11th International Conference of the Immersive Learning Research Network
Selected Academic papers for the iLRN Proceedings
DOI: https://doi.org/10.56198/5xdy4g60

Pedagogical Extension of the Smart Object Concept for Embodied Conversational Agents

Jean-Benoit Culié[1], David Panzoli[1] and Michel Galaup[2]

[1] University of Toulouse, France
[2] INU Champollion, Albi, France
IRIT UMR CNRS 5505
UMR EFTS
culie3@gmail.com
panzoli@irit.fr
michel.galaup@univ-jfc.fr

Abstract. The concept of smart object, which emerged with the advent of virtual environments in the early 2000s, consists of equipping interactive objects with data enabling them to inform agents, users or non-player characters, about their use or interaction methods. This concept simplifies the scripting of non-player characters' behaviours by delegating and distributing part of their control in the objects. This way, it brings adaptability, modularity and robustness to behaviour. In this article, we describe how we have extended the concept of smart object by giving it a pedagogical scope. Doing so, we enable a virtual companion to behave autonomously in the informed environment, but also to take on a pedagogical role towards the learner. By exemplifying the benefits of this original architecture through three case studies based in a virtual factory, we demonstrate that the companion operates on any environment in a generic fashion, and furthermore adapts the training content to the learner's level of expertise.

Keywords: Immersive Learning, Embodied Conversational Agents, Smart-Objects, Learning Factories.

1 Introduction

For the purposes of entertainment, instruction or training, virtual environments enable users, players or learners to immerse themselves in synthetic worlds. They are usually dynamic, and ideally interactive, evolving as users interact with them. Whether these environments digitally reproduce public places (cities, resorts, tourist attractions, etc.) or workplaces (workshops, factories, warehouses), populating them with non-player characters, i.e. autonomous characters controlled by algorithms or systems trained by machine learning, offers numerous advantages. Virtual users can be used to bring a street or neighbourhood to life, enhancing the user's immersion, or to better illustrate the function of a space or the flow of people, as in a railway station [16], for example. In cultural heritage applications, virtual tourists can enhance the reconstruction of a famous or ancient place by adding dynamism to the scene, as in this exploration of the city of Pompeii [11]. 3D models and appropriate animations can also be used to pass on historical knowledge about life at the time by accurately recreating costumes, religious or social practices. Finally, in immersive training, the roles assumed by virtual agents add a social dimension to skills learning. For instance, they can illustrate activity in the workspace, assist the user in collaborative tasks, or even replace a missing player in a scenario. They can be integral to the training scenario if they are used intelligently to make mistakes for the learner [4]. Finally, by adopting dialogic, social and emotional interaction skills, they also become capable of playing the role of educational tutor to the learner.

There are many techniques for animating or simulating behaviour (agent-centred or multi-agent approaches, advanced planning algorithms or even interactive storytelling. Yet, in the case of virtual agents that evolve autonomously in a complex environment (i.e. filled with objects, machines or other agents) a winning strategy is to delegate some of the processing to the objects in the environment themselves, in order to reduce the complexity of the agent's behavioural engine.

The term informed environment refers to an environment where objects' graphical representations (geometry, textures, etc.) are enriched with semantic information about their nature, function, or dynamics, creating *smart objects*. These objects adjust 3D interactions [1], automate animations [13], and plan complex behaviors [9], reducing code complexity through a" divide and conquer" strategy [3] while promoting a flexible and reusable architecture [6].

First introduced in 1999 [9], the concept of *smart objects* builds on Gibson's affordances theory [7], which describes how individuals perceive and act upon their environment based on available possibilities. Kallmann [9] extended this idea to interactive virtual objects, enabling them to respond intelligently to user actions, adapt behaviors, and offer context-appropriate possibilities for action.

In immersive training or cultural heritage applications, agents pedagogical value depends on their level of interaction with the user. Crowds provide general spatial insights, while animated characters offer detailed information. Conversational agents, however, excel at engaging learners through natural, explicit interactions. The next section explores the roles conversational agents play in pedagogical transmission, while Section 3 extends the *smart object* concept for embodied conversational agents. Section 4 demonstrates this approach, and Section 5 summarizes our contributions.

2 State of the Art in Conversational Agents

Embodied conversational agents (ECAs) are advanced virtual agents designed to convincingly simulate human interactions. They recognize and respond to verbal and non-verbal behaviors and can incorporate affective capabilities, such as emotional expression through voice intonations, facial expressions, or body language. Recent studies propose using emotion recognition to regulate their empathic behavior [5]. For Cassell et al. [2], "embodied conversational agents are specifically conversational in their behaviours, and specifically human in the way they use their bodies in conversation". The authors identify the defining properties of an ECA: recognise and respond to verbal and non-verbal cues, generate verbal and non-verbal signals, process conversational functions such as speech turn-taking, feedback and repair mechanisms, and give signals indicating the state of the conversation and contribute to the discourse with new suggestions. Provoost et al. [14] define three rules to be followed, also including the concept of agency, for a software entity to be considered an ECA. Firstly, it must have a virtual or physical incarnation. Secondly, it must interact with a user. And finally, it must have a reasonable sense of agency, meaning that its behaviour must be autonomous and that it must demonstrate some form of reasoning. In other words, the ECA must be autonomous and intelligent in a way that is not controlled by a human operator.

Enhancing an autonomous agent with natural conversational skills allows ECAs to take an active role in teaching and knowledge transmission. This approach integrates theoretical and practical knowledge, with a focus on learning by interactive imitation-modeling. The tutor guides the learner, enabling them to understand, interact, conceptualize, and construct knowledge, ultimately shaping the learner's behavior. In pedagogical ECAs, Rickel and Johnson significantly contributed with "Steve" [15], a task-oriented animated agent for virtual training. Steve interacts convincingly through verbal and non-verbal cues, such as making eye contact, showing actions, and collaborating with learners. Rickel et al. [8] outlined essential ECA capabilities: conducting demonstrations, guiding virtual navigation, using gestures and gaze for focus, providing non-verbal feedback, understanding conversational signals, arousing emotions, acting as a teammate, and adapting to educational needs.

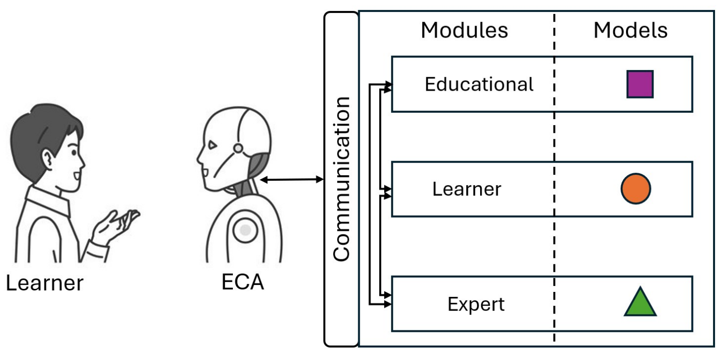

Fig. 1. Classical structure of an ITS. It is made up of three functional modules which communicate with each other: educational, learner and expert. Each module is linked to its model. The ITS, via an interface, communicates with the learner using the ECA as a medium.

The main technique used to equip ECAs with these capabilities is the *Intelligent Tutoring System (ITS)* [12]. This is an expert system responsible for making decisions about the course of the learning activity, in accordance with the learner and the training objectives, among other things. More prosaically, the ITS metaphorically represents the decision-making centre of the ECA. Conversely, the ECA can be seen as the embodiment of the ITS, or even the medium through which the decisions taken by the ITS are communicated to the learner. Typically, and as illustrated in figure 1, an ITS is made up of three modules which work together. Each module is responsible for an aspect of learning and operationalises the decisions of the ITS through the ECA. To achieve this, each of these three modules is associated with a model in which the designers represent, structure and organise the knowledge and rules on which the module is based. The teaching module is responsible for designing and implementing the teaching strategy. Its model describes how information should be presented to the learner, and how its acquisition should be assessed. The learner module collects information about the learner and their learning progress. It is associated with a model that constitutes an abstract representation (evolving over the course of interactions) of the learner's preferences, cognitive style and prior knowledge. Finally, the expert module is responsible for acquiring expertise, by providing answers, explanations and advice pertaining to the task. The collaboration of these three modules enables the ITS to make informed decisions, and to respond in a relevant way to the problems that the learner may encounter during the learning activity. For example, the teaching module can proactively detect that the learner has not understood how a machine or a component works. Based on feedback from the learner, the learner module can determine the causes : gaps in the learner's prior knowledge, or a failure to concentrate on the task. In this case, it can provide multimodal feedback to the learner via the ECA. For example, the ECA will express an annoyed expression on his face, modulate the intonation of its voice, and invite the learner to move closer to the component in question in order to show them again how it works. Unfortunately, the adaptability of the relationship between the ITS and the learner is not reflected in the relationship between the ITS and the learning environment. The models manipulated by the ITS are high-level and often generic. If they take into account the machines and objects in the environment which will be used as the basis for the training, this is only at the initial design stage, and never beyond during the course of the activity. This results in two limitations. On the one hand, the ITS and the environment evolve in a dissociated manner during the learning activity, the pedagogical ECA operating on its own internal representations without the possibility of adapting to changes in the environment resulting from the learner's activity and interactions. Similarly, modulating the difficulty of a manipulative task is very complex if it has not been planned beforehand at the design stage, regardless of the extent to which it is deemed appropriate by the ITS during the activity. On the other hand, the specific design of the ITS for a given environment introduces a coupling between these two components, with any modification to the environment requiring a return to the ITS design phase.

ITS-driven pedagogical ECAs are primarily designed to effectively manage the relationship with the learner, at a dialogical, non-verbal or emotional level. However, they are not anchored in the environment in the same way as simpler agents whose behaviour emerges from interactions with the environment and the objects that make it up. In the next section, we propose an extension to the concept of *smart object* which enables a pedagogical agent to build a reasoning in an adaptive way, based on information extracted from the informed environment.

3 Pedagogical *Smart*-Objects

The usual scripting of virtual training aims to reproduce the learner's protocol or professional activity, with the aim of being able to nominally follow a path mapped out by the designer. However, the acquisition of real expertise requires, in addition, the mobilisation of knowledge and skills to manage the disruptions, degradations and contingencies that may occur during the training. The scenario must therefore be able to help the learner construct mental schemes by promoting an active approach on their part, in accordance with the principles of experiential learning [10]. These patterns are thus constructed in a personalised way, following the opportunities the learner encounters while dealing with various situations. They are then reused in the nominal activity, problem solving and optimisation strategies that make up expertise. Adaptability in experiential learning is defined as the ability to customise, among other things, the difficulty and granularity of the tasks and exercises that make up the training content. This adaptability concerns the user's interactions with the learning environment, whether at the start of the scenario or dynamically as the user progresses. Logically, teaching companions or tutors should also be able to adapt to the different levels of complexity of the training. However, we have shown in the previous paragraphs that the architecture of an ITS does not allow the ECA to respond satisfactorily to this problem. On the other hand, we find that non-pedagogical autonomous agents, those that have no purpose other than populating the virtual environment (see introduction), show much better adaptability in a dynamic environment. The reason for this lies in their behavioural engines, which are simpler (finite state machines, behaviour trees) but take advantage of the informed environment and the *smart objects* in which useful knowledge is distributed. In the research work reported in this article, we therefore hypothesise that it is possible to extend the concept of *smart objects* to the perimeter of pedagogy.

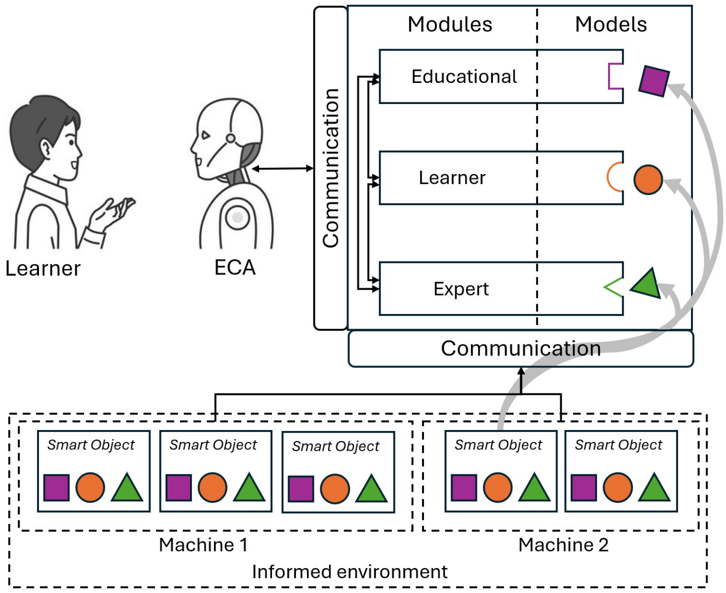

Fig. 2. The proposal of pedagogical smart objects makes it possible to distribute educational, learner and expert models in the informed environment. The operation of the ITS modules remains unchanged, but the models on which they are based are now extracted contextually from the objects with which the learner interacts around him.

In our proposal, each component of a machine is represented by a pedagogical *smart object*, which integrates a set of pedagogical metadata with its graphical representation and the interaction possibilities pre-existing within the *smart object* in its original form. These are of various kinds. They include detailed instructions on appropriate didactics, helping the ITS to emphasise a specific point in the learning of the element in question, or even proposing different ways of presenting or explaining how it works. These instructions have several levels of granularity, ranging from a simple demonstration of the part and its operation to the manipulation of the 3D model in order to discover all its subtleties in detail. They also detail the hierarchical organisation of the machine into subcomponents (themselves pedagogical *smart objects*), as well as the operational mechanism of each of them. They list the potential faults that could affect them and describe the corresponding troubleshooting methods. The distribution of information in the environment introduces an unprecedented modularity in the way the ITS is integrated into the learning environment. Pedagogical and expert models lose their monolithic and rigid character in favour of a new organisation that is more flexible and adaptive to dynamic changes in context. As a corollary, this new organisation adds the new possibility of substituting one element for another in order to play on the granularity of the complexity of the environment, and consequently adjust the difficulty of the training.

In addition, the pedagogical *smart objects* keep track of the history of their personal relationship with the learner. Each object possesses and maintains a record of past interactions, including the length of engagement, the frequency and precision of interactions, mistakes made, remedies or advice already given, and so on. In short, each element of the environment, machine or sub-component of the machine (a handle, a button, a motor, an analogue or digital dial, a control panel, etc.) is therefore able to inform the ITS of its past relationship with the learner. These data is communicated to the ITS to enable adaptation of its teaching strategy seamlessly, without requiring the use of modules other than those that already work well in the traditional ITS. Fig. 2 illustrates the fact that the modules of the ITS are kept identical, but the models on which it operates are replaced by models distributed in the objects, which constitutes the core of the contribution of this research work.

In practice, the ITS is initially configured without any pre-existing data in its various modules. This gives the flexibility needed to obtain the models at the start of the activity, or even during the activity if the environment changes. To do this, it can initiate requests to collect information from the machines and components involved in the training. The data collected are then integrated into the ITS and constitutes the models on which the ITS operates and makes decisions to control the learning activity, role for which it is initially intended. In practice, the virtual environment is programmed using the Unity 3D game engine. The objects that make it up, in the form of 3D meshes, are transformed into smart objects by including a script to map their pedagogical metainformation and affordances. A communication protocol enables the ITS to retrieve these data from the smart objects in real time and use them to take decisions. This distributed, modular architecture means that ITS operation can be adapted to the informed environment, without the need for code modifications or human intervention. This configuration offers the advantage of transforming the ECA into a versatile instructor, capable of teaching and manipulating various machines as long as the appropriate architecture is respected. This allows adherence to the specific teaching protocol of each machine, whether this involves practical demonstrations and on-the-fly tests (sections 4.1 and 4.2) or the diagnosis of a specific failure (section 4.3), thus ensuring a consistent, precise and scalable teaching methodology. Three examples presented in the next section illustrate the practical benefits of the pedagogical *smart objects* architecture.

4 Case Study

The following case studies are set in a fully-immersive Virtual Reality (VR) training simulation where learners are trained to operate a parcel unloader in a virtual warehouse. These introductory scenarios guide learners through the virtual environment, its controls, and the procedure for unloading a trolley of parcels onto a conveyor belt. Throughout the training, a virtual agent named Emilie supports the learning process by guiding the learner, demonstrating machine operations, and explaining their states. Emilie utilizes the distributed architecture of pedagogical *smart objects*, acting as a medium for the environment to deliver educational content naturally through speech, non-verbal behaviors, and emotions. The case studies highlight the functionalities of the ECA's three models: educational, learner, and expert.

4.1 Enforce an Active Pedagogy From the Educational Model

After introducing the proper handling of a component, Emilie offers on-the-fly tests to assess the learner's understanding, updating the models accordingly. The pedagogical *smart object*, such as the LED signal tower (see Fig. 3 left), embeds teaching content and evaluative questions. Emilie selects questions from this pool, tracking answers to enrich its pedagogical metadata. Before moving on to the next part of the training, Emilie asks the trainee if they need her to repeat what she just covered. Should the learner agree, Emilie requests the *smart object* to provide an alternative explanation. The first repetition typically consists in a brief summary of the original content. However, repeated requests will be interpreted as an incapacity to understand and will lead to the delivery of increasingly detailed explanations. As the training progresses, when Emilie needs to check the learner's level of attention with live quizzes, the parcel unloader will present a prioritised set of questions. Priority will be given to questions that the learner has answered incorrectly in the past, or questions relating to explanations that caused problems and have had to be repeated.

If the learner becomes inattentive, Emilie first responds by displaying negative expressions, such as an angry face and a lower-pitched voice tone (see Fig. 3 right). If the inattentiveness persists, Emilie pauses the training and asks the learner if everything is alright or if they would prefer to take a short break. Emilie adapts not only to the learner's level of attention but also to their emotional and cognitive states, using data from biometric sensors mounted on the VR headset. Emotion are classified by using a neural network trained on the eye-tracking dataset VREED [17] and the cognitive load is calculated by doing inference on a neural network given by the VR Headset constructor (HP Reverb G2 Omnicept edition). If the learner's cognitive load rises higher than 0.8 or if they exhibit repetitive negative emotions, such as anger or stress, Emilie may pause the training and check in with the learner, offering the option to take a break outside the simulation.

Fig. 3. Emilie showing the lights of the parcel unloader and explaining their signification (left). Emilie, angry at the user's lack of attention (right).

4.2 Adapting the Content to the Learner Model

In this section we illustrate how the learning model can be leveraged to provide adaptive training based on what is known or assumed about the learner's actual knowledge. In the scenario, the learner is instructed to start the conveyor belt on two occasions. The first time, as Emilie familiarises the learner to the general startup procedure of the parcel unloader, the command panel of the conveyor belt is introduced to them with a simplified layout to avoid overwhelming (see Fig. 4 left). Starting the belt involves pushing a big red button and watch an LED light up green to indicate that the appliance is operating correctly. On this occasion, the *smart object* of the command panel updates its internal learner model to reflect their newly acquired knowledge. On the second occasion, when the whole system needs to be restarted after the motor has been replaced (see section 4.3), the conveyor belt command panel is replaced by a more realistic version, closer to the objectives of transferring this knowledge to real-life conditions. This time, the learner must check the belt temperature on the LCD screen and set the speed potentiometer correctly according to the parameters of the new motor (see Fig. 4 right). As a generic agent, Emilie seamlessly adapts her instructional speech to this new version of the component, receiving content and instructions from the component's *smart object* itself. This example demonstrates how it is possible to replace any component with a more or less complex version, with respect to the level of confidence of the learner. This principle of modularity let us play dynamically with the level of detail of the training.

Fig.4. Simplified command panel (left) versus more realistic depiction (right).

4.3 Carrying out a Diagnosis using the Expert Model

Expert models representing the inner functioning of complex machines benefit as well from the distributed nature of the *smart objects*. The parcel unloader, for instance, consists of components like the tilting platform, conveyor belt, forbidden zone detectors, and LED signal tower. Some of these are recursively composed of lower-order components, down to elementary components that can fail like bearings and motors brushes. In the virtual environment, all of these components are *smart objects* able to communicate upwards and downwards in the hierarchy.

Therefore, once a component breaks, a signal is sent to its parent *smart object* and spreads up the tree until reaching the root parcel unloader. During the process of sending signals, all the related components cease to work and declare themselves out of order, triggering sound effects and realistic visual cues of the specific malfunction to help learners construct a mental scheme of the breakdown. At the end of the introductory scenario, a motor brush is scripted to overheat, causing the motor to break down with the characteristic sound effect of a motor overheating and a visual cue of smoke escaping from the motor (see Fig. 5 left). As soon as the top component parcel unloader is notified by signals travelling up from sub-components below, it notifies Emilie of the problem. Emilie acknowledges the breakdown and verbally informs the learner. Since the resolution procedure is embedded in the pedagogical *smart objects*, she is simultaneously informed of the diagnostic steps to be taken. The troubleshooting process is displayed above Emilie using a tree-like graphical representation of all the possible faulty components with associated cause. Suspects are preceded by an orange square. As a matter of fact, the fault could as well be caused by faulty bearings caused by an excessive load or inadequate lubrication. Yet, Emilie has been informed that the faulty component is the motor, therefore she inspects this component first while explaining why the smoke and the noise led her to put the motor under scrutiny. As shows Fig. 5 right, Emilie isolates the cause of the failure (red squares), and yet carries on investigations. Once the faulty component is identified, Emilie starts the repair, informed yet again of the repairing process by the *smart object* itself. As soon as the component is fixed, messages travelling across the hierarchy of *smart objects* informs all components that everything's back to normal.

Fig. 5. The unloader failure has been triggered by an action in the scenario on an undisclosed component (the electric motor). The learner can visually and audibly notice the problem whereas the companion is alerted by the unloader smart object itself (left). Methodically checking the components enabled the problem to be isolated: a broken motor which was preventing the treadmill from operating (right).

5 Conclusion

Autonomous agents in virtual environments for technical training or cultural heritage are of two types. On the one hand, some of them are simple agents, designed solely to populate or animate the environment, but exhibiting unique capacities for adaptation in a dynamic environment that is constantly changing as a result of the learner's actions. On the other hand, others are designed as genuine conversational teaching agents capable of interacting naturally with the human learner, however offering a limited experience in the face of a dynamic environment. We have proposed an extension of the *smart object* concept, primarily focused on modularity and adaptability of navigation and interaction behaviours, to the perimeter of the pedagogical relationship with the learner. We have developed a pedagogical agent capable of reasoning from information extracted from the environment itself, using an ITS enriched by pedagogical content disseminated in the objects taking part in the training. Our contributions consist of a more flexible ITS architecture, which responds to the general problem of assistance or apprenticeship in an immersive training environment by offering greater adaptability in a dynamic context, but also greater modularity by allowing transparent substitution of objects, components or machines. Additionally, this distributed architecture facilitates the ECA's ability to instruct multiple learners on the operation of various machines within a multi-player training environment.

Future work will aim to remedy the manual definition of pedagogical content (dialogues, explanations, live surveys, etc.) by using an LLM pre-trained on a corpus of technical manuals and guides, and operating on the basis of the internal states of the *smart objects* and the variables of their internal models.

Acknowledgements

This research work is being funded as part of the Campus des Metiers et des Qualifications d'Excellence Industrie´ du Futur (CMQE IF) project, led by the La Decouverte high school in Decazeville; winner of the call for projects´ from the Programme d'Investissement d'Avenir (PIA 3) "Territoires d'innovation pedagogique – Campus des´ metiers et des qualifications". This research is co-financed by ALFI Technologie.

References

1. Abaci, T., Ciger, J., Thalmann, D.: Planning with smart objects. In: International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision. No. CONF (2005).
2. Cassell, J., Bickmore, T., Campbell, L., Vilhjalmsson, H., Yan, H.: Human conversation as a system framework: Designingembodied conversational agents. Embodied conversational agents pp. 29–63 (2000).
3. Cernˇ y, M., Plch, T., Marko, M., Gemrot, J., Ondr` a´cek, P., Brom, C.: Using behavior objects to manage complexity inˇ virtual worlds. IEEE Transactions on Computational Intelligence and AI in Games 9(2), 166–180 (2016)
4. Edward, L., Lourdeaux, D., Barthes, J.P.A., Lenne, D., Burkhardt, J.M.: Modelling autonomous virtual agent behaviours` in a virtual environment for risk. Int. J. Virtual Real. 7(3), 13–22 (2008).
5. Fung, P., Dey, A., Siddique, F.B., Lin, R., Yang, Y., Wan, Y., Chan, H.Y.R.: Zara the supergirl: An empathetic personality recognition system. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations. pp. 87–91 (2016).
6. Getuli, V., Capone, P., Bruttini, A., Sorbi, T.: A smart objects library for bim-based construction site and emergencymanagement to support mobile vr safety training experiences. Construction Innovation (2021)
7. Gibson, J.J.: The theory of affordances. the ecological approach to visual perception. The people, place and, space readerpp. 56–60 (1979).
8. Johnson, W.L., Rickel, J.W., Lester, J.C., et al.: Animated pedagogical agents: Face-to-face interaction in interactive learning environments. International Journal of Artificial intelligence in education 11(1), 47–78 (2000)
9. Kallmann, M., Thalmann, D.: Direct 3d interaction with smart objects. In: Proceedings of the ACM symposium on Virtualreality software and technology. pp. 124–130 (1999).
10. Kolb, D.: Experiential Learning: Experience as the Source of Learning and Development. Prentice-Hall (1984),https://books.google.fr/books?id=zXruAAAAMAAJ
11. Ma¨ım, J., Haegler, S., Yersin, B., Muller, P., Thalmann, D., Van Gool, L.: Populating Ancient Pompeii with Crowds of¨ Virtual Romans. In: 8th International Symposium on Virtual Reality, Archeology and Cultural Heritage - VAST (2007).
12. Mousavinasab, E., Zarifsanaiey, N., R. Niakan Kalhori, S., Rakhshan, .: Intelligent tutoring systems: a systematic reviewof characteristics, applications, and evaluation methods. Interactive Learning Environments 29(1), 142–163 (2021).
13. Peters, C., Dobbyn, S., Nameee, B.M., O'Sullivan, C.: Smart objects for attentive agents. In: proceeding of the 11thInternational Conference in Central Europe on Computer Graphics, Visualization and Computer Vision'2003 (WSCG)(2003).
14. Provoost, S., Lau, H.M., Ruwaard, J., Riper, H.: Embodied conversational agents in clinical psychology: a scoping review.Journal of medical Internet research 19(5), e151 (2017).
15. Rickel, J., Johnson, W.L.: Steve: An animated pedagogical agent for procedural training in virtual environments. Intelligentvirtual agents, Proceedings of Animated Interface Agents: Making Them Intelligent pp. 71–76 (1997).
16. Shao, W., Terzopoulos, D.: Autonomous pedestrians. Graph. Models 69(5-6), 246–274 (2007).
17. Tabbaa, L., Searle, R., Bafti, S.M., Hossain, M.M., Intarasisrisawat, J., Glancy, M., Ang, C.S.: Vreed: Virtual realityemotion recognition dataset using eye tracking & physiological measures. Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies 5(4), 1–20 (2021).