

Design and Evaluation of a Choreography-Based Virtual Reality Authoring Tool for Experiential Learning in Industrial Training

Fernando Cassola, Daniel Mendes, *Member, IEEE*, Manuel Pinto, Leonel Morgado, Sara Costa, Luís Anjos, David Marques, Filipe Rosa, Ana Maia, Helga Tavares, António Coelho, and Hugo Paredes

Abstract—The use of virtual reality (VR) for industrial training helps minimize risks and costs by allowing more frequent and varied use of experiential learning activities, leading to active and improved learning. However, creating VR training experiences is costly and time-consuming, requiring software development experts. Additionally, current authoring tools lack integration with existing data and are desktop-oriented, which detach the pedagogic process of creating the immersive experience from experiencing it in a situated context. We present a novel interactive approach for immersive authoring of VR-based experiential training by the trainers themselves, from inside the virtual environment and without the support of development experts. The design includes identifying interactable elements, such as 3-D models, equipment, tools, settings, and environment. The trainer also specifies by demonstration the actions to be performed by trainees, as a virtual choreography. During course execution, trainees' activities are also registered as virtual choreographies and matched to those specified by the trainer. Thus trainer and trainee are culturally situated within their area semantics and social discourse, rather than adopting concepts of the VR system for the learning content. We conducted a usability case study with professionals from an international wind energy company, using detailed models of wind turbines and real-world procedures. Trainers set up a training course using the immersive authoring tool, and trainees executed the course. The learning experience and usability were analyzed, and the training was certified by comparing real-world task completion between a user that had undergone virtual training and a user that did not.

Index Terms—Virtual reality, industrial training, experiential learning, authoring, virtual choreography.

1 INTRODUCTION

IMMERSIVE environments have long been an enticing way to improve industry training, decoupling training from physical resources and risks, offering the promises of cost minimization and increased safety [1]. Further, by mirroring the physical environment onto the virtual one, multiple experts and learners can remotely inspect status and provide feedback or intervene. Simulation and digital twins are thus approaches supporting situated and experiential learning, both for logistics uses and pedagogic transformation [2]. Areas of application besides work safety [3] include sports coaching training [4], cooperation in aircraft mechanical maintenance [5], [6], and mining [7]. We follow the current that defines experiential learning or interactive learning as the well-known term “learning by doing,” which has its roots in the research of works of Dewey [8], Lewin [9], and Piaget [10]. This is because there are several authors,

who indicate that, when the student plays an active role in the learning process, the learning is highly increased [11]. We can then consider that the Experiential Learning model corresponds to a cyclical process of learning experiences, defined in four stages (concrete experience, abstract conceptualization, reflective observation, and also active experimentation) [12]. This derives from the concept of embodied cognition, which acknowledges that our bodies are in causal coupling with our surroundings, leading to a development and operational organization of our minds as “an emergent property of interactions among internal and external resources” [13]. A corollary of this for industrial training is that the tools, parts, and other materials employed in the learning content are part of this cognitive structure, “integrated into the way people think, see, and control activities, part of the distributed system of cognitive control” [13].

And yet, VR training experiences are not commonplace. Their creation involves significant budgets and time allocation, using professional development tools such as Unity [14] or Unreal Engine [15], requiring software development experts [16]. This is hampering the adoption of experiential and situated learning immersive experiences. Further, the involvement of software developers as experts outside the trainer and trainee content area, risks shifting the

- F. Cassola, D. Mendes, and A. Coelho are with INESC TEC and Faculty of Engineering of the University of Porto, Portugal.
E-mail: fernando.c.marques@inesctec.pt.
- M. Pinto and H. Paredes are with INESC TEC and University of Trás-os-Montes e Alto Douro, Portugal.
- L. Morgado is with INESC TEC and Universidade Aberta, Portugal.
- S. Costa, L. Anjos, D. Marques, F. Rosa, A. Maia, and H. Tavares are with Vestas Wind Systems, Portugal.

Manuscript received June 16, 2021; revised February 16, 2022.

training concepts to different cultural and social contexts, rather than keeping the industrial training socially and culturally situated. One approach to lessen this is empowering trainers and instructors with end-user authoring tools for VR (e.g., EON Reality [17] and XVR Simulation [18]). Such tools have limitations such as being unable to create elements from data models that organizations already possess, e.g., engineering CAD models. They are also non-immersive, due to their professional development orientation, which involves many fine adjustments and high levels of detail. This detaches the authoring processes from the immersive experience, thus trainers while developing procedures in professional development tools are not entirely aware of the perspective that trainees will face when immersed. Thus, the non-immersive situated experience of the trainer creating the training is cognitively excised from the immersive situated experience of trainee learning activities.

The work herein presents a way to address the current limitations by empowering trainers and instructors to perform immersive authoring of immersive training, thus pairing the situated, embodied experiences of trainer and trainee. We also enable trainers to create the training activities without the intervention of software developers, and activity records generated by trainees are directly matched to trainers' specification. This enables the authoring and the training analysis to remain situated within the cultural and social background of the content area community.

Within an immersive environment, digital models are provided from pre-existing organizational data: engineering CAD models of wind turbines of Vestas Wind Systems Company [19]. They thus look authentic for training courses on wind turbine maintenance, supporting experiential learning. After setting up the foundations for a course, the trainer demonstrates the interactions specified for trainees, inside the immersive environment. Instead of being recorded as interactions with software elements (meshes, coordinates), the interactions are identified as virtual choreographies within the semantic domain of the training course. When the trainees later act in the immersive environment, their actions are also interpreted as virtual choreographies, and at the same semantic level, thus remaining culturally and socially situated within the learning community. This enables matching trainer's actions to trainee's actions, even if the actual environment changed the underlying technology. For example, removing a cable is checked as such, not as specific textures/meshes/coordinate systems. Trainers are thus free to generate diversified, rich scenarios and interactions, without risking their technological obsolescence. Trainees can then experience those experiential and situated learning scenarios. Industry areas of application include manufacturing, not only for training but also for certification: analytics can be provided and matched for trainees' choreographies that depend only on the semantic training level, not the software development aspects.

The situated nature of our immersive authoring approach thus operates at embodied, cultural, and social dimensions of situated distributed cognition. This enables sociocultural scaffolding of workplace learning, rather than the mere provision of learning activities from trainer to trainees. Trainers can interpret trainees' actions in the immersive environment and readily revise, re-purpose or cre-

ate new contexts and training activities, to develop varied strategies, such as patterns of individual vs. collective learning interaction [20]: request help to solve work problems, share knowledge to initiate discussions, co-create solutions, adapt knowledge to the local needs, or formalize knowledge as a shared activity. For instance, a trainer in the workplace typically assembles over years of experience a collection of examples of procedure-execution approaches that have shown to be risky or even potentially harmful. Since immersive authoring can be done directly by the trainer, without involving software developers, these examples can readily be rendered as virtual experiences for sharing this knowledge. Similarly, trainees can also employ immersive authoring to express viewpoints in distributed, situated context, rather than being limited to verbalizing them. They can jointly create and develop new approaches to procedure execution in the same fashion, and the final knowledge is not merely a set of bullet items, but a virtual, immersive experience, which can be appropriated in embodied, culturally, and socially situated ways to create awareness, build shared understanding, adapt a solution to a new situation, or validate new solutions. That is to say, the scaffolding enabled by this approach leads to the appropriation of knowledge by experiencing it and ultimately to maturation [20].

We tested our approach through a usability case study with expert professionals, whom experienced both trainee and trainer perspectives, with 3-D models of authentic wind turbines and real-world maintenance procedures. As such, the main contributions of the work presented in this paper are:

- 1) A novel immersive approach for authoring VR training courses. Although the developed tool was applied to the specific scenario of wind turbine maintenance, the proposed approach is designed in such a way that it can be applied to other scenarios that require operations on 3-D content;
- 2) The use of virtual choreographies in experiential learning. Virtual choreographies can be used to abstract interactions from computational concepts and exact values, such as time and position, by storing and analyzing actions at a semantic level. This provides support for more straightforward and flexible analysis and certification of trainees' performance;
- 3) A usability case study with expert professionals to test the proposed approach. Gathering participants with extensive field knowledge and training experience allowed us to collect valuable feedback. Additionally, the use of real models and procedures ensured the correct assessment of the applicability of our approach.

This paper is an expanded version of previous publications [21], [22], where we introduced the immersive authoring tool. Here, we provide a better contextualization of the work according to the state of the art, provide additional details about our approach, explain how we use virtual choreographies, and report a usability case study with professionals.

The remainder of this document is structured as follows. Section 2 contextualizes our work regarding the state-of-the-

art in immersive training and virtual choreographies. Then, in Section 3 we present our tool for creating experiential learning courses in VR. Section 4 details on the use of virtual choreographies for storing and matching trainers and trainees actions. We follow, in Section 5, with a description of the usability case study guided through three steps (course authoring, learning experience, and certification), that was conducted to test our approach, including the attained results and its discussion. Finally, in Section 6 we conclude the paper and point out directions for future research.

2 RELATED WORK

We based this work upon research on immersive learning, virtual choreographies, and virtual training authoring, which are presented in this section.

2.1 Immersive Training in Virtual Reality

A recent survey on uses of immersive learning environment has identified six clusters of uses reported in the literature: complementing (usages for emphasis, multimodal interaction, complement/combine contexts, media or items), simulating (reproducing the physical world, or leveraging logistic affordances), exploring (data collection, and interactive manipulation and exploration), engaging (skill training, engagement, and collaboration), experiencing (for augmented context, emotional and cultural experiences, or changing human behavior), and accessing (providing accessibility, perspective switching, or enabling seeing the invisible) [2].

These uses reflect a wide perspective on the learning activities that can be carried out in these environments, and on the diversities of associated purposes. For instance, reducing costs and risks, which are commonly referenced advantages ([3], [23]), can be seen as examples of such uses: logistics (reducing costs) and simulating the physical world (reducing risks). One can similarly situate in these clusters other commonly referenced advantages. For instance, more flexibility scheduling training sessions is also a logistics use, whereas engaging by simplifying the complexity of an action to match a learner's current abilities is a case of skill training use. Engagement-oriented approaches are also common, using immersive environments to trigger emotions seeking to further the interest of learners [24].

This diversity of uses is seen as an enabler of experiential and situated learning, allowing trainees to repeat simulated circumstances, supporting self-regulated learning, and also trainer reviewing of the trainees actions [25], [26]. For example, Çakiroğlu & Gökoğlu [27] created a virtual reality environment for basic fire safety training of primary school children, reporting significant improvement of fire safety behavioral skills, which were transferred to physical environments. Lee et al. [28] conducted a meta-analysis on virtual reality training for limb function in stroke patients, demonstrating moderate effectiveness in improving the function in these patients. In mechanical training, there are numerous examples. For instance, Im et al. [6] employed a VR engine to simulate assembly/disassembly procedures, reporting elevated levels of interest and learning outcomes;

other instances include team procedures and team coordination for aircraft engine assembly [5] or safety training for mining operations [7].

In the specific case of this work, our focus was primarily on the "Simulating" cluster of uses of immersive learning environments (simulate & logistics). We sought to simulate the physical world for fidelity of situated learning, by the CAD files of the components. We also sought to increase the availability of components to increase training frequently, that is, a logistics use. Further, we leveraged the "Engaging" cluster, by enabling trainers to author the specific sequence procedures that trainees would follow.

2.2 Virtual Training Authoring

To create new virtual learning experiences, the most common approach is to resort to tools designed for traditional desktop setups, even for immersive training, as demonstrated by existing firefighters [29] and surgeons [30] training tools, for example. While these can leverage on previous knowledge, they do not properly convey the experience trainees will have. For an accurate assessment of the result, trainers need to deploy the experience in a VR setup and test it. Then, to perform eventual modifications, they must leave the immersive setup and return to the desktop environment. This cycle needs to be repeated until the desired result is achieved. Also, in a recent study, González et al. [31] compared a training authoring tool using a desktop and an augmented reality-based environment. Although the results showed no significant differences, they concluded that the desktop interface proved to be more efficient.

To prevent the back and forth between a desktop authoring tool and the evaluation of the result in VR, the training experience can be directly authored in the immersive environment. Eroglu et al. [32] proposed an immersive road network authoring system, targeted at automated vehicle testing. As such, it focuses solely on the environment creation, allowing users to select and manipulate objects, design road networks, and define traffic direction, for example.

Coelho et al. [33] developed an immersive authoring tool for multi-sensory VR experiences. It allows users to configure visual, audio, haptic, and olfactory output, being possible to specify their timings and duration, for instance. Their results "emphasize that when authoring immersive content, an immersive interface should be available, as it provides a better overview and perception of the contents that are being authored" [33].

The previous examples revealed advantages of using immersive approaches for authoring immersive content. However, they do not target training activities. One great possibility of learning experiences in VR is the ability to observe procedures as if they were being performed in front of the user. Wolfartsberger and Niedermayr [34] proposed an "authoring-by-doing" approach to allow trainees to watch previously recorded actions performed by experts, as semi-transparent animations, and mimic those actions. We, however, intend to go further, allowing trainers to create entire courses in an immersive environment. For this, it needs to be possible to specify multiple elements, such as tools, tasks, and actions, allow trainees to observe trainers performing procedures, practice and test those procedures,

and be able to assess trainees performance. Here, the ability to match trainers and trainees actions is of utter importance.

2.3 Virtual Choreographies

Human behavior is a critical element in society. The scientific field that studies a broader understanding of human behavior is Psychology [35]. However, this work does not aim to study this area of knowledge, namely the theories and types of behavior, but rather how a virtual reality platform can be used as an authoring tool to create experiential learning sessions and evaluate whether users are performing the intended behaviors within the platform.

Representing human behavior from an information systems perspective emerged after the culmination of the cold war, around 1990, when new challenges for NATO emerged [36]. All because of the advent of innovative technologies that were beginning to have a tremendous impact on implementing simulation systems and decision support tools. It is then that the digital representation of human behaviors becomes vital to power decision support tools and simulators [36].

There are several references to model human behaviors, such as Human Behavior Representation (HBR) framework [36], [37], PECS (Physical conditions, Emotional state, Cognitive capabilities, Social status) [38], BDI (Belief-Desire-Intention) model [39]. However, ontologies allow a greater and more detailed knowledge at the level of the representation of behaviors [40].

Virtual choreographies are sets of actions, associated with any active element of an environment (e.g., people, machines, states). They can be predefined or inferred from examples, and offer a means to provide an understanding of those actions within a wider context of meaning, that is, a higher semantic level [41]. The motivation for them was overcoming the rapid obsolescence of recorded or predefined actions when they are tied to a technological platform, such as a specific immersive environment [42]. The relevance of this concern is the diversity of virtual systems where several users collaborate: in such environments, the meaning of actions is more complex and diverse than the mere spatial coordinates and timestamps, and the interactions have meaning beyond mere textures or surfaces: mechanical maintenance [5], factory assembly [43], disaster training [44], among others. Other areas, such as scientific experiments, also have to analyze complex behavioral contexts [45]. Virtual choreographies enable hypotheses testing, supporting empirical check and validation of phenomena, for instance, in gaming contexts [46], [47].

By converting data about specific actions into a higher semantic concept, virtual choreographies, one can analyze the purpose and the wider context - aspects not readily derived from raw data such as coordinates or timestamps. For instance, virtual choreographies were used in the analysis of human behaviors in an office space, contributing to their understanding in the scope of energy consumption impact and potential for behavior change [48]. Here, we generate choreographies from trainers' exemplified actions, and match them in real-time to trainees actions, to determine which choreographies have been pursued and can be certified. Potentially, that matching could also be done

to known undesirable behaviors, to trigger warnings and advice.

This matching between different virtual choreographies, based on purpose and context, rather than low-level data, is made possible due to the platform-independent nature of virtual choreographies, which are based on the semantics of the knowledge domain not on the semantics of the technological platform [41]. Thus, choreographies only need to be associated with a platform in order to be reproduced visually, not for matching to others [41]. Platform features are also relevant when one seeks to infer choreographies from low-level platform actions, but not after that inference. This detachment between choreographies and technical platform holds the potential to generalize choreographies and transfer them to other fields of knowledge, for instance, process management [45].

The virtual choreographies approach employed Model-Driven Engineering principles [41], [45]: semantic representations are transformed automatically from the training domain into the platform-specific domain, rather than ad hoc. Techniques for this include ontology mapping [41], formal models [45], transformation languages [49], partial order planning [50] or hierarchical state machines [5].

In the present work, we have followed a mixed approach. First, we made an ontology mapping [41] to identify the set of users' behaviors, then we have created a hierarchical state machine [5] to infer them based on the performed user actions. This was based on the representational approach of Demetrius et al., based on xAPI. [49]. In summary, the use of virtual choreographies was necessary for training and evaluation of the behaviors performed by the users. In that sense, initially, these serve to identify which behaviors are necessary to perform a given task, subsequently, they are used to teach the learners, and finally allow to validate that learning.

3 IMMERSIVE AUTHORING TOOL

We have developed a VR authoring tool enabling trainers to design training courses in immersive environments. This tool is meant to be applicable to various areas of engineering, by focusing on the demonstration of training procedures upon components in general. Our current prototype supports training on maintenance procedures upon wind turbines. First, it is necessary that the trainer, inside the immersive tool, makes the configuration of the whole course, from the point of view of its structure. Namely, following a certain data structure, and according to the technical procedures manual of that course, it will be inserted the necessary information for the best contextualization of the course (code, name, description, etc.). Then it will be necessary to make the initial setup, associating all the technical contents (contents related to the model and 3-D environment) to that course. Subsequently, it is then required that the trainer, always within the immersive environment, performs a set of actions, following the procedures manual for that training session, and thus allowing the platform to detect and record which sequential behaviors (see section IV - choreographies) should be performed later by students in order to achieve success in that training module. We will now describe in

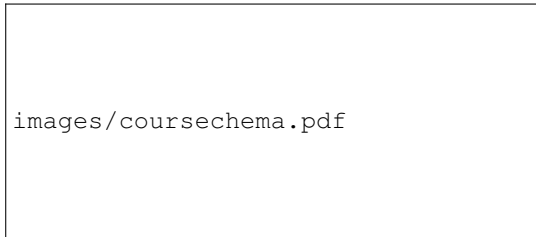


Fig. 1. Course's class diagram.

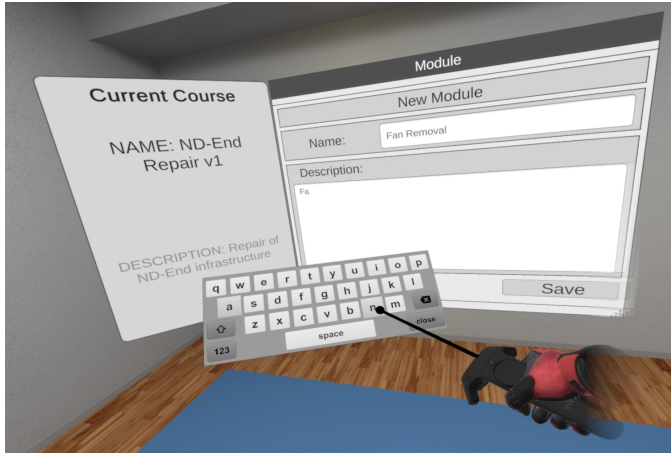


Fig. 2. Module creation panels.

more detail each of the steps necessary for the creation of a course.

3.1 Defining the Structure of the Course Data

Various pieces of information are provided within the VR environment, through menu options. These include the space itself, its contents, the phase for which training procedures are being demonstrated, and the status of that demonstration. Messages provide alerts or other instant feedback regarding non-visual aspects; text input occurs through a virtual floating keyboard; some traditional dialogue windows are employed for form filling; and information placed on side-by-side panels to allow users to view it by just rotating their heads, minimizing effort by avoiding additional menu navigation.

In this interface, the trainer can design the training course structure and edit it. This includes the training modules, the procedures that comprise them, as well as the steps within each procedure (Fig. 1). For each element, names and descriptions are entered (Fig. 2). All except the steps of the actions: those are demonstrated immersively, instead of through dialogues. Future work seeks to recreate the windows-based interactions by more immersion-adequate metaphors.

3.2 Environment Setup by the Trainer

Using the tool, the trainer can specify the various elements of the training environment. The first aspect is choosing the 3-D model of the machinery. Several models are provided to the trainer, which we imported from CAD-based data for

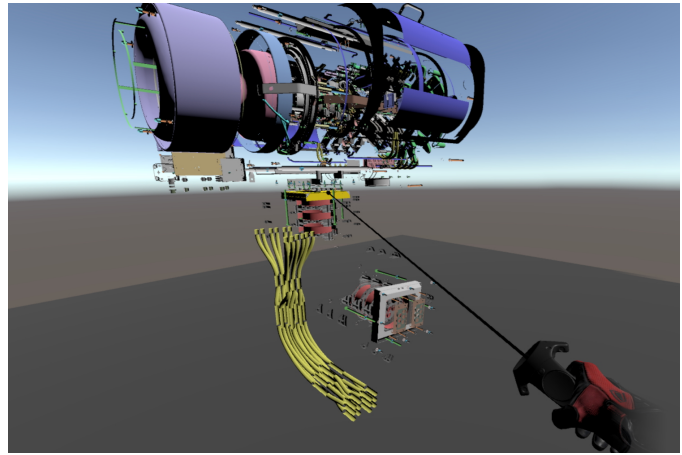


Fig. 3. Defining the interactive parts of the model.

visual accuracy. Instead of simply having a list while selecting, the trainer sees a preview of each model, in 3-D within the immersive environment, while choosing. These CAD models shall be provided by manufacturing companies, and should be usable directly by the trainers without the support of software developers as long as they are placed in the right directory. Eventually, it might be required to perform some simplification of the model complexity so that it can be used in real-time, but this can be done with off-the-shelf tools (such as CoreTechnologie's 3D_Evolution Simplifier [51] or ITI's CADfix [52]). The geometric information in these models is hierarchically stored in a tree, to identify parts and components, which we use to restrict training scope and identify interactive parts.

The second aspect ensues, with the chosen 3-D model seen in the virtual space. The trainer selects whereupon the model the training will take place, by choosing a model section. To do this, the model is progressively refined, by exploding selected components, until the intended section is exposed, which the trainer can then choose (Fig. 3). This is meant to limit the number of possible interaction alternatives for trainees, and likewise restricts the trainer's available interactions for procedure demonstration. The number of interaction options can further be restricted by the trainer, within the visible and chosen section, by identifying a subset among the exposed components. This is also done inside the immersive environment by simple part selection. The process can be repeated, as long as there are subsections in the model that can be exploded, reaching parts that are even more specific.

Also as part of the environment setup process, the trainer can provide technical documentation for the trainee to check during training and select the operators tools that will be available (Fig. 5). The documents are provided to the trainee on a floating pane (Fig. 4).

Similar setup aspects are the trainer's setup of spatial environment configurations, for the space where trainees will interact with the model. For instance, the actual looks of the virtual world: initially empty, others can be chosen (e.g., a factory floor). When the trainer chooses one, it is displayed, so the trainer gets to be immersed in it just as trainees will be (Fig. 6).

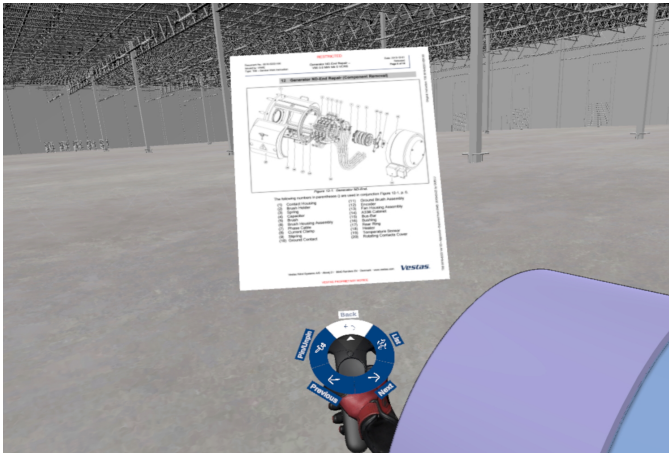


Fig. 4. Technical documentation floating pane.

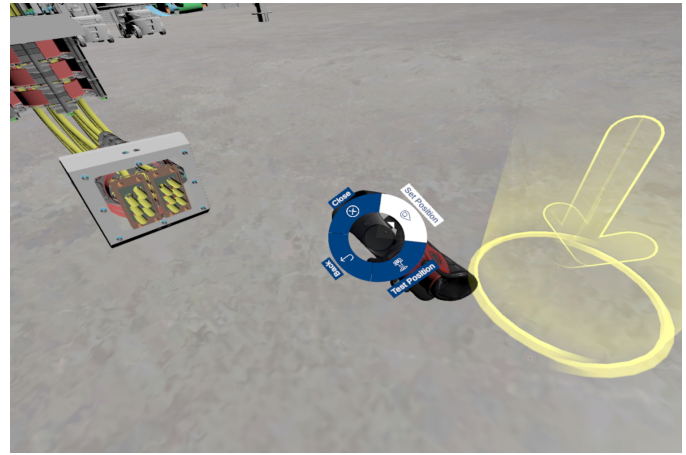


Fig. 7. Defining the initial position for the trainee.

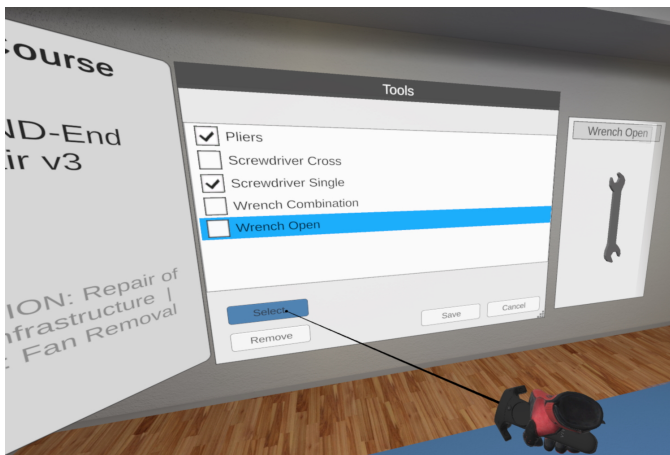


Fig. 5. Choosing tools.

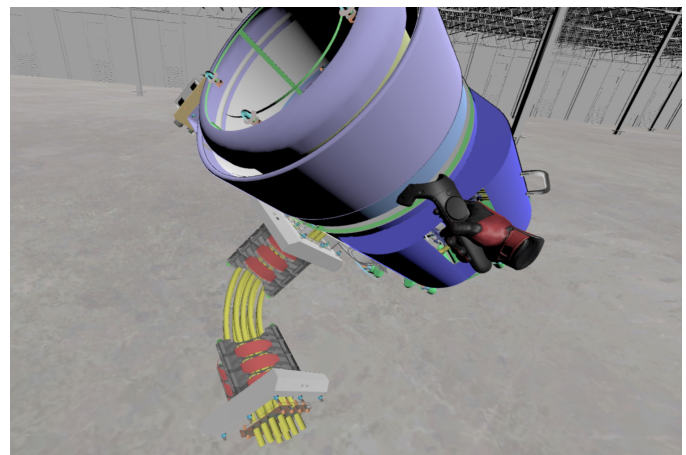


Fig. 8. Positioning the target training component model on the environment.

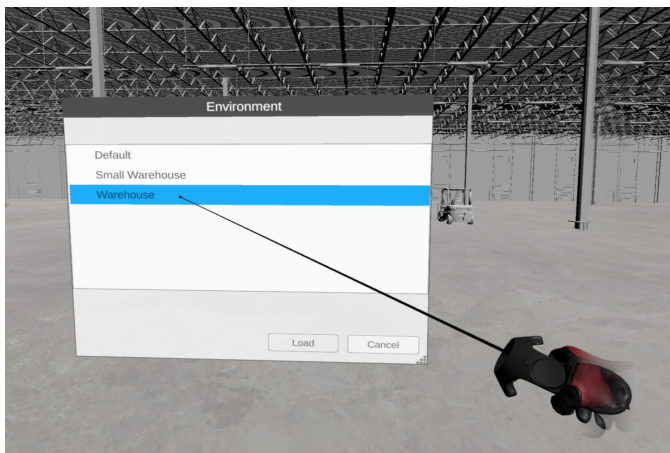


Fig. 6. Choosing an environment.

Finally, the trainer setups the trainees' starting position, also known as spawn location, and positions the 3-D model within the virtual space. The spawn location is specified immersively, by moving oneself to that place and selecting the setup option from the menu. This renders a visual cue on the floor, for awareness and subsequent editing (Fig. 7). By experiencing the trainee's perspective from its starting

position, the trainer can also adjust the height, since some models can be large and require a higher spawn location. The immersive height adjustment is made by air-grabbing with a hand controller and gesturing with it as if climbing an invisible ladder. The positioning of the 3-D model is done by simply grabbing it, dragging, and releasing it onto the intended location (Fig. 8), which is highly adequate for coarse transformations but lacks accuracy for precise manipulations [53]. More exact positioning can be achieved through the position and rotation locking functionalities, allowing users to perform such transformations separately. As suggested by previous research [54], this separation between translation and rotation is the most relevant aspect to consider when aiming to provide accurate object manipulations in mid-air.

3.3 Procedure Recording

As mentioned in Section 3.1 (Fig. 1), procedure are comprised by steps, and steps are a sequence of actions. While their naming and descriptions are typed during the setup of the course structure, the actual specification of actions takes place by immersive demonstration. To demonstrate the sequence of actions within a step, the trainer first selects that

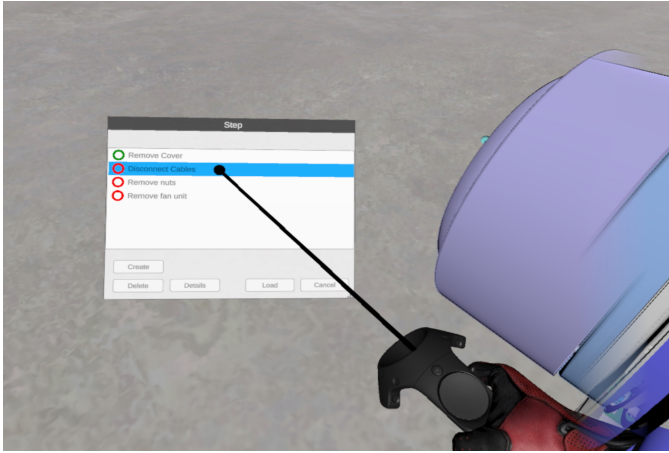


Fig. 9. Choosing step to record.



Fig. 10. Using a tool while training.

step from the list in procedure dialogue (an icon indicates if the step has been recorded or not (Fig. 9).

Before the actual recording of actions for a step, the trainer can define a part as a drop zone, enabling it as a dockable destination for other parts. This has two purposes: firstly, it simplifies interaction, by ensuring that placement is triggered easily, thus avoiding awkward attempts to position virtual parts in space with extreme precision; secondly, it minimizes undesirable situations akin to those common in real life when small parts of machinery become lost or obscured among more bulky ones. The tools selected at setup are available in order to be included as part of the recorded actions (Fig. 10).

While the trainer performs the desired actions directly on the 3-D model, demonstrating them, the system will record them as a virtual choreography. This choreography of actions becomes associated with that step, and each recorded action is represented by a sphere displayed alongside the hand controller, for visual tracking and feedback of the recording status (Fig. 11). The content of the spheres indicates the type of that action and its order number within the step.

The virtual choreography contains the actions at a semantic level associated with the training intent, it does not simply records what the trainer does in the VR authoring

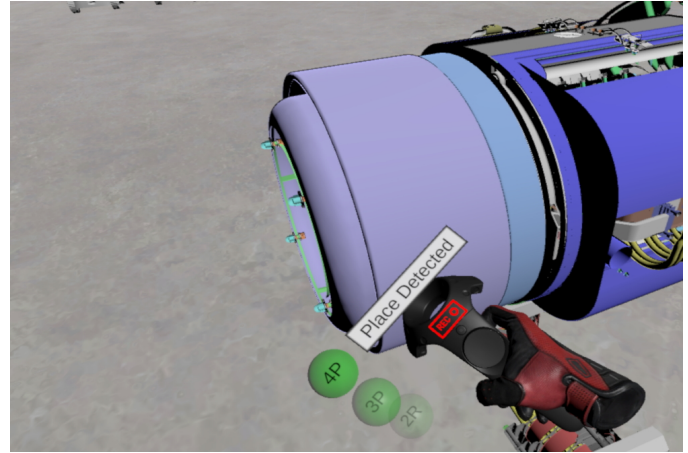


Fig. 11. Recording actions by demonstration.

tool such as, specific coordinates, meshes, or the Unity platform's commands. Using the terminology of section 2.3, the virtual choreography is stored in the knowledge domain, not the platform domain. For this, our implementation used a syntax based on xAPI, serialized as JSON, per Lacet et al.'s method [49]. For instance, removing a machine cover is stored as the verb "remove" and the object "cover," rather than the Unity-dependent sequence of cursor hovering a polygon, picking up the polygon, moving it outside the main model, and releasing it while away from an attachment point. A detailed explanation of how the Virtual Choreographies work is presented in Section 4.

3.4 Training Execution

Once the trainer records all actions, of all steps, of all procedures, of all course modules, a trainee can execute that training, by attempting to perform the procedures on the 3-D model, following the specified technical documentation. If the documentation was included in the course's setup, it would be available as a floating panel via a menu option. Trainees can also request to watch a replay of the trainer demonstration of a procedure's next action.

All previously mentioned settings are considered during training execution. For instance, only the parts set up for interaction can be interacted upon; actions requiring specific tools require them; only tools set up to be part of the course will be made available, and so forth.

While executing the training, every action the trainee takes is recorded by the system, also as a virtual choreography at a high semantic level, as explained in the previous section. This choreography can be analyzed by the trainer subsequently, either as a full replay, or from a learning analytics perspective: the platform-independence of the choreography, anchored in the knowledge domain enables a wide range of learning analytics, visualization, and learning management approaches.

This choreography-based approach also enables automated matching of the trainee's actions with the trainer's intent since irrelevant actions can be disregarded: the trainee is provided feedback about the correctly performed actions. See the next section that details this approach.

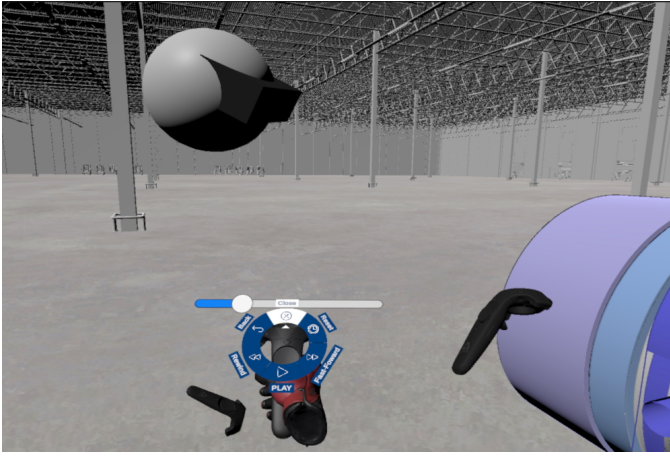


Fig. 12. Playback training.

3.5 Playback

It is possible to playback all the operations that were performed during the training generation sessions as well as the training and certification itself. Regarding the interface, this provides the usual viewing commands as if it were a film (play, stop, forward, rewind), with a timeline appearing (Fig. 12). This mechanism allows trainees, in the learning phase, to see which procedures are correct by observing precisely the actions performed by the trainer and, on the other hand, allows the trainer to see exactly which actions are performed by the students both in the training and certification phases.

4 VIRTUAL CHOREOGRAPHIES

In the related work section, we indicated using virtual choreographies both for training and certification within the 3-D developed application. This section details how the approach was applied in this system.

4.1 Creating a Virtual Choreography

Management of virtual choreographies seeks to achieve the following criteria: be platform-independent; be independent of the type of actor (real or virtual); be independent of the environment or scenario; support multi-user actions [41]. Under these requirements, our technological approach was inspired by the proposal by Lacet et al. [49], [55], which describes stories using domain semantics, and then replays them by mapping the domain semantics to the rendering platform semantics, not the stories directly. We implemented the reverse process: detect platform events and identify their domain semantics, which are stored. This takes place via this workflow (Fig. 13):

- 1) Users (actors) perform actions on the 3-D platform, aiming to perform a function or procedure. Those platform-dependent actions are sent to the Virtual Choreography Engine (VCE) server through a RESTful API, with the following parameters:
 - code [Hash code received at start of recording]
 - steporder [Contextual step order]

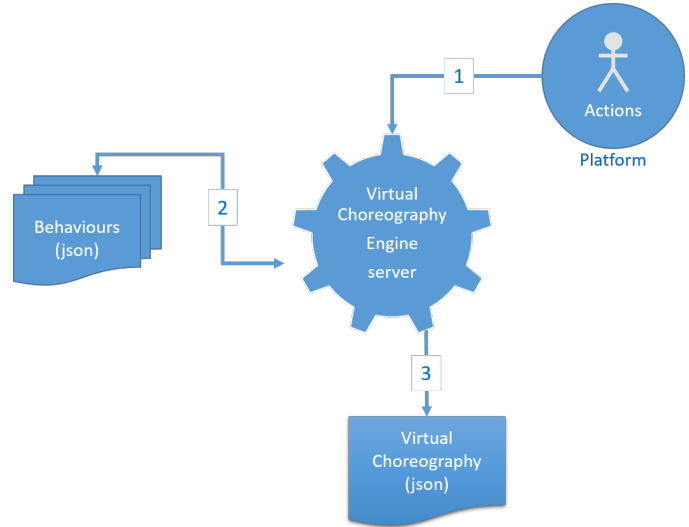


Fig. 13. Virtual Choreography Engine workflow.

- actionname [Platform-dependent action]
 - order [Number of iterations processed between actions]
 - partID [Identification of part/tool used in the action]
 - partplace [Identification of the local/used part or tool]
 - dropzoneplace [Location of the matched dropzone, if one]
 - usertype [User type, trainer or trainee]
- 2) Each action is sent as it is performed. With this information, the VCE server searches its action-behavior dictionary, currently implemented as a set of JSON files (see example below), to check if the parameterized action, within the context of previous actions, matches a behavior within the choreography semantic space. This cleans up actions that are irrelevant for future use or analysis, like repeatedly picking up and dropping a tool. The choreography semantic space determines the kinds of behaviors that are considered, and those are the ones being recorded, not platform-dependent actions.

```
{
  "id": "02",
  "name": "place",
  "description": "",
  "environment_details": [
    {
      "environment": "unity",
      "number_of_actions": 3,
      "actions": [
        { "value": "hover", "order": 1, "source_status": false, "destination_status": false, "description": "" },
        { "value": "attach", "order": 2, "source_status": false, "destination_status": false, "description": "" },
        { "value": "detach", "order": 3, "source_status": true, "destination_status": false, "description": "" }
      ]
    }
  ]
}
```



```

    }},
    {
      "environment": "secondlife",
      "number_of_actions": 3,
      "actions": [
        {
          "value": "LlDetectedGrab", "order":
            1, "source_status": false, "
            destination_status": false, "
            description": ""},
        {
          "value": "LlAttachToAvatar", "order
            ": 2, "source_status": false, "
            destination_status": false, "
            description": ""},
        {
          "value": "LlDetachFromAvatar", "
            order": 3, "source_status":
            false, "destination_status":
            false, "description": ""}
      ]
    }
  ]
}

```

- 3) If there is a matching behavior, the VCE Server will register it in the choreography. This is currently a JSON file with the following format:

```

{
  "platform": "unity",
  "courseID": "327766093",
  "moduleID": "684119581",
  "moduleOrder": "1",
  "procedureID": "943008379",
  "procedureOrder": "1",
  "0": {
    "stepID": "1634143756",
    "stepOrder": "1",
    "0": {
      "actionName": "remove",
      "order": 1,
      "partID": "753902_High.prt",
      "partPlace": "",
      "dropzonePlace": ""
    }
  },
  "1": {
    "stepID": "1634143756",
    "stepOrder": "2",
    "0": {
      "actionName": "remove",
      "order": 2,
      "partID": "753802_High.prt",
      "partPlace": "",
      "dropzonePlace": ""
    },
    "1": {
      "actionName": "place",
      "order": 3,
      "partID": "753701_High.prt",
      "partPlace": "a",
      "dropzonePlace": ""
    }
  }
}

```

With this actions–behavior dictionary approach, our mechanism generates virtual choreographies that are platform–independent (the platform actions were translated into behaviors within the choreography semantic space). They are also independent of the type of actor performing the action: the platform can send actions to the server both when they are performed live by a human user, or when they are being performed by automated characters, or even when replaying a platform–specific log. They are

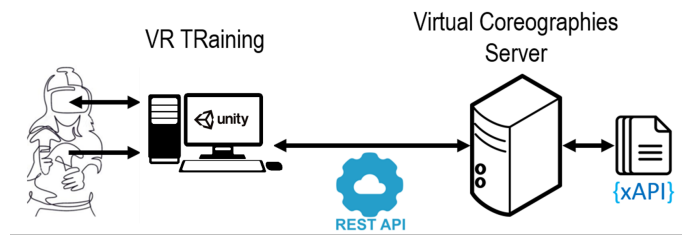


Fig. 14. VR Training environment.

independent of the software implementation of the virtual environment, because the platform–dependent actions are translated by the VCE. Finally, since the involved actor is part of the communication to the VCE, this approach enables several users to act concurrently.

4.2 Architecture

Fig. 14 shows the architecture of the VR Training environment. Data on course structure, environments, tools, models, and placements are stored locally in the tool running location.

The actions for creating the training procedure, training, and even the training carried out within the immersive environment, described in the previous points, are converted into virtual choreographies. When the user is performing a specific procedure, actions are generated within the virtual environment. Each action and the corresponding object are sent, through a restful API, to the virtual choreography server that keeps each one. The server engine seeks to find in the behavior dictionary (in the present case, it is understood that a behavior can correspond to a sequence of actions) a sequence of actions that is coincident. Each matched behavior is added to the choreography, continuously generating a set of sequential and associated behaviors for a given procedure that uses xAPI syntax [49]. We now present in detail how this approach works.

4.3 Setting Up Virtual Choreographies

First of all, the trainer must define which actions will enable the training and certification sessions to take place. That is, for the application to validate the procedures carried out by the learners, all the actions associated with each step must first be configured.

In two previously mentioned projects [41], [49] the approach taken to configure these basic behaviors involved the creation of a 2-D application that allowed the trainer to define these actions through a sequence of commands, something close to a drag–and–drop system (actions, actors, objects, etc.) — generating the virtual choreography (file in xAPI format).

However, the challenge present in this project was that everything was carried out within the immersive environment, so the way in which the sequence of actions or behaviors is defined is totally carried out within the platform, without the need to resort to other applications. Let’s detail this procedure:

4.3.1 Step

First, one must indicate which step is going to be configured. Then the trainer will go to the place where the operations are to be performed (Fig. 9).

4.3.2 Record

By clicking on the Record option, the virtual choreography server is informed that a new choreography will be generated. For that, several parameters (platform, courseID, moduleID, moduleOrder, procedureID, procedureOrder, stepID, stepOrder) are sent through the restful API.

4.3.3 Perform Actions

The platform is now available to start recording the sequence of behaviors that will create a virtual choreography. While the user performs actions, the platform communicates this information to the server.

4.3.4 Save

At any time, the user can click on stop recording, and the server is notified that the choreography configuration has been completed. The new virtual choreography will be available.

4.4 Training with Virtual Choreographies

Once all the desired choreographies are generated, there is the possibility to perform a real training scenario inside the virtual platform. After the trainee has chosen the module and procedure to train in, the trainee can start and even load the technical reference instructions into the environment. Each action that the learner performs is sent to the virtual choreography server that, after comparing it with the base choreography, sends a response as to whether that action is complete (success or failure). This sequence of actions also allows the generation of a final report regarding the expected actions, showing the ones that succeeded and the ones that failed.

4.5 Certification with Virtual Choreographies

The certification process is somewhat similar to the training process, except that no feedback is given on the student's actions every time. The student only receives the final report with the actions planned and those which were right or wrong. The student's choreography is stored for later consultation.

5 USABILITY CASE STUDY

To assess the validity of our approach, we conducted a usability case study. It was divided into three parts: course authoring, learning experience, and certification. In the first two, we evaluated the user experience of the developed tool regarding the course creation functionalities (as a trainer) and the learning experience (as a trainee). In the certification, we compared the task performance of users with and without training on our tool, in a real-world maintenance procedure on a physical model. Except for a participant that used our tool as a trainee and later executed the certification task, no participants were present in more than one part.

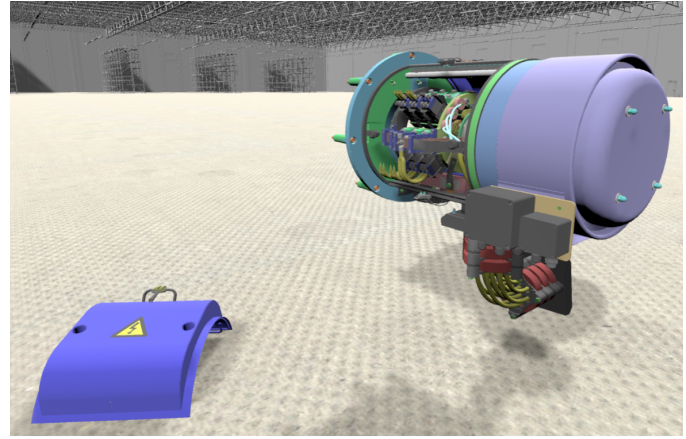


Fig. 15. Main cover removed from the virtual model of the Generator used in the evaluation tasks.

5.1 Participants

There was a total of nine participants (all male) in our usability case study: two expert trainers for the course authoring part; six professionals for the trainee learning experience; and two professionals (one shared with the trainee part) for the certification. Participants' ages ranged from 22 to 46 years old (trainers avg. = 31, trainees avg. = 35, certification avg. = 27). Two participants had graduated high school, while others had BSc (four), and MSc (three) degrees. Most of the subjects (seven) had prior field knowledge (e.g., field technicians and service managers). In particular, the two participants of the certification part were 22 and 31 years old, had a high school diploma, and two and ten years of experience as a turbine technician, respectively. Both trainers already had regular contact with VR, but only one trainee had prior experience utilizing VR equipment. All users had previous knowledge about the components used in the study, either on their physical form through field experience or on their virtual counterparts (CAD models).

5.2 Method

For the authoring and learning parts, all sessions followed the same structure. It started with a visual demo of what the platform was focused on, and the interactions available to them. After this, there was a period of adaptation (five to ten minutes) where participants were allowed to freely experiment with the tool, not using the same data that was going to be tested in the next step. Then, participants performed a set of tasks. Finally, they were asked to fulfill a questionnaire.

In the certification, participants executed a standard set of evaluation tasks for new collaborators who had just completed their training. Their performance was evaluated by the same expert trainer.

5.3 Tasks

Trainers were asked to create a course focused on the disassembly of a Generator (Fig. 15), which included: adding the information regarding the course, modules, procedures, and steps; choosing and configuring the Generator model and its parts; selecting the environment as well as the necessary

tools and documentation; and recording the required actions. This was done following an existing document regarding the disassembly of that Generator, which is currently used for fieldwork, which trainers could access inside the tool. We counted as an error whenever the trainer intended to perform a specific operation but selected a different option.

Trainees, on the other hand, executed the same course that was previously configured by our team. The course consisted of a set of six modules, and each one revolved around removing a subcomponent of the Generator's main part. The tasks were designed so that there was only one correct way to execute them, and errors were counted by the overseeing team (errors consisted of removing components that were not the current target). The procedures of the first and second modules were simple as the subcomponents are clear to see for users that do not have previous knowledge on the Generator (e.g., it is simple to understand what a cover is in this context). By the nature of the process itself, the procedures of the six modules grew in complexity, as more simple elements were followed by smaller and more precise elements (e.g., cover vs. brushes). Participants could watch the prerecorded trainer actions and consult the documentation. Other than that, no aid was given to participants.

In the certification part, participants executed the same task as trainees performed in our tool, as it is a very exemplificative real-world intervention, this time on a functional physical model of the Generator.

5.4 Measures

For each part of our usability case study, we had different metrics. For the course authoring and learning experience parts, we were mostly interested in subjective measures, namely participants' opinions regarding several aspects and their overall satisfaction. These were collected through a questionnaire comprised of statements to which the participants had to respond using a Likert scale (1 - totally disagree, 5 - totally agree). Questions focused on the overall usability and the fidelity of virtual components in relation to their physical counterparts, and are shown in Table 1. We analyzed each question separately. Additionally, we registered any comments participants might have. In both parts, we also measured the number of errors, as defined in the Section 5.3, and task execution time for the trainees. For the certification, we were interested in knowing if the task could be successfully completed and, in case of success, its completion time and number of errors. In all cases, there was no maximum number of errors that a participant could perform.

5.5 Apparatus

The authoring and learning sessions were performed in a laboratory with restricted access, thus providing a controlled environment and free of external inputs (Fig. 16). The certification part was executed in a warehouse, also with controlled access (Fig. 17).



Fig. 16. Trainer participant setting up a course.



Fig. 17. Participant performing a certification task with a physical model of the Generator.

5.6 Results

We analyzed the results regarding each part of our usability case study separately, as they focus on distinct components of our approach.

5.6.1 Course Authoring

For the creation of the course, the two trainers made a median of 8.5 errors (IQR = 1.5). We argue that, considering the entire course dimension and that this was the first time participants interacted with our tool, a promising result. The questionnaires responses (Table 1) revealed that trainers found the tool useful and easy to get used to, and that the virtual model was faithful to the physical counterpart. These participants also reported a median overall satisfaction of 2.5 out of 5 (IQR = 0.5). According to their comments, their

TABLE 1

Questionnaires' results for the course authoring (trainers) and learning experience (trainees). Higher values indicate higher agreement with the statement (1 - totally disagree, 5 - totally agree). Showing median and interquartile ranges.

Statement	Trainers (n = 2)	Trainees (n = 6)
1. The task was easy to complete.	3 (0)	4 (1.5)
2. The functionality was useful.	4 (0)	4.5 (1)
3. I could easily identify the several components of the model.	5 (0)	5 (0)
4. I could navigate the work instructions as easily as I would with the physical counterpart.	3 (0)	4.5 (1)
5. I could interact with the virtual model like what I would do with a physical model.	3.5 (1.5)	4.5 (1.75)
6. I could perform the scenario that was proposed.	3.5 (1.5)	4 (2.25)
7. I think that I would like to use the system frequently.	4 (1)	4.5 (1)
8. I feel very confident using this system.	3.5 (0.5)	5 (0)
9. I found this system unnecessarily complex.	2 (0)	2 (1.5)
10. I need to learn a lot of things before I could get going with this system.	2 (0)	3.5 (1.75)

answers were not higher due to two main reasons. The first was related to text input, as they found the virtual keyboard in VR far from optimal. Indeed, one of the participants suggested the possibility to later update titles and descriptions on a desktop computer. The other reason was that going back and forward between different scenes for the course and model setup was cumbersome. As a future iteration, we intend to merge as many settings as possible in the same scene.

5.6.2 Learning Experience

Regarding the learning experience, participants made a median of 2 errors (IQR = 1.5), with an average task completion time of 10.4 minutes ($\sigma = 5.4$, with a minimum duration time of 6 min. and a maximum of 21 min.). We believe that this range in completion times might be due to participants' previous experience with virtual 3-D training tools, despite not being immersive. Participants answers to the questionnaire were highly positive (Table 1). Despite considering that they needed to learn a considerable amount of new concepts, mostly related to their inexperience with VR environments, participants highlighted the tool's easy of use, utility, and the virtual model's fidelity. This was also reflected in the overall satisfaction, which attained a median of 4 out of 5 (IQR = 1.5). Additionally, the participants made a point of commenting on the similarity of the virtual model to the physical Generator, including the colors used, which allowed recognizing real-world procedures and scenarios. They also indicated that it would be very useful for training service technicians. However, they also pointed out that our tool imposed some sequence orders that are not necessarily true, as there are situations where the order in which components are removed is not relevant (e.g., removing a set of screws or brushes). This is a limitation of our choreography-based approach, which we aim to tackle with future research.

5.6.3 Certification

Lastly, the certification part of our user evaluation was supervised by an experienced trainer. One of the participants had previous training with our tool (22 years old with two years of experience), while the other did not (31 years old

with ten years of experience). Both of them completed the task successfully, without any error. However, the participant with previous VR training completed the task within 35 minutes – who highlighted that it was easy because he had experienced the procedure virtually – whereas the other participant took 42 minutes. While we recognize that just two participants is too low a number to draw meaningful conclusions, we find these results and opinions promising, which encourages us to further explore our approach.

6 CONCLUSION

The presented approach is a proof-of-concept that immersive learning can be authored by trainers, without the intervention of software developers, inside the same immersive environment and with the same tools and artifacts as those that the trainee will use. Conversely, the trainees can also author their intent in the same way, for trainers to experience and understand the trainees' perspective. This enables a situated learning approach, where trainers and trainees communicate intent, actions, and perspectives embodying their knowledge in the physical and semantic contexts of their areas of expertise, without the mediation of software development tools or professionals.

Various aspects of this approach should be subjected to further study in the future. For instance, employing a virtual keyboard is not optimal, and alternatives should be sought to improve the overall satisfaction of course authors in these tasks. For typing that must occur within the virtual environment, optimizations might include voice recognition, automatic text completion suggestions, as well as exploring several novel methods being introduced by other researchers [56]. We envision furthering that by increasing interoperability of immersive authoring with traditional Web-based learning management systems. Most of the typing tasks would then occur within traditional Web interfaces rather than using the virtual keyboard. Our Inven!RA initiative follows this very path [57]. By providing interoperability between Web-based planning, design, and tracking, and its immersive counterparts, users would not have to go back and forward so often between different scenes for the course and model setup. The Inven!RA approach posits that specific interventions can be specified as

activities, with a variety of available analytics, which can be combined into a larger “inventive activities plan” (IAP), where the various activities’ analytics are mapped to overall learning objectives. These are but two particular aspects that require further exploration.

We acknowledge that the number of participants is a limitation of our work and intend to conduct tests with a larger number of participants once pandemic status enables it. An additional limitation is the approach for matching actions to choreography elements. It is based on a simple decision tree, lacking the flexibility for more complex situations. For instance, tightening a set of screws can often be done in any order, but our approach is rigid. This area of matching can be tackled by different action-matching techniques, such as pre- and post-conditions [50], and pattern-based deep learning approaches should also be explored. There are also areas of inquiry regarding the flexibility of physical components, such as cables, and realizing how they would impact procedure definition and matching.

This system opens up new fields of application for virtual immersive learning environments. In this sense, there are countless areas where this type of approach can be applied. Namely in the most diverse industrial training environments where it is necessary to follow a set of procedures (automotive industry, aeronautics, assembly of parts, etc.), as well as in procedural training contexts (health, vocational training, cooking, etc.). But also in complex areas, where the learner’s ability to demonstrate intent and perspective is paramount for enhanced instructor support and feedback. In brief, immersive authoring by instructors can empower widespread adoption of immersive learning scenarios in which behavioral and experiential learning in realistic environments plays an important role.

ACKNOWLEDGMENTS

This paper is an expanded version of papers presented at the 2021 IEEE Conference on Virtual Reality and 3D User Interfaces. This work was financed by the ERDF - European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation - COMPETE 2020 under the Portugal 2020 Partnership Agreement, and through the Portuguese National Innovation Agency (ANI) as part of project POCI-01-0247-FEDER-038524, and through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia, within project UIDB/50014/2020.

REFERENCES

- [1] Å. Fast-Berglund, T. Fässberg, F. Hellman, A. Davidsson, and J. Stahre, “Relations between complexity, quality and cognitive automation in mixed-model assembly,” *J. of Manuf. Syst.*, vol. 32, no. 3, Jul. 2013, doi: 10.1016/j.jmsy.2013.04.011.
- [2] D. Beck, L. Morgado, and P. O’Shea, “Finding the gaps about uses of immersive learning environments: a survey of surveys,” *J. of Universal Comput. Sci.*, vol. 26, pp. 1043–1073, Aug. 2020, doi: 10.3897/jucs.2020.055.
- [3] A. A. García, I. G. Bobadilla, G. A. Figueroa, M. P. Ramírez, and J. M. Román, “Virtual reality training system for maintenance and operation of high-voltage overhead power lines,” *Virtual Reality*, vol. 20, no. 1, Jan. 2016, doi: 10.1007/s10055-015-0280-6.
- [4] H. C. Miles, S. R. Pop, S. J. Watt, G. P. Lawrence, and N. W. John, “A review of virtual environments for training in ball sports,” *Comput. & Graph.*, vol. 36, no. 6, pp. 714–726, Oct. 2012, doi: 10.1016/j.cag.2012.04.007.
- [5] A. Pinheiro *et al.*, “Development of a mechanical maintenance training simulator in opensimulator for f-16 aircraft engines,” *Entertainment Comput.*, vol. 5, no. 4, Dec. 2014, doi: 10.1016/j.entcom.2014.06.002.
- [6] T. Im, D. An, O.-Y. Kwon, and S.-Y. Kim, “A virtual reality based engine training system - a prototype development & evaluation,” in *Proc. of the 9th Int. Conf. on Comput. Supported Educ. - Vol. 1: CSEDU*, Porto, Portugal, Apr. 21–23, 2017, doi: 10.5220/0006263702620267. pp. 262–267.
- [7] A. Grabowski and J. Jankowski, “Virtual reality-based pilot training for underground coal miners,” *Safety Sci.*, vol. 72, Feb. 2015, doi: 10.1016/j.ssci.2014.09.017.
- [8] J. Garrison, “John dewey’s theory of practical reasoning,” *Educational philosophy and theory*, vol. 31, no. 3, pp. 291–312, Jan. 1999, doi: 10.1111/j.1469-5812.1999.tb00467.x.
- [9] E. H. Schein, “Kurt lewin’s change theory in the field and in the classroom: Notes toward a model of managed learning,” *Syst. Pract.*, vol. 9, no. 1, pp. 27–47, Feb. 1996, doi: 10.1007/BF02173417.
- [10] G. Brown and C. Desforges, *Piaget’s Theory*. Routledge, 2013.
- [11] M. McCarthy *et al.*, “Experiential learning theory: From theory to practice,” *J. of Bus. & Econ. Res.*, vol. 8, no. 5, Jun. 2010, doi: 10.19030/jber.v14i3.9749.
- [12] D. A. Kolb, *Experiential learning: Experience as the source of learning and development*. FT press, 2014.
- [13] J. Hollan, E. Hutchins, and D. Kirsh, “Distributed cognition: toward a new foundation for human-computer interaction research,” *ACM Trans. on Comput.-Human Interaction*, vol. 7, no. 2, pp. 174–196, Jun. 2000, doi: /10.1145/353485.353487.
- [14] “Unity,” <https://unity.com/> (accessed Jun. 15, 2021).
- [15] “Unreal engine,” <https://www.unrealengine.com/> (accessed Jun. 15, 2021).
- [16] H. Gaspar, L. Morgado, H. Mamede, T. Oliveira, B. Manjón, and C. Gütl, “Research priorities in immersive learning technology: the perspectives of the ilrn community,” *Virtual Reality*, Jul. 2019, doi: 10.1007/s10055-019-00393-x.
- [17] “Eon reality,” <https://eonreality.com/> (accessed Jun. 15, 2021).
- [18] “Xvr simulation,” <https://www.xvrsim.com/> (accessed Jun. 15, 2021).
- [19] “Vestas wind systems a/s,” <https://www.vestas.com/> (accessed Jun. 15, 2021).
- [20] T. Ley, R. Maier, S. Thalmann, L. Waizenegger, K. Pata, and A. Ruiz-Calleja, “A knowledge appropriation model to connect scaffolded learning and knowledge maturation in workplace learning settings,” *Vocations and Learn.*, vol. 13, no. 1, pp. 91–112, Aug. 2020, doi: 10.1007/s12186-019-09231-2.
- [21] F. Cassola, M. Pinto, D. Mendes, L. Morgado, A. Coelho, and H. Paredes, “Immersive authoring of virtual reality training,” in *2021 IEEE Conf. on Virtual Reality and 3D User Interfaces Abstr. and Workshops (VRW)*, Lisbon, Portugal, Mar. 27–1, 2021, doi: 10.1109/VRW52623.2021.00199. pp. 633–634.
- [22] F. Cassola, M. Pinto, D. Mendes, L. Morgado, A. Coelho, and H. Paredes, “A novel tool for immersive authoring of experiential learning in virtual reality,” in *2021 IEEE Conf. on Virtual Reality and 3D User Interfaces Abstr. and Workshops (VRW)*, Lisbon, Portugal, Mar. 27–1, 2021, doi: 10.1109/VRW52623.2021.00014. pp. 44–49.
- [23] K. K. Bhagat, W.-K. Liou, and C.-Y. Chang, “A cost-effective interactive 3d virtual reality system applied to military live firing training,” *Virtual Reality*, vol. 20, no. 2, Feb. 2016, doi: 10.1016/j.ssci.2014.09.017.
- [24] L. M. Alves Fernandes *et al.*, “Exploring educational immersive videogames: an empirical study with a 3d multimodal interaction prototype,” *Behav. & Inf. Technol.*, vol. 35, no. 11, Aug. 2016, doi: 10.1080/0144929X.2016.1232754.
- [25] H. De Visser, M. O. Watson, O. Salvado, and J. D. Passenger, “Progress in virtual reality simulators for surgical training and certification,” *Med. J. of Aust.*, vol. 194, Aug. 2011, doi: 10.5694/j.1326-5377.2011.tb02942.x.
- [26] A. Mossel, A. Peer, J. Göllner, and H. Kaufmann, “Requirements analysis on a virtual reality training system for cbrn crisis preparedness,” in *Proc. of the 59th Annu. Meeting of the Int. Soc. for the Syst. Sciences*, Berlin, Germany, Aug. 2–7, 2015.

- [27] Ü. Çakiroğlu and S. Gökoğlu, "Development of fire safety behavioral skills via virtual reality," *Comput. & Educ.*, vol. 133, pp. 56–68, May 2019, doi: 10.1016/j.compedu.2019.01.014.
- [28] H. S. Lee, Y. J. Park, and S. W. Park, "The effects of virtual reality training on function in chronic stroke patients: a systematic review and meta-analysis," *BioMed Res. Int.*, vol. 2019, Jun. 2019, doi: 10.1155/2019/7595639.
- [29] D. Puel, "An authoring system for vr-based firefighting commanders training," *Electron. Imag.*, vol. 2018, no. 3, pp. 469–1, Jan. 2018, doi: 10.2352/ISSN.2470-1173.2018.03.ERVR-469.
- [30] M. Sarov, R. Gao, J. Youngquist, G. Sarosi, S. Kurenov, and J. Peters, "An authoring interface for surgeon-authored vr training," in *CARS 2018 - Comput. Assisted Radiol. and Surgery Proc. of the 32nd Int. Congr. and Exhib.*, Berlin, Germany, Jun. 20–23, 2018.
- [31] A. Vargas González *et al.*, "A comparison of desktop and augmented reality scenario based training authoring tools," in *2019 IEEE Int. Symp. on Mixed and Augmented Reality (ISMAR)*, Beijing, China, Oct. 14–18, 2019, doi: 10.1109/ISMAR.2019.00032. pp. 339–350.
- [32] S. Eroglu *et al.*, "Design and evaluation of a free-hand vr-based authoring environment for automated vehicle testing," in *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, Lisboa, Portugal, Mar. 27–1, 2021, doi: 10.1109/VR50410.2021.00020. pp. 1–10.
- [33] H. Coelho, M. Melo, J. Martins, and M. Bessa, "Collaborative immersive authoring tool for real-time creation of multisensory vr experiences," *Multimedia Tools and Appl.*, vol. 78, no. 14, pp. 19 473–19 493, Feb. 2019, doi: 10.1007/s11042-019-7309-x.
- [34] J. Wolfartsberger and D. Niedermayr, "Authoring-by-doing: Animating work instructions for industrial virtual reality learning environments," in *2020 IEEE Conf. on Virtual Reality and 3D User Interfaces Abstr. and Workshops (VRW)*, Atlanta, GA, USA, Mar. 22–26, 2020, doi: 10.1109/VRW50115.2020.00038. pp. 173–176.
- [35] J. V. McConnell and R. P. Philipchalk, *Understanding human behavior*. Harcourt Brace Jovanovich, 1992.
- [36] U. D. Dompke, "Human Behaviour Representation - Definition," in *RTO SAS Lecture Ser. on "Simul. of and for Mil. Decis. Making"*, Rome, Italy, Oct. 2001.
- [37] V. Engmo, *Representation of Human Behavior in Military Simulations*. Norwegian Defence Research Establishment, 2008.
- [38] B. Schmidt, "Human Factors in Complex Systems The Modelling of Human Behaviour," *Tech. Rep.*, 2005. [Online]. Available: <http://www.or.uni-passau.de>
- [39] M. E. Bratman, *Intention, Plans and Practical Reason*. The David Hume Series, 1987.
- [40] E. Norris, A. N. Finnerty, J. Hastings, G. Stokes, and S. Michie, "A scoping review of ontologies related to human behaviour change," *Nature Human Behav.*, vol. 3, no. 2, pp. 164–172, Feb. 2019, doi: 10.1038/s41562-018-0511-4.
- [41] E. Silva, N. Silva, and L. Morgado, "Model-driven generation of multi-user and multi-domain choreographies for staging in multiple virtual world platforms," in *Int. Conf. on Model and Data Eng.*, Larnaca, Cyprus, Sep. 24–26, 2014, pp. 77–91.
- [42] A. Lopes *et al.*, "Use of a virtual world system in sports coach education for reproducing team handball movements," *J. For Virtual Worlds Res.*, vol. 2, no. 1, Mar. 2009, doi: 10.4101/jvwr.v2i1.399.
- [43] A. Nakasone *et al.*, "Openenergysim: A 3d internet based experimental framework for integrating traffic simulation and multi-user immersive driving," in *Industry Track SimulationWorks*, Apr. 2012, doi: 10.4108/icst.simutools.2011.238605.
- [44] J. Ribeiro, J. Almeida, R. Rossetti, and A. Coelho, "Using serious games to train evacuation behaviour," in *Iberian Conf. on Inf. Syst. and Technologies, CISTI*, Madrid, Spain, Jun. 20–23, 2012, pp. 1–6.
- [45] A. Weiss, V. Andrikopoulos, M. Hahn, and D. Karastoyanova, "Model-as-you-go for choreographies: rewinding and repeating scientific choreographies," *IEEE Trans. on Services Comput.*, Jul. 2017, doi: 10.1109/TSC.2017.2732988.
- [46] R. Young and M. Riedl, "An architecture for integrating plan-based behavior generation with interactive game environments," *J. of Game Develop.*, Jun. 2004.
- [47] S. Deterding, M. Sicart, L. Nacke, K. O'Hara, and D. Dixon, "Gamification. using game-design elements in non-gaming contexts," in *Proc. of the 2011 Annu. Conf. Extended Abstr. on Human Factors in Comput. Syst. - CHI EA '11*, May 7–12, 2011, doi: 10.1145/1979742.1979575.
- [48] F. Cassola, J. Iria, H. Paredes, L. Morgado, A. Coelho, and F. Soares, "Using choreographies to support the gamification process on the development of an application to reduce electricity costs," in *Games and Learning Alliance*, Lisbon, Portugal, Dec. 5–7, 2017.
- [49] D. Lacet, F. Penicheiro, L. Morgado, and A. Coelho, "Preserving story choreographies across multiple platforms: An approach to platform-independent reuse of characters' behaviors for games, simulations, animations and interactive videos," in *Proc. of the 9th Int. Conf. on Digit. and Interactive Arts*, Braga, Portugal, Oct. 23, 2019.
- [50] A. Vilela, A. Marques, H. Costa, J. Rafael, R. F. F. Prada, and L. Morgado, "Aplicação de avatares autónomos para desempenhar o papel de membros na execução de trabalhos em equipa," 2012.
- [51] "3d_evolution simplifier," <https://www.coretechnologie.com/products/3d-evolution/simplifier.html> (accessed Nov. 22, 2021).
- [52] "Cadfix," <https://www.cadfix.com/> (accessed Nov. 22, 2021).
- [53] D. Mendes, F. M. Caputo, A. Giachetti, A. Ferreira, and J. Jorge, "A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments," *Comput. Graph. Forum*, vol. 38, no. 1, pp. 21–45, Mar. 2019, doi: 10.1111/cgf.13390.
- [54] D. Mendes, M. Sousa, R. Lorena, A. Ferreira, and J. Jorge, "Using custom transformation axes for mid-air manipulation of 3d virtual objects," in *Proc. of the 23rd ACM Symp. on Virtual Reality Softw. and Technol.*, Gothenburg, Sweden, Nov. 8 – 10, 2017, doi: 10.1145/3139131.3139157. pp. 1–8.
- [55] D. Lacet, F. Penicheiro, and L. Morgado, "Magical board theatre: interactive stories that can be played on multiple boards: two educational prototypes," in *Videojogos 2020 - 12th Int. Conf. on Videogame Sciences and Arts*, Mirandela, Portugal, Nov. 26–28, 2020, pp. 164–176.
- [56] J. Adhikary and K. Vertanen, "Text entry in virtual environments using speech and a midair keyboard," *IEEE Trans. on Visualization and Comput. Graph.*, vol. 27, no. 5, pp. 2648–2658, 2021, doi: 10.1109/TVCG.2021.3067776.
- [57] D. Cota, T. Cruzeiro, D. Beck, A. Coelho, and L. Morgado, "In-ventivetr@ining-inven! ra architecture activity provider modules for online tracking of microelectronics student projects." *Revista de Ciências da Computação*, vol. 16, 2021.

Fernando Cassola is PhD Researcher at INESC TEC and an Invited Assistant Professor at Faculty of Engineering of the University of Porto. Received his PhD (2021) in Computer Science about the use of gamification to promote behavior change. His work is mainly focused in Gamification, Virtual Choreographies, Virtual Environments, and Behavior Change.

Daniel Mendes is an Auxiliary Researcher at INESC TEC and an Invited Assistant Professor at Faculty of Engineering of the University of Porto. Received his PhD (2018), MSc (2011), and BSc (2008) degrees in Computer Science and Engineering from Instituto Superior Técnico, University of Lisbon. His work is mainly focused on virtual and augmented reality, multimodal interfaces, 3D user interfaces, and touch / gesture-based interactions.

Manuel Pinto received his MSc degree from the University of Trás-os-Montes and Alto Douro. His research work was mainly focused on virtual reality and 3D environments.

Leonel Morgado is Associate Professor with Habilitation, at Universidade Aberta, where he lectures on research methods, programming, and immersive learning, and Senior Researcher at INESC TEC. He is also Vice-President for Scientific Quality of the international research association, Immersive Learning Research Network, and Secretary of the Assembly of the Portuguese Society of Videogame Sciences. His main research interest is the use and development of educational technology, which he pursues since 2000, focusing on immersive environments since 2006.

Sara Costa is a current employee of Vestas Wind Systems. Has an MSc degree in Auditing and Taxation from the Catholic University of Porto. Her work is mainly focused on business development and analysis.

Luis Anjos is a current employee of Vestas Wind Systems. Received his MSc degree from the University of Minho. His work is focused on Brain Computer Interfaces and Neuroscience.

David Marques is a current employee of Vestas Wind Systems. Received his MSc degree from the Polytechnic Institute of Cávado and Ave. His work is focused on XR and Game Development.

Filipe Rosa is a current employee of Vestas Wind Systems. Received his MSc degree from the University Institute of Maia. His work is focused on Solution Architecture and Infrastructure.

Ana Maia is a current employee of Vestas Wind Systems. Received her MSc degree from the Faculty of Engineering of the University of Porto. Her work is focused on XR Development.

Helga Tavares is a current employee of Vestas Wind Systems. Received her MSc degree in Business Economics and Strategy from the University of Porto. Her work is mainly focused on business development and analysis.

António Coelho is Associate Professor with Habilitation at the Department of Informatics Engineering of the Faculty of Engineering of the University of Porto where he teaches in the areas of Computer Graphics, Programming and Digital Games, and is the director of the Doctoral Program in Digital Media at the University of Porto. He is also a senior researcher at INESC TEC where he coordinates the Computer Graphics and Virtual environments area.

Hugo Paredes received B.Eng. and Ph.D. degrees in Computer Science from the University of Minho, Braga, Portugal, in 2000 and 2008, and the Habilitation title from the University of Trás-os-Montes e Alto Douro (UTAD), Vila Real, Portugal in 2016. Since 2003, he has been at UTAD, where he is currently Associate Professor with Habilitation, lecturing on systems integration and distributed systems. In 2017 he was a visiting faculty at Human Computer Interaction Institute at Carnegie Mellon University, Pittsburgh, PA. Currently he is the director of the Master Program in Informatics Engineering at UTAD, and a member of the managing board of the PhD in Computer Science at UTAD. He is Assistant Coordinator of the Centre for Computer Graphics and Information Systems (CSIG) at the Institute for Systems and Computer Engineering, Technology and Science (INESC TEC), where he is a Senior Researcher.