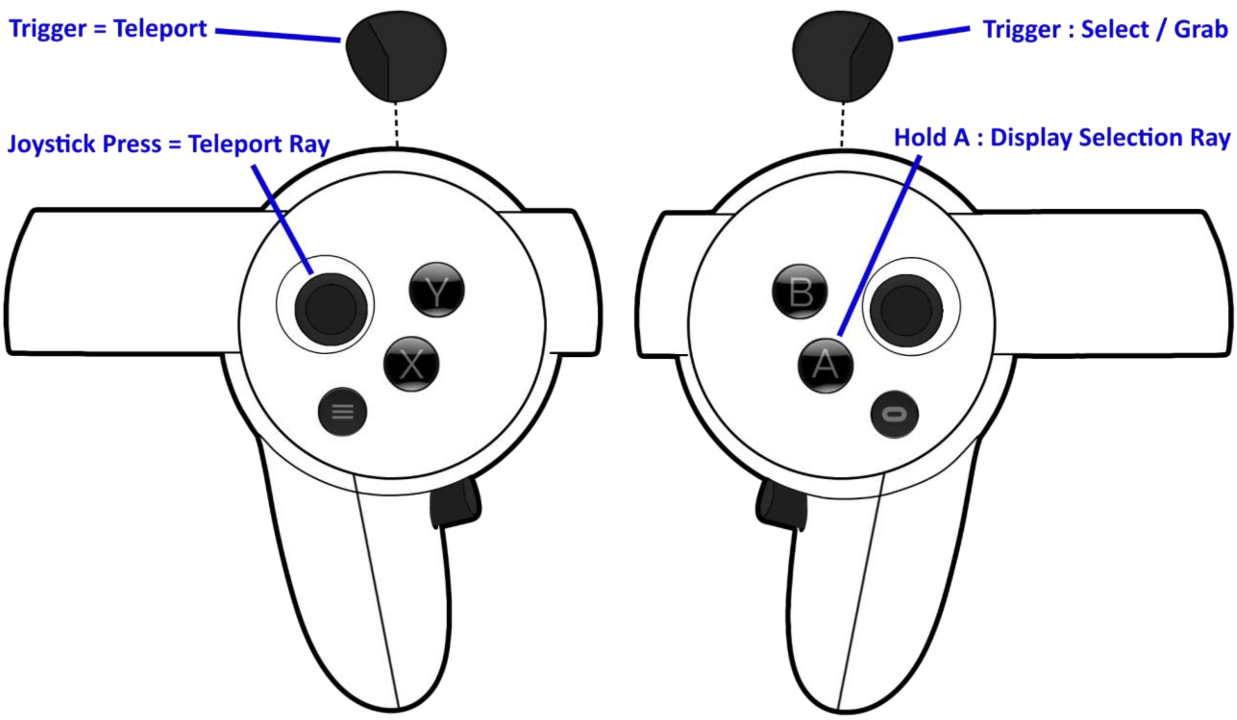


Figure 8: Key mapping of VR controllers for BA1

3. BEACON APPLICATION 2: V-LAB, VIRTUAL LABORATORY

F. 3.1 DESCRIPTION

V-Lab is a VR application development framework for educational scenarios mainly involving scientific processes executed in virtual biology laboratory environments such as chemistry and biology laboratories. V-Lab has been developed by Hellenic Open University. It aims to train students in making use of the lab equipment and conducting virtual experiments at a distance before they use the on-site lab. The goal is to offer a safe environment for the trainees to learn by trial and error and exploration of various possibilities, without the risk of equipment damages and accidents, and therefore reduce the overall cost of the lab training process.

Figure 9: Visuals of the V-Lab environment

G. 3.2 STATE OF THE ART

Interactive computer-based applications for science and biology learning has in the past been developed and tested among students and claimed encouraging learning results[7] [8] [9]. Virtual labs, also known as online laboratories or remote laboratories, have emerged as a powerful tool in the field of education and research. These digital environments provide learners with hands-on experiences and experimental simulations that mirror real-world laboratory settings. By leveraging advancements in technology and connectivity, virtual labs offer numerous benefits such as increased accessibility, scalability, and cost-effectiveness. Such realistic and instructive virtual labs are Labster, developed by the Danish multi-national company of the same name[10], and Learnexx 3D, developed by Solvexx Solutions Ltd, based in the UK[11].

Much research has focused on the development and evaluation of virtual labs across various disciplines; for example, the effectiveness of virtual labs in teaching chemistry concepts to school and undergraduate students has been thoroughly examined. The application learning outcomes of students who used virtual labs has been compared with those who engaged in traditional laboratory sessions. The results demonstrated that virtual labs indeed facilitated effective knowledge acquisition but also promoted deeper understanding and engagement among the learners[12].

Furthermore, virtual labs have also made significant contributions to engineering education. Recently, a virtual lab platform that allowed students to design, simulate, and test electronic circuits remotely was developed. Through a series of experiments, the effectiveness of the platform in improving students' practical skills and problem-solving abilities was evaluated. The results indicated that virtual labs not only provided students with a safe and controlled learning environment but also enabled them to gain valuable hands-on experience, comparable to traditional in-person laboratories[13].

Overall, these recent works highlight the growing interest in virtual labs and their potential to revolutionise education and research. By leveraging technologies such as simulation and virtual reality, virtual labs offer a wide range of benefits, including enhanced learning outcomes, increased accessibility, and improved engagement.

---

[7] George Korakakis, Evangelia A. Pavlatou, John A. Palyvos, and Nicolas Spyrellis. 2009. 3D Visualization Types in Multimedia Applications for Science Learning: A Case Study for 8th Grade Students in Greece. *Comput. Educ.* 52, 2 (February 2009), 390–401. DOI:https://doi.org/10.1016/j.compedu.2008.09.011

[8] Ross Shegog, Melanie M. Lazarus, Nancy G. Murray, Pamela M. Diamond, Nathalie Sessions, and Eva Zsigmond. 2012. Virtual Transgenics: Using a Molecular Biology Simulation to Impact Student Academic Achievement and Attitudes. *Res. Sci. Educ.* 42, 5 (October 2012), 875–890. DOI:https://doi.org/10.1007/s11165-011-9216-7

[9] Leonard A. Annetta, Meng-Tzu Cheng, and Shawn Holmes. 2010. Assessing Twenty-First Century Skills through a Teacher-Created Video Game for High School Biology Students. *Res. Sci. Technol. Educ.* 28, 2 (July 2010), 101–114. DOI:https://doi.org/10.1080/02635141003748358

[10] https://www.labster.com/

[11] http://learnexx.com/

[12] Zeynep Tatli and Alipaşa Ayas. 2010. Virtual laboratory applications in chemistry education. *Procedia - Soc. Behav. Sci.* 9, (January 2010), 938–942. DOI:https://doi.org/10.1016/j.sbspro.2010.12.263

[13] Lely A. Luengas-C, Luis Felipe Wanumen, and Gloria Andrea Cavanzo. 2021. Teaching Electrical Circuits Through the Virtual Lab. *WSEAS Trans. Eng. World* 3, (2021), 81–91.

H. 3.3 CONTENT

V-Lab incorporates high realism with respect to the simulation of lab instruments and the interaction of the user with them. The 3D graphics used are high-end while the simulated experiments are reproduced in a realistic fashion through a user-friendly interface.

The application is available in two versions: (a) a desktop-only version which enables use on traditional, mouse/keyboard systems and, thus, ensure accessibility by a large audience, and (b) a VR-enabled version which enables users to immerse inside the lab environment and interact with the equipment using modern VR equipment and techniques, thus leveraging the benefits of immersion and presence to the educational process and user engagement.

An accompanying SDK focusing on little-to-no-code component-based development of interactive objects completes the V-Lab picture allowing the production of custom spaces and simulation scenarios.

The current version of V-Lab simulates the microscopy procedure. In this, the user needs to make use of the mechanical parts of the photonic microscope (knobs, lenses, etc.) in order to microscope a specimen of frog blood cells.

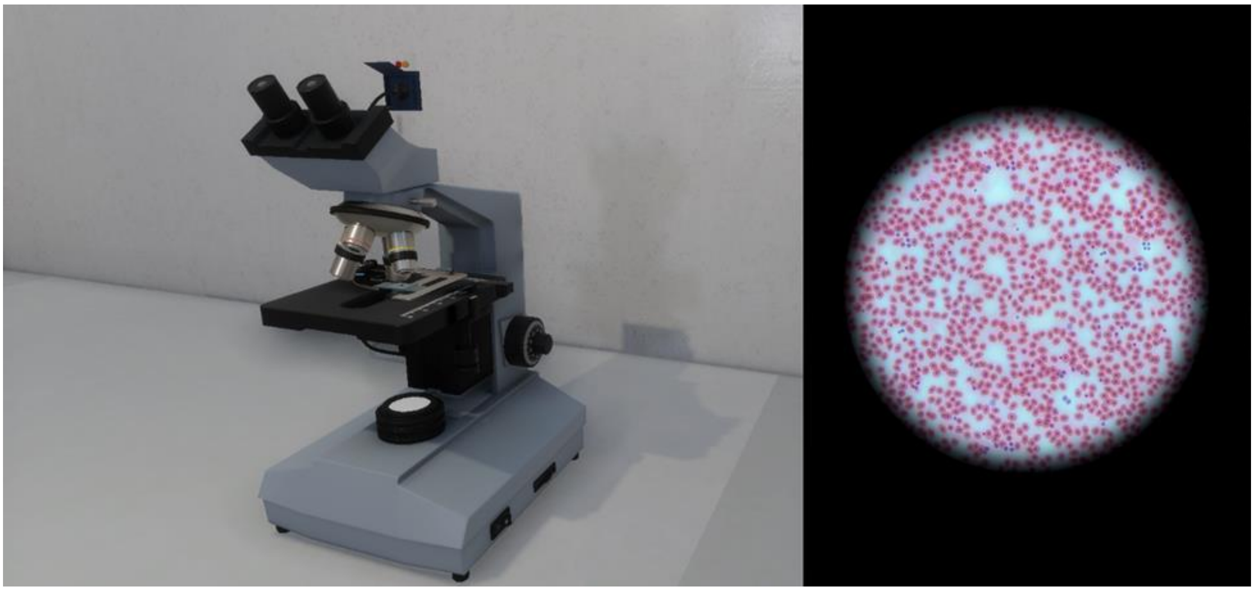


Figure 10: Visualization of the inside view of the microscope

The virtual microscope, exactly like the physical one, contains the following sets of lenses:

- 2 ocular lenses (left and right)
- 1 condenser lens
- 4 objective lenses (with the following magnifications: 4X, 10X, 40X and 100X)

Each time, the user microscopes with an objective lens. The clarity of the microscopy image of the specimen with that particular objective lens placed on the microscope stage depends on the height of the stage (adjusted by the coarse focus and fine focus knobs), the height of the condenser lens (adjusted by the condenser knob) and the position of the 2 ocular lenses (adjusted by rotating themselves).

The procedure can be divided into two main sets of steps: the preparation of the microscope (turning on the switch, adjusting the knobs, etc.) and the actual microscoping with all different objective lenses.

A screen-grab video of V-Lab (desktop version) is shown in the following link:

https://www.youtube.com/watch?v=ULrMzbtZAyE

I. 3.4 ARCHITECTURE

V-Lab has been developed with the Unity game engine.

In V-Lab, the interaction between the user and the environment is similar to the one of adventure games; the user navigates in the lab and uses specific objects alone (presses switches, turns knobs, etc.) or combines two objects with each other (e.g. moving an object onto another one).

The various simulated objects and instrument components in V-Lab are represented by MonoBehaviour classes; that is, classes which derive from a generic Unity class called MonoBehaviour which is attached to the GameObjects of the respective objects. For example, the *PhotonicMicroscope_LightIntensityKnob* is a class for the knob configuring the light intensity in any photonic microscope and is attached in any light intensity knob of microscope of that type. While in Onlabs the *PhotonicMicroscope_LightIntensityKnob* was a separate class, in V-Lab it derives from *LightIntensityKnob* class, a generic class for light intensity knobs of instruments of all kinds, which in turn derives from the Knob generic class. In other words, in V-Lab a broad hierarchy of classes has been designed, which requires a more abstract but also more efficient coding style.

Classes have features in the form of variables; for example, *PhotonicMicroscope_LightIntensityKnob* has a Position of type integer in the 1-24 range. They also have permitted functions on them; for example, the afore-mentioned class has a “turn” function, responsible for the turning of the knob. In V-Lab, the latter function is implemented in a generic form in Knob class, that is, the grandparent class of *PhotonicMicroscope_LightIntensityKnob*, and is inherited to it, while in Onlabs the respective function is implemented locally in the *PhotonicMicroscope_LightIntensityKnob* class. This generic way of function implementation in V-Lab is valid for the functions of other classes as well, with the exception of a few ones which are too specialized and need to be overridden.

The key concept underlying the architecture is the separation of the main simulation engine – the “V-Lab core” component – from scenario-specific assets. The core is available as a self-contained SDK consisting of a set of binary libraries for the Unity game engine, a client API, developer tools and documentation. Among the developer tools provided is a Unity editor tool that (currently) allows for exporting the contents of a scene to a file which can be loaded by the resulting executable at runtime. Thanks to that, V-Lab applications allow for a significant degree of flexibility as the scene setup can be modified and adapted to varying scenarios (in the same context and target area) after distribution without the need for work within the engine’s development environment.

The supported development workflow is as follows:

1. Create a new, empty project in the Unity game engine.
2. Import V-Lab core libraries.
3. Design or import a generic or scenario-specific scene.
4. Import scenario-specific assets into the project, and potentially use them in the Unity scene.
5. Export the overall scene setup, along with additional parameters, as a V-Lab scene file via the editor tool.
6. Develop Behaviour scripts for the scenario-specific assets the V-Lab API and the engine's own APIs.
7. Build executable applications for various target platforms and distribute them along with V-Lab scene files.

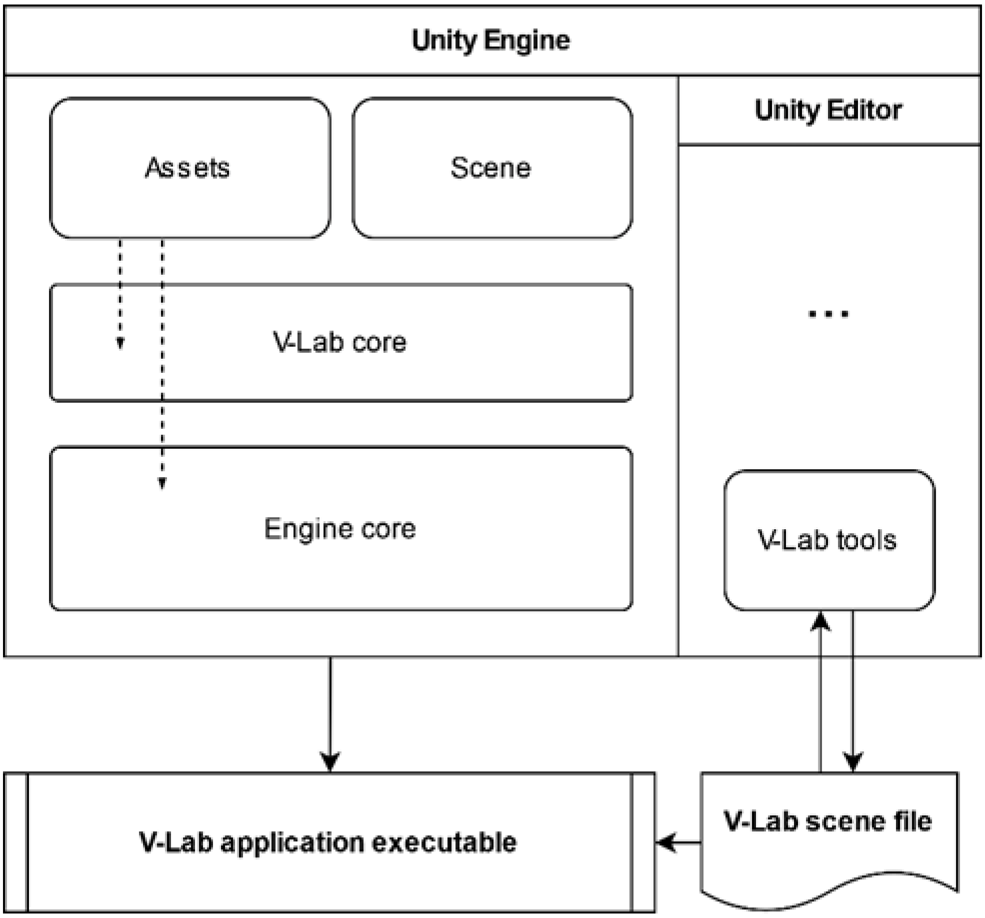


Figure 11: Architecture scheme of the V-Lab Beacon Application

As part of the overall architecture, an additional framework is provided. Thanks to it, the developer can introduce interactivity in a scene in a little-to-no-code fashion, taking advantage of Unity's component-oriented nature. In particular, a set of components can be added to scene elements that allows for the definition of:

2. Points of interactivity, such as switches, knobs, etc.
3. Effectors that apply effects such as spatial transforms, property modification, etc.
4. Variables to connect the above two types of components in an event-driven fashion.

Variables in this framework are first-class objects, not exposed in interactivity or effector components themselves, thus allowing for increased flexibility in expressing interactivity logic and even interfacing with third-party libraries, such as behaviour tree implementations for advanced, behaviour-rich scenarios.

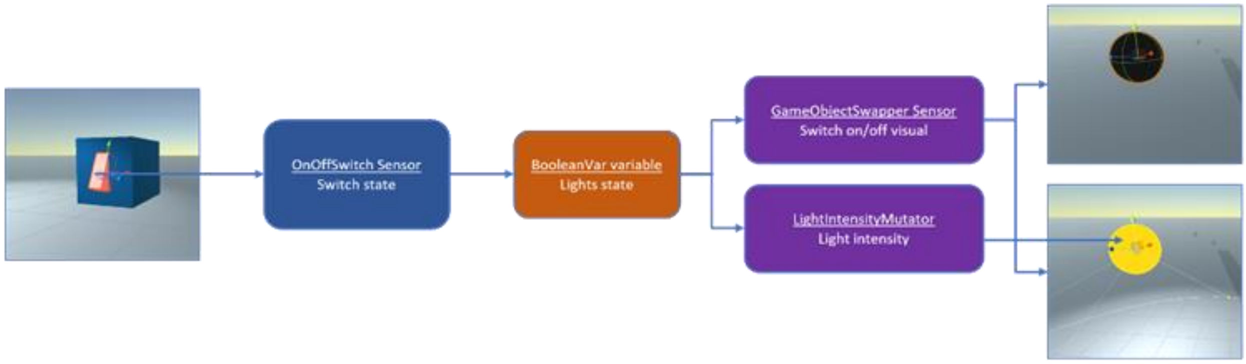


Figure 12: Definition of a light switch using our framework

Figure 12 above illustrates a simple switch/lamp system created using components of the framework.

All application-specific assets, including 3D models, audio and scripts, can be distributed as free and (if applicable) open-source assets, thus complying with the rationale and serving the purposes of XR2Learn.

A. 3.5 INSTRUCTIONS

Below are the instructions for the installation and usage of both desktop and VR versions of V-Lab as well as the instructions for virtually performing the microscopy procedure.

3.5.1 Installation

1. 3.5.1.1 Desktop version

Simply download the latest release from the respective GitHub repository (https://github.com/XR2Learn/Beacon-app-2) and run the executable file.

2. 3.5.1.2 VR version

Before running the VR version please make sure that your OpenXR-compatible VR environment is fully set-up, configured, tested and ready to accommodate execution of desktop VR applications working with XR rigs in tethered mode. This includes the following steps:

- Ensure VR headset is connected to computer via compatible cable.
- Start the installed OpenXR runtime (Oculus app, SteamVR, etc.)
- Ensure proper communication between OpenXR runtime and connected headset and controllers.
- Enable VR headset operation in tethered mode (if needed, e.g., for Meta Quest 2 headsets, start Quest Link on the headset).

After the above steps are complete, download the latest release from the respective GitHub repository (https://github.com/XR2Learn/Beacon-app-2) and run the executable file, just like with the desktop version.

3.5.2 Usage

3. 3.5.2.1 Desktop version

The user navigates with the arrow keys (or WASD keys) and uses the mouse to interact with the various objects and instruments as follows:

- **Left Mouse Click**: turn on/off a switch, focus on an instrument (e.g. look through the ocular lenses of the microscope)
- **Mouse Drag**: move a movable object or rotate a knob
- **Mouse Scroll**: zoom in/out
- **Mouse Move with Control Button pressed**: look upwards/downwards

4. 3.5.1.2 VR version

Interaction with the microscope and other available interaction points is possible at a (reasonable) distance thanks to a ray emitted forward from the user's hand. The ray changes color when it intersects with any point of interactivity (controls). To activate a control, press the controller's trigger button while the ray intersects with the control and then handle the control by performing appropriate gestures depending on the type of the control. Examples are:

- Trigger press/release within a 500 ms period to click on a switch or activate an ocular.
- Move or turn the controller upwards/downwards to rotate a knob.
- Hold trigger and move controller to grab and move an object.

Navigation is possible among predefined locations (anchors) on which the user can "teleport". This helps to minimize potential motion sickness effects, and allows the user to focus on specific points of interest rather than being distracted by the environment or encumbered by furniture or decorative elements while navigating. There are numerous anchors available for both interacting with the lab equipment available and observing the environment from various viewpoints. To teleport, hold the grasp button and point towards an anchor on the floor by means of a curved line emitted from the controller. To facilitate destination selection, anchors are glowing during teleport mode. To make it easier to observe the environment and focus on equipment without having to physically turn your body around you can snap-turn while remaining in position using the controller's joystick.

To bring up the application's menu, press the controller's primary button.

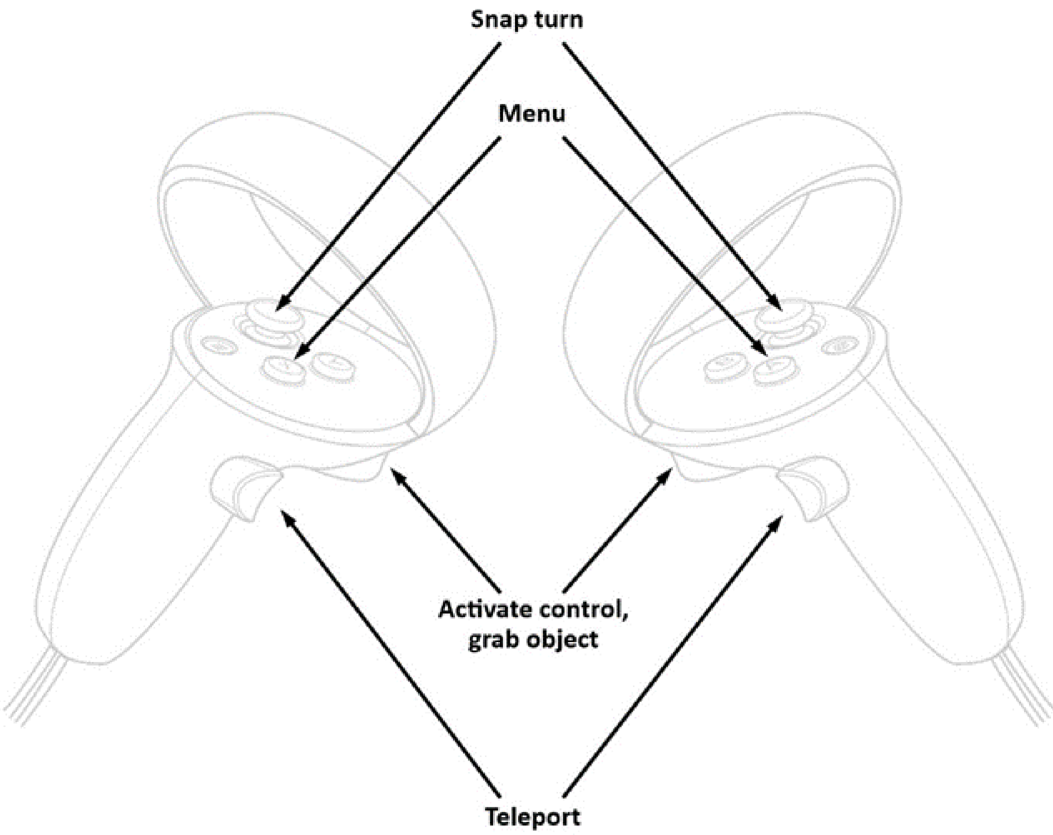


Figure 13: Actions mapping on Meta Quest 2 Touch controllers.

3.5.3 Microscopy process

For the microscopy procedure to be performed correctly, the following steps need to be made:

- Turn the microscope switch on
- Use the light intensity knob to increase the light intensity of the microscope lamp
- Rotate the aperture knob to the left to open the iris diaphragm so the light of the lamp can go through it
- Turn the condenser knob leftwards until the condenser lens reaches its highest position
- Rotate the revolving nosepiece so as to set the objective lens with the lowest magnification (4X) as active
- Put a microscopy specimen in the specimen holder on the microscope stage
- Look through the microscope oculars
- Focus on the microscopy specimen by rotating the coarse focus knob, the fine focus knob and the oculars
- Rotate the revolving nosepiece in order to select the next objective lens in order (10X) as active and focus anew on the specimen
- Repeat the last step for the 40X and 100X objective lenses
- Having finished focusing, stop looking through the oculars, remove the specimen from the microscope stage, close the iris diaphragm by rotating the aperture knob to the right and turn off the switch

4. BEACON APPLICATION 3: INDUSTRIAL EQUIPMENT TRAINING AND SAFETY

B. 4.1 DESCRIPTION

Beacon Application 3 provides a machine-learning methodology based on synthetic datasets from VR industrial environments that can be used for human training in safety procedures and proper machinery handling. The specific application creates synthetic data sets using virtual models of safety equipment namely helmets, protection glasses, gloves, etc. Next, ML models are trained to detect specific objects in actual working environments. The goal of this application is to detect dangerous situations in industry environments and prevent accidents at work either by identifying improper procedures or violations of safety rules. It's tailored to easily create safety training scenarios for professionals, engineers, and students in the industry. In detail, Beacon Application 3 provides a robust detection system to identify hazards in manufacturing processes and enhance the safety management of Industry 5.0. Considering the characteristics of the Fifth Industrial Revolution, the current detection system takes advantage of more sophisticated approaches and digital methodologies, such as game development platforms and advanced AI algorithms, to provide automated, reliable, real-time, and cost-effective safety management capabilities. In addition, the system is flexible and adjustable to meet the requirements imposed by the complexity of the human–machine–environment coupling. Specifically, the synthetic dataset creation methodology provides all the methodologies required to generate data via a VR environment, ensuring that the data can be modified and restructured to simulate new scenarios and environments.

C. 4.2 STATE OF THE ART

The evolution of industrial practices over the past centuries has been marked by successive revolutions, each introducing a new era of production and efficiency. Industry 5.0 is the newest one, aiming to place humans at the heart of production processes. This section presents the various industrial revolutions, concluding on Industry 5.0 that embraces emerging technologies such as AI/ML, VR/XR technologies etc. Moreover, it describes the need for flexible and adaptable safety management methods for Industry 5.0. In this context, the employee's safety insurance in manufacturing is of high importance. The manufacturing sector has historically been characterized by elevated injury rates stemming from the complexities and risks associated with its operations. Specifically, in 2020, the EU reported that manufacturing had the highest number of non-fatal accidents (18.2% of the total) and was the sector with the second-highest number of fatal accidents (14.6% of the total)[14]. By establishing protective measures, employees can concentrate on their tasks rather than being preoccupied with potential risks and hazards in their environment. Studies indicate that employees who perceive their workplace as secure and safe tend to

[14] Eurostat—Statistics Explained Accidents at Work—Statistics by Economic Activity. Available online: https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Accidents_at_work_-_statistics_by_economic_activity#Developments_over_time

exhibit enhanced performance compared to those who feel insecure[15,16]. One fundamental component to ensure human-centered manufacturing towards Industry 5.0 is the establishment of intelligent factory safety management[17]. Conventional approaches to safety management involve employee training, routine inspections, the utilization of warning signs, etc. Those preventive measures hold significant importance for proactively mitigating safety hazards and risks. However, those traditional measures ignore the complexity of human–machine–environment coupling, which requires more sophisticated approaches able to provide automated, reliable, real-time, and cost-effective safety management methods. In this context, advanced technologies can be used to create intelligent factory safety management. This can be achieved by establishing an efficient monitoring and analytics mechanism to identify potential hazards in manufacturing. The system should comprise cameras strategically positioned in the space to capture real-time footage, which is then processed by object detection algorithms trained specifically on application-related data. Real-time detection can trigger alerts to relevant personnel when non-compliance events occur. The robustness and efficiency of the monitoring system heavily depend on the available data for object detection training. However, the complexity of the interaction between humans, machines, and the environment imposes challenges in formulating robust safety management strategies. A significant challenge lies in the adaptability of these safety methods; while one approach can be suitable for a particular case, it might not be relevant or applicable to another[18]. Therefore, Beacon Application 3 provides an innovative safety management approach that is adaptable for exploitation in various manufacturing environments. This is achieved using VR/XR technology to generate accurate datasets tailored to specific applications and scenarios. Initially, high-quality synthetic datasets can be generated via Virtual Environments (VEs) by exploiting the proposed synthetic data generation methodology. These data are artificially produced instead of being acquired from real-world events, which offers numerous notable advantages, including the possibility for adaptability across diverse environments. The methodology that Beacon Application 3 provides defines the steps to create synthetic datasets that can simulate a wide range of conditions, variables, scenarios, and settings, which becomes particularly valuable in environments where a large amount of real-world data acquisition is challenging, time consuming or costly. Afterward, AI object detectors can be trained on the synthetic datasets and used for real-time monitoring based on data streams from cameras.

D. 4.3 CONTENT

Beacon Application 3 provides a flexible and adjustable methodology that can be exploited by factory safety management to detect hazards in real-time. Considering the importance of personal protective equipment (PPE) utilization regarding personnel

[15] Karanikas, N.; Melis, D.J.; Kourousis, K.I. The Balance Between Safety and Productivity and its Relationship with Human Factors and Safety Awareness and Communication in Aircraft Manufacturing. Saf. Health Work 2018, 9, 257–264. https://doi.org/10.1016/j.shaw.2017.09.001.

[16] Shikdar, A.A.; Sawaqed, N.M. Worker productivity, and occupational health and safety issues in selected industries. Comput. Ind. Eng. 2003, 45, 563–572. https://doi.org/10.1016/S0360-8352(03)00074-3

[17] Lu, Y.; Zheng, H.; Chand, S.; Xia, W.; Liu, Z.; Xu, X.; Wang, L.; Qin, Z.; Bao, J. Outlook on human-centric manufacturing towards Industry 5.0. J. Manuf. Syst. 2022, 62, 612–627. https://doi.org/10.1016/j.jmsy.2022.02.001.

[18] Wang, H.; Lv, L.; Li, X.; Li, H.; Leng, J.; Zhang, Y.; Thomson, V.; Liu, G.; Wen, X.; Sun, C.; et al. A safety management approach for Industry 5.0′s human-centered manufacturing based on digital twin. *J. Manuf. Syst.* **2023**, *66*, 1–12. https://doi.org/10.1016/j.jmsy.2022.11.013

safety, the BA3 focuses on the detection of that equipment to enhance industrial safety (). However, the proposed methodology is independent of the use case and can be applied to various scenarios and environments.

The first stage of the developed methodology involves the creation of large-scale annotated datasets using 3D software tools, such as Blender, and a game development platform, like Unity. The various steps to achieve this are detailed, and all the mentioned information can be adjusted. The generated data can be modified and restructured to suit evolving requirements or to simulate new environments, enhancing the adaptability of the proposed methodology.

The second stage concerns the training and evaluation of a ML model that can be deployed on video surveillance systems to identify the target hazards in real-time. The proposed approach has been evaluated through a series of experiments to determine the optimal ratio of synthetic and real data for constituting the training set of object detectors, aiming to achieve the highest possible performance with the minimum number of real-world samples.

Finally, BA3 provides a synthetic dataset of four PPE classes, namely *vest*, *helmet*, *glove*, and *goggles* as well as a trained object detector on this dataset that can be exploited for real-time applications. The software tools and data are available in the XR2Learn's repository in GitHub (https://github.com/XR2Learn/Beacon-app-3).

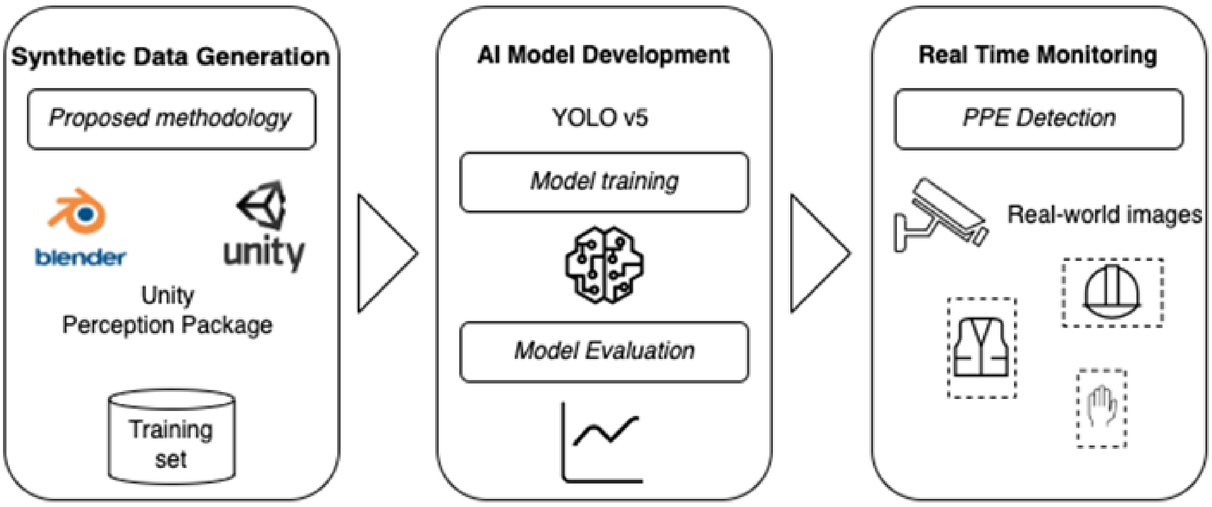


Figure 14: The detection system for real-time hazard detection

E. 4.4 ARCHITECTURE

In this section, we introduce a detailed description of the architectural approach of the innovative methodology designed for the generation of a large-scale annotated synthetic dataset using VR/XR tools. The designed workflow offers a versatile approach to creating annotated data for object detection algorithms without constraints related to object properties or specific fields of application. The result of this generation process is a comprehensive set of RGB synthetic images accompanied by corresponding text files containing the coordinates of bounding boxes outlining objects. The synthetic dataset sets the foundation for training the AI model. Specifically, the proposed detection system includes the methodology for the generation of synthetic datasets as well as the creation of an AI model consisting of model training and evaluation. The workflow comprises two main stages:

1. Synthetic Dataset Creation: The initial stage contains the processes to generate a dataset of synthetic images via a VR enviroment
2. AI Model Creation: The second stage focuses on the definition of the AI model architecture for object detection tasks as well as the training of the model.

4.4.1 Synthetic Dataset Creation

During the first phase of the methodology we focus on the production of a rich and diverse dataset that can cover a wide range of scenarios, making it a valuable asset for a variety of object detection applications. Figure 15 illustrates the proposed methodology.

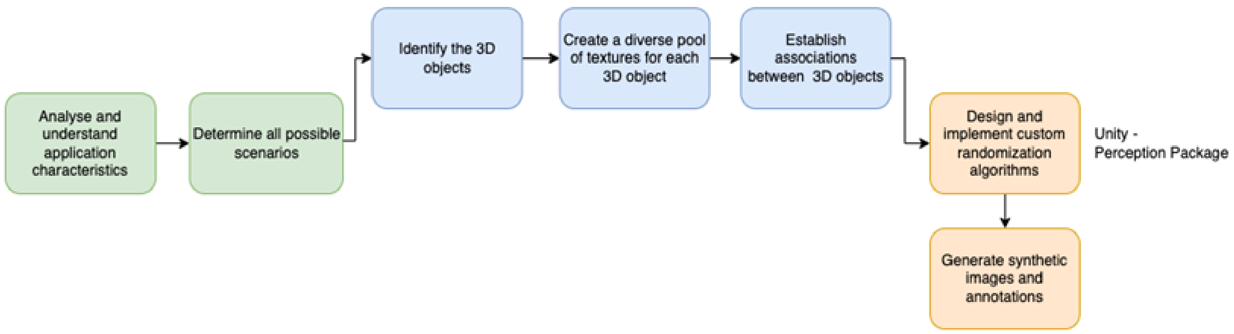


Figure 15: Methodology for synthetic data generation

The first step of the pipeline is the in-depth analysis and understanding of the application characteristics. This step involves a comprehensive analysis of potential variations in geometry, appearance, and utility of the 3D models. This analysis guides the definition of all possible scenarios to be represented within the dataset and provides insights on how parameters such as size, rotation, and hue should be manipulated to achieve this diversity.

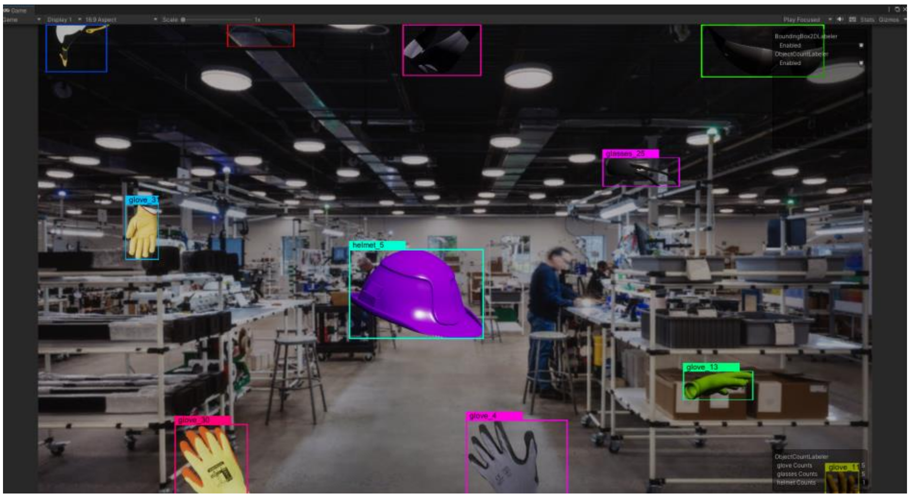
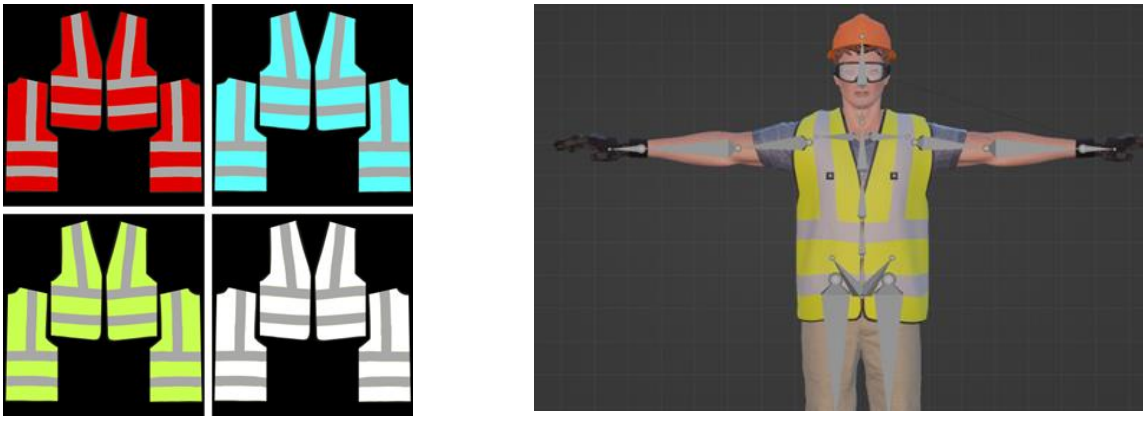


Figure 16: Different 3D models of safety equipment.

Subsequently, considering the purpose of the application, the dataset requirements, and the insights extracted from the previous analysis, the 3D models that will be used for the generation of synthetic datasets are identified and selected. For our case, 3D models that represent the human body as well as personal protective equipment (PPE) are considered, specifically including vests, helmets, gloves, and goggles. Then, leveraging GIMP 2.0[19], we are equipped to create a diverse pool of textures for each object, enhancing the dataset's robustness. Subsequently, we leverage Blender to establish associations between the object of interest and other objects closely linked to it in real-life scenarios, thereby improving the overall lifelikeness of the dataset. Figure 16 shows various texture variations in a 3D object, specifically for the vest (right) as well as the defined association between 3D objects (left).



(a) (b)

Figure 17: (a) Different variations in vest texture; (b) associations between 3D objects in a Blender environment.

After preprocessing the various 3D objects, they are integrated into the Unity 3D gaming engine, where the actual dataset generation unfolds. The dataset generation is performed using Unity Perception[20], which is a set of tools and packages provided by Unity Technologies designed to aid in the creation of synthetic datasets for the development of ML models. It streamlines the critical processes of generating and annotating large amounts of data by simulating various scenarios. Using this toolkit, the user can create diverse and rich datasets by controlling the content of the scene, such as active lighting sources, visible 3D models, and their properties. Moreover, users can design and implement randomization algorithms tailored to their specific requirements. These algorithms serve to depict the desired scenarios while introducing the necessary noise to enhance dataset robustness. In our scenario, multiple custom algorithms are designed and implemented to change the scene's parameters and options during the generation process. Table 1 below presents all custom randomizer functionalities that are used for our case.

[19] GIMP—GNU Image Manipulation Program. Available online: https://www.gimp.org/

[20] Borkman, S.; Crespi, A.; Dhakad, S.; Ganguly, S.; Hogins, J.; Jhang, Y.-C.; Kamalzadeh, M.; Li, B.; Leal, S.; Parisi, P.; et al. Unity Perception: Generate Synthetic Data for Computer Vision. arXiv 2021. http://arxiv.org/abs/2107.04259.

AlgorithmID	Functionality
GridPicker/GridEnabler	Picks/Enables random layout to be displayed
SeatPicker	Picks and enables random human postures
ForegroundObjectRandomizer	Randomly changes rotation and scale parameters of human 3D model
WearablesRandomizer	Picks and enables random 3D PPE on each human 3D model
HueRandomizer	Randomly changes the hue of the 3D object
CustomTextureRandomizer	Randomly changes the texture of the 3D object

Table 1: Custom randomization algorithms and their functionality

To provide a realistic context, we chose to display the 3D objects as being worn by human models, with each model assuming its unique body posture. To ensure diversity, we develop various custom randomization scripts inspired by the principles of perception, enabling us to create eight distinct layouts, each featuring a portion of the final samples. The layouts are grids of fixed positions where 3D models are being placed. In every frame, a human model equipped with safety wearables is randomly selected and placed in each corresponding position of the active layout. The selection of both the human model and the equipment category follows a stochastic pattern. It is noteworthy that each spawn point imposes constraints on the models, governing their size and rotation within well-defined boundaries that respect the objects' geometry and utility. In Figure 17, a layout of 2 and 8 people is shown, respectively. Each individual is depicted in a different body posture and is equipped with some or all of the PPE.

Figure 18: (a) Layout of 2 (left) and 8 (right) employees.

To further increase the robustness of the AI model trained on the synthetic data, we insert background noise into the virtual scene. This takes place further away on the depth axis so as not to restrict foreground objects' visibility. There, we place a 3D plane to surround the camera's viewport area, depicting real-life scenes, and focusing exclusively on industrial spaces to match our objective. The chosen image changes throughout the sampling process, enhancing the noise resistance of the model during training. Figure 18 depicts the unity of the 3D space, showcasing the arrangement of background noise alongside the foreground 3D objects.

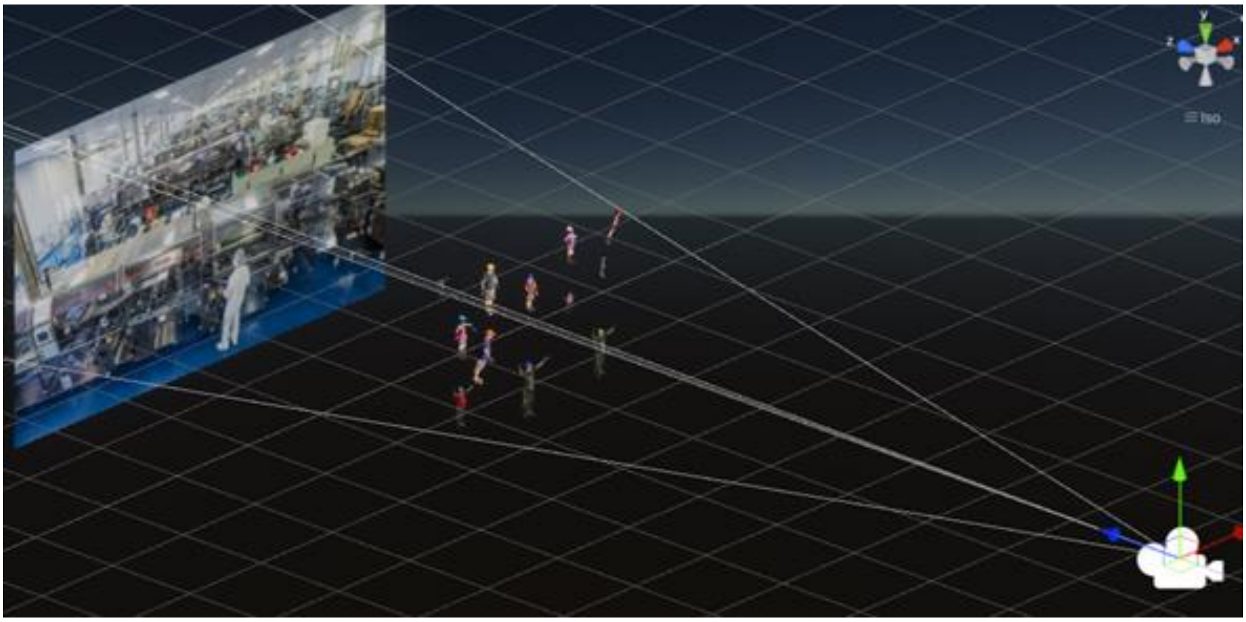


Figure 19: Unity 3D Space. From left to right, a real-life scene as background noise, foreground 3D objects, and the camera's viewpoint with the colored arrows to representing the X, Y, Z axis is 3D space

The ground truth bounding boxes are provided by Perception's labelling method, which allows quick annotation of our 3D models. The created virtual scene is equipped with an orthographic camera able to capture bounding boxes of all objects of interest within its viewport. During the generation process, for every sample, the ecosystem's randomization pipeline applies all randomizing scripts, resulting in a unique frame being sampled. It should be mentioned that the number of generated images, as well as the classes of the dataset, are adjustable and can be chosen by the user. The process resulted in a pool of images, each of them having multiple instances of the foreground 3D models, along with the same amount of JSON files containing the desired annotations for the object of interest. Figure 19 illustrates a synthetic image generated following the proposed methodology.

Figure 20: Synthetic Image

4.4.2 AI Model Creation

The synthetic dataset can be used to create robust object detection algorithms. Depending on the specific requirements and objectives of each application, various ML models can be considered. It is important to select an architecture that aligns well with the needs and performance criteria of the application.

For the implantation of the ML model, the YOLO architecture is chosen as an object detector due to its high accuracy and quick inference speed. Specifically, we focus on the YOLOv5[21] family, which offers various versions of pre-trained models. As the BA3 aimed to produce a trained AI model for real-time applications, we decided to utilize the YOLOv5s. This version represents the "small" variation in this architecture, while it stands out with its optimized design, making it highly efficient and suitable when real-time inference is demanded. YOLOv5 employs the Generalized Intersection over Union (GIoU) loss as a bounding box regression function[22] which solves the inaccurate computation associated with non-overlapping bounding boxes. Rather than initiating the training of the YOLOv5s model from scratch, which demands a large volume of data and considerable computational resources, we utilize the YOLOv5s model pretrained on the COCO dataset[23] and then fine-tune it. By doing so, it is possible to achieve robust object detection capabilities with less data and in fewer epochs. The model is trained for 15 epochs with a batch size set at 32.

The proposed methodology was evaluated in a series of experiments aimed to examine the potential benefits and limitations of combining real and synthetic images for the creation of a training set that can be used to train AI models for real-world applications. To achieve this, we begin with a relatively small number of real images and progressively augment the dataset with synthetic images. The real images are sourced from the CHV dataset, of which 50 are allocated for the training set and 25 for the validation set. In each experiment, we trained the YOLOv5s model on the combined training dataset while we evaluated it on the CHV test set, comprising 133 images, to ensure a consistent benchmark for evaluation.

Table 4 presents the number of real and synthetic images that constitute the training set for each experiment. In the first one, working under the constraints of limited data availability, the model is trained exclusively on a set of 50 real images. In all the following six experiments, the number of real images remains the same while we incrementally increase the number of synthetic images.

[21] Ultralytics YOLOv5: A State-of-the-Art Real-Time Object Detection Systm 2021. Available online: https://docs.ultralytics.com

[22] Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; IEEE: New York, NY, USA, 2019; pp. 658–666.

[23] Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft COCO: Common Objects in Context. In Computer Vision—ECCV 2014; Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2014; Volume 8693, pp. 740–755, ISBN 978-3-319-10601-4.

Experiment ID	Real Images Number	Synthetic Images Number
E1_50_0	50	0
E1_50_50	50	50
E1_50_100	50	100
E1_50_150	50	150
E1_50_300	50	300
E1_50_600	50	600
E1_50_1200	50	1200

Table 2: Number of real-world and synthetic images for each experiment.

YOLOv5s ML model was trained and evaluated in several experiments at the same CHV test set. In the initial experiment, the model, trained with only a small number of real images, achieved a mAP of 14.3%. In the second experiment, the incorporation of synthetic data led to a slight improvement in the model's performance, increasing overall mAP to 16.3%. In the third one, a significant rise in mAP is observed, escalating to 71.2%. The highest mAP value of 84.1% is achieved in the sixth experiment, in which the training set contains 50 real images and 600 synthetic ones. In the last experiment, the number of synthetic images in the training was set to 1200 and it was observed that the mAP value slightly decreased to 81.0%. One explanation for this decrease in performance is the model's over-adaptation to the characteristics of the synthetic data. When exposed to a significant volume of synthetic data, the model develops a bias, becoming particularly adept at recognizing objects in virtual conditions. Consequently, its ability to generalize and recognize objects in real-world scenarios could be compromised. This trend suggests a potential saturation point beyond which adding more synthetic data may not necessarily lead to performance gains and, in fact, could risk the model's efficacy in real-world conditions, leading to diminished returns.

	E1_50_0	E1_50_50	E1_50_100	E1_50_150	E1_50_300	E1_50_600	E1_50_1200
mAP	14.3%	16.3%	17.3%	71.2%	79.5%	84.1%	81.0%

Table 3: mAP values of object detector for each experiment.

To ensure the robustness and reliability of our findings, the experiment was executed multiple times, each time selecting a different random set of 75 images from the CHV dataset. The methodology remains consistent. Specifically, we begin with 50 real images and then gradually introduce synthetic data to observe the impact on performance. As illustrated in Figure 21, the performance of the models showcases similar trends across all experiments, validating the authenticity of our initial results. It should be mentioned that the best performance of all models is observed when combining 50 real training images with 600 synthetic images. This configuration yields an average mAP score of 84.3%, with a standard deviation of 0.4%.

Remarkably, by employing a training set comprising just 50 real images combined with synthetic data to train an object detector, we manage to bridge the performance gap, achieving a result that is only a slight 4.7% behind the scenario of having only real images.

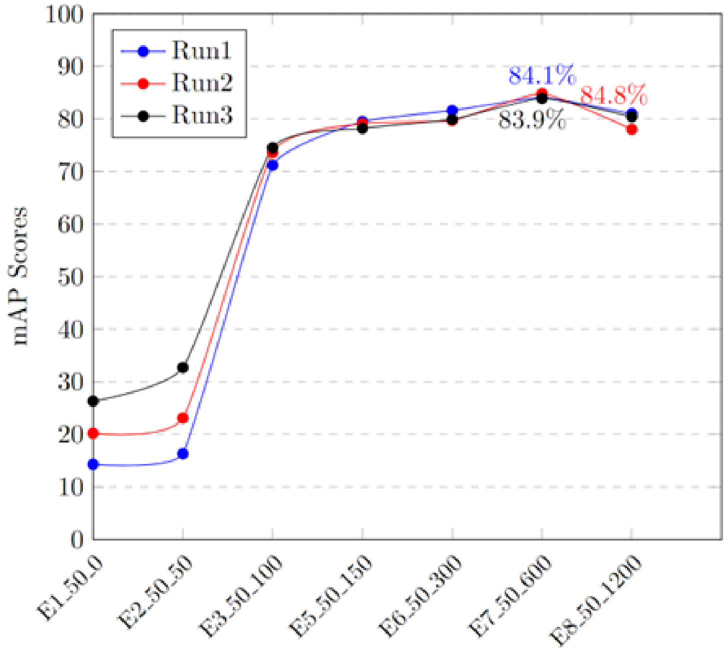


Figure 21: Model's performance across different ratios of real-world and synthetic images.

F. 4.5 INSTRUCTIONS

The latest release of the Beacon Application 3 with all the complete documentation and technical description is available on the respective GitHub repository (https://github.com/XR2Learn/Beacon-app-3) and the technical wiki.

4.5.1 Dependencies

Beacon Application 3 is entirely based on the Unity Perception Package that allows for quick sampling and labelling objects directly through the Unity Platform and the CHV dataset into a format compatible with the YOLOv5 model.

4.5.2 Quick Start

The following steps provide a proof of concept tutorial that demonstrates the potential of achieving excellent results by combining a small subset of real images with a larger synthetic dataset. The subsequent steps are aimed at preparing the dataset for training an object detection algorithm, specifically designed to recognize and classify person, vest, and helmet categories.

1. 4.5.2.1 Synthetic Dataset Preparation

To effectively train an object detection system using the YOLOv5 model, the dataset should include both the images and corresponding text files that provide annotations for the objects present in these images. For detailed insights into the data structure and annotation format, please refer to the information available at this link.

The synthetic dataset includes labels for various classes as listed below:

```
1: gloves
2: glasses
3: helmet
4: vest
5: person
```

Annotations labeled with 1 correspond to gloves, those with label 2 represent glasses, and so on for the other categories.

First extract the content of the rar file of the synthetic dataset:

```
unrar x <name_of_rar_file>.rar
```

This guide focuses on training an object detection algorithm to identify a person, a vest, and a helmet. Therefore, we will remove the classes for gloves and glasses from the dataset:

```
python utils/delete_lines.py /path/to/synthetic_dataset 1 2
```

Last by running the following script all the folders of the images and the labels will be organized in the necessary format:

```
./create_synthetic_dataset.sh
```

Upon executing the script, you will be prompted with the following message: **"Enter full path of the synthetic dataset:"**

Enter the complete path to the synthetic dataset to initiate the preprocessing process. After running the bash script, the labels corresponding to each class will be updated as follows:

```
0: person
1: vest
2: helmet
```

All the necessary preprocessing for the synthetic dataset is done.

2. Important note: If your intention is to train on different classes, such as including all classes or just two classes like gloves and person, please be aware that this will require adjustments to both the files provided in the "utils" folder and the bash scripts.

3. 4.5.2.2 CHV Dataset Preparation

The Color Helmet and Vest (CHV) dataset is a dataset that contains real rgb images tailored for the detection of individuals wearing vests and helmets in four distinct colors.

Here are the labels and their corresponding classes:

```
0: person
1: vest
2: helmet blue color
3: helmet yellow color
4: helmet white color
5: helmet red color
```

First download the dataset by following this link
Once downloaded, execute the provided bash script to prepare the dataset:

```
./create_chv_dataset.sh
```

Following the script's execution, you will need to input both the full path to the CHV dataset and the location where the processed dataset should be saved. In the location to save the dataset you also need to add the name of the saving folder at the end of the path insertion. For example: **"path/to/save/the/dataset/name_of_the_dataset"**

After running the bash script, the labels corresponding to each class will be updated as follows:

```
0: person
1: vest
2: helmet
```

All the necessary preprocessing for the CHV dataset is done!!!

4. 4.5.2.3 Combine CHV and Synthetic Dataset

To combine the datasets run the following bash script:

```
./create_combined_dataset.sh
```

The script will ask the full path of the synthetic dataset and the full path of the CHV dataset. Upon running the bash script, a new folder named **combined_dataset** will be generated. Inside this folder, you will find:

- A test dataset consisting exclusively of 133 real images from the CHV dataset. This selection ensures testing of the model's real-world applicability.
- In the train folder, there will be a total of 50 images from the CHV dataset and 600 images from the synthetic dataset, facilitating comprehensive training.
- The validation dataset will encompass 25 images exclusively from the CHV dataset, ensuring a diverse and balanced dataset for model evaluation.

If you want to use different number of images of each dataset you must change the values (*NUM_OF_REAL_IMAGES*, *NUM_OF_VALIDATION_IMAGES*, *NUM_OF_SYNTHETIC_IMAGES*) in the **create_combined_dataset.sh** file.

The dataset is complete and ready for training!!!

5. 4.5.2.4 Training process

To train the YOLOv5 model, follow these steps:

a. Clone the official repository and install the required packages:

```
git clone https://github.com/ultralytics/yolov5  # clone
cd yolov5
pip install -r requirements.txt  # install
```

b. Navigate to the 'data' folder and create a YAML configuration file:

```
cd data
touch syn_chv.yaml
```

c. Open the yaml and add the following content:

```
path: path/to/dataset # add the full path of the dataset
train: images/train
val: images/test
test: images/test

# Classes
names:
  0 : person
  1 : vest
  2 : helmet
```

d. Start training the model by running the following commands:

```
cd ..
python3 train.py --epochs 10 --data syn_chv.yaml --weights yolov5s.pt
```

That's all !! The training process is finished.

6. 4.5.2.5 Live Detection

Utilizing the trained model, you can now create a real-time detection to check whether a person is equipped with helmets or vests. This process simply requires a camera to be operational.

To initiate live inference, execute the following commands:

```
cd ..

mv hazard_detection.py path/of/yolov5_repo

cd path/of/yolov5_repo

python hazard_detection.py --weights /path/to/weights/best.pt --source 0 --conf-thres 0.70
```

In the event that the camera fails to recognize the subject, consider altering the **--source** parameter to a different value, such as 1 or 2.

The hazard_detection.py script is designed to identify individuals who are either wearing or not wearing the necessary safety gear. In cases where the appropriate gear is missing, an alert will be displayed in the terminal.

Figure 22: Safety equipment identification in real time

5. CONCLUSION

The purpose of this document is to accompany the three Beacon Applications by providing valuable information on how to install, configure and use each one of them. The three Beacon Applications provide a starting point for future XR2Learn users and creators. Since the purpose of the document is to accompany the software, we here emphasized on a selected set of use cases and interaction examples to guide newcomers with the best practices for creating and consuming immersive experiences as a learning tool in various industries.

During the lifetime of the XR2Learn project, these Beacon Application will also guide other contributors to make the XR2Learn platform content grow with new use cases, new supported platforms, and hopefully reach the interest of new industrials that didn't think immersive technologies could be helpful, affordable or approachable so far.

This project is hopefully the starting point of a larger democratization of immersive technologies for Industries 5.0, with all the safety, economical, and learning benefits they have proved to be bringing over the past decades.