Supporting Situated Learning for Virtual Communities of Practice: Representation and Management of Situated Knowledge

Yongwu Miao
Institute for Computer Science and Interactive Systems
University Duisburg-Essen, Germany
miao@informatik.uni-duisburg.de

Abstract

This paper presents an approach to develop a web-based, multi-user, 3D simulation learning environment supporting virtual communities of practice. This approach emphasizes helping geographically distributed learners to acquire and construct knowledge by doing shared “authentic” activities, to interact with each other as community members, and to perform informal, unstructured, spontaneous, and situation-oriented learning. This paper focuses on description of representation and management of situated knowledge and of support for situated instruction.

1. Introduction

In the past decade, many researchers and educational practitioners believed that virtual reality (VR) and computer simulation technologies offer strong benefits that can support education, especially, facilitate constructivist learning activities [14]. VR and computer simulation both provide an authentic context and allow students to experience a broad range of real-life situations, objects, and phenomena in the simulated world. The use of VR and computer-based simulation has been proposed to help learners successfully to transfer knowledge acquired in a simulated situation to the corresponding real-life situation [5].

Currently, educational VR and computer-based simulation systems (e.g., car-driving, assembling or manipulating a complex mechanical device, conducting a surgical operation, and so on) are often developed for training individuals or teams. The training programs are often formal and structured. The trainee has to follow a strict program and experience pre-defined scenarios [1, 12, 13]. The focus of these simulation systems for training is on the fidelity of the renderings and the accuracy of the simulated behaviors.

This paper presents an initial attempt in another direction applying VR and computer simulation technologies in education. It focuses on the implementation of a new teaching approach in a simulation environment, a pedagogy based on situated learning theory. The new approach is to develop a web-based, multi-user, low-cost, 3D simulation learning environment, which supports people in virtual communities of practice to perform informal, unstructured, spontaneous, collaborative, situated learning, rather than individuals or teams to conduct formal, structured, intensive, programmed instruction.

2. Situated Cognition and Learning

Situated cognition and learning is the study of cognition within its natural context. This perspective emphasizes that individual minds usually operate within environments that structure, direct, and support cognitive processes. In the view of situated learning, knowledge and understanding is fundamentally a product of the learning situation and the nature of the learning activity [9]. Brown, Collins, and Duguid [3] argue that knowledge indexes the situation in which it arises and is used. The embedding circumstances efficiently provide essential parts of its structure and meaning. Jonassen [7] points out that the most effective learning contexts are those which are problem or case based and activity oriented, that immerse the learner in the situation requiring him or her to acquire skills or knowledge in order to solve the problem. The emphasis on the physical, environmental, and social contexts for cognition was termed situatedness/embeddedness by Lucy Suchman [10].

Wenger uses the term of communities of practice to describe the impact of situated learning [11]. A community of practice is a group of people who work...
The major feature of DRIVE is most aptly defined in dimension, driving simulation system, named DRIVE. DRIVE has been developed to build a web-based, multi-user, 3-dimensional driving simulation system. Communities of Practice

3. Support for Situated Learning in Virtual Communities of Practice

Based on situated learning theory, an approach has been developed to build a web-based, multi-user, 3-dimensional driving simulation system, named DRIVE. The major feature of DRIVE is most aptly defined in the word “community”, rather than “simulator”. DRIVE focuses on helping people to learn traffic knowledge and obtain basic experience to deal with situations in a virtual community.

DRIVE uses a “driving school” as a structural metaphor, which consists of an administrative building, a set of instructional buildings, and a set of virtual driving places. The administrative building contains several rooms where administrative functionality is distributed such as registration, learning material editor, question/test editor, avatar and vehicle definition and selection, driving place editor, situated knowledge editor, and 2D driving place monitor. Each instructional building contains a number of classrooms. Each classroom is a shared space that contains single-user tools or shared application tools such as searching engine, 3D graphics viewer, text-based chat, audio tool, shared whiteboard, and shared text editor. A driving place is a shared space containing a set of static objects such as road, buildings, and trees, and dynamic objects such as traffic and pedestrians (avatar).

A user reaches a virtual driving school by accessing the home page of the virtual driving school. The user can enter the virtual school if s/he has registered and logged in correctly. Users can enter the administrative building to select a vehicle and an avatar. Only authorized users can do constructive work such as editing learning material, defining questions and tests, defining a driving place, and defining situated knowledge. It is important to note that a driving place is defined by specifying only static objects besides traffic lights. In DRIVE it is not necessary to design traffic scenarios done in other driving simulators. Users can do learning activities in each classroom such as search and read online reading material, do exercise/test, search and replay accident cases, talk, discuss, analyze, lecture, and so on by means of tools equipped in the classroom.

Users can enter a virtual driving place where s/he can control the behavior of the selected vehicle or the selected avatar moving within the virtual driving place. A user can also enter an existing vehicle in the driving place if getting permission. Users inside the same vehicle have the same view of the driving place and only one user can control the vehicle at any point of time. However, they can change the control right between them according to a role-based floor control policy. Users will take a certain role (e.g., a driver, a coach, or an observer) and they can delegate and change their roles. Unlike in real world, unlimited users can “sit” inside the same virtual vehicle. The driver can decide whether to close or open the virtual door. All users inside a vehicle can coordinate and...
communicate by using an audio tool. A driver can establish an audio connection as well with the driver in vehicles near him.

If a user has no experience, s/he can ask for demonstration. DRIVE will assist users to control the vehicle or the avatar, and let users observe correct and professional behavior. If a user asks for guidance and coaching, DRIVE will provide appropriate instruction according to the current situation and the user’s state. Otherwise, DRIVE will keep “quiet”. Furthermore, if a user requires, DRIVE can actively provide adaptive guidance or coaching in situations which the user has never encountered before, or has not handled correctly. In addition, if an event such as a collision occurs or an interesting situation happens, a user is allowed to save the episode as a case which can be “replayed” and analyzed later in classrooms.

4. Provision of Situated Instruction

DRIVE has been briefly described in section 3 from a user’s perspective. This section describes how to provide situated instruction in DRIVE. As illustrated in Fig. 1, five components are involved in supporting situated instruction: simulation module, perception module, situated knowledge base, community module, and instruction module.

![Fig. 1: Architecture to support situated instruction](image)

The simulation module is a software component which simulates a microcosm, called virtual driving place. A virtual driving place is composed of a set of objects. The objects have various state variables, and at a given point of time, each state variable has a value. The combination of the values of the state variables of all objects is the state of the virtual driving place at a given time. As mentioned before, in a virtual driving place some objects are static such as road, traffic sign, and building. Traffic lights may change the status according to the parameters assigned. Vehicles, bicycles, and pedestrians are active and controlled normally by users. This module receives user input events to control the vehicle or the avatar (e.g., speed up/walk, brake/stop, look back, change direction, switch on/off lights, open/close virtual door, and so on), and to get commands (e.g. ask for guidance, express intention, need demonstration, and so on). The events will be processed appropriately. The encoded actions coming from instruction module will be handled in the same way (see detailed description later). Because various physical phenomena are reified in the simulation through the causal relations between the events in the world, an event may trigger other events in a successive manner. Obviously, users’ behavior is unpredictable. Consequentially, the change of the state of the virtual driving place is undetermined. Users of DRIVE have the possibility to experience boundless situations, rather than a fixed flow of pre-defined training scenarios. The simulation module will render a series of 3D images to present the state of the virtual driving place from the perspective of the driver or the avatar. Furthermore, this module maintains the history of states of the virtual driving space.

The role of the perception module is to interpret the current state of the virtual driving place into a context by considering history of state change. A context is a multi-dimensional data object and each dimension represents a contextual factor. A contextual factor (e.g., “my speed”) may be represented directly as a state variable in the virtual driving place, or may be represented as a function of several state variables (e.g., “the distance from my car to the car ahead”) and even historical state (e.g., my car is slowing down). One and only one context can be interpreted from a state of the virtual driving place. At the same time, the interpreted user actions will be sent to community module as record of user performance information.

The role of the situated knowledge base is to manage and retrieve situated knowledge. It judges situations according to the current context and then retrieves situated knowledge. A situation is identified as an opportunity for training by considering several contextual factors. For example, “over speed” is a simple situation which is judged according to two factors: my speed and allowed speed. A complex situation is judged by considering many contextual factors. A situation is used to represent a family of contexts or specialization of a context. In the former case several contexts belong to the same situation; in the latter case several situations correspond to the same context. It is worth to note that a situation may contain other situations (sub-situations). In order to get situated knowledge, it is needed to judge which situations there are in the current context. A situated knowledge denotes the knowledge which should be taught and
applied in a certain situation. A representation of knowledge contains encoded correct actions for autonomous vehicles, coaching material for human users, and typical cases stored and analyzed by users.

The role of the community module is to manage users’ information. Useful information for providing situated instruction includes performance, preference, and profile information provided by the user at registration and captured by DRIVE when the user performs the task. DRIVE assumes any user can handle any situation well. Only if a user fails to handle a situation, DRIVE will assess user’s skills as several degrees and store the evaluation associated with the situation.

The instruction module is a software component to provide instructions to users. This module will decide what content should be presented to the user in which form of instruction and in which kind of media. This module gets a set of situations and a set of knowledge representation from the situated knowledge base, and then refers to the user information associated with these situations. If the user has no record in a situation, the knowledge associated with this situation will not be considered. Otherwise, DRIVE chooses appropriate content according to the evaluation to the user and the user’s preference information. Finally, this component will determine the form of instruction (e.g., warning, hint, comment, recommendation, detailed explanation, and so on) and the kind of media and means (e.g., text-based message or augmented graphics in the simulator window, web page with multimedia elements, or prepared verbal instruction) according to the user information and features of the selected context. If the user wants to observe the demonstration, the encoded correct actions for an autonomous vehicle or an avatar will be selected and passed to the simulation module, which will decode and control the vehicle or the avatar and behave like an autonomous vehicle or an avatar.

Finally, the process to provide situated instruction is briefly described in a more formal way. Given at time \( t \), a state of the virtual driving place is represented as state \( s(t) = (x_{10}, x_{20}, \ldots, x_q) \), where \( q \) is the number of all state variables of all objects in the virtual driving place at time \( t \) and \( x_i \) is the value of state variable \( x_i \) at time \( t \). A context space \( C \) is defined as a Cartesian product \( C = \prod_{i=1}^{n} F_i \), where \( F_i = \{ f_{i1}, f_{i2}, \ldots, f_{in_i} \} \). The perception is defined as \( m \) mappings of the domain of all states \( ST \) into a contextual factor, \( m: ST \rightarrow F_i \). Thus, a context \( e = (f_1, f_2, \ldots, f_n) \) is an \( n \)-dimensional data point within the context space \( C \). The individual component \( f_i \) of a context \( e \) is an element of the corresponding contextual factor, \( f_i = f_{i_{n_i}} \). A situation \( s \) is a subset of an \( m \)-dimensional \( (1 \leq m \leq n) \) sub-space (cube or range) within the context space. It can be represented as \( s \subseteq F_1 \times F_2 \times \ldots \times F_m \), where \( \forall j: (1 \leq j \leq m) \); \( \exists i: (1 \leq i \leq n) \). A knowledge representation \( k \) is a XML file. The situated knowledge base is defined as a subset of a Cartesian product \( SKB \subseteq S \times K \), where \( S \) is the set of all situations and \( K \) is the set of all knowledge representations, i.e., a situated knowledge is defined as a pair \((s, k)\), \( s \) is a situation and \( k \) is a knowledge representation. Therefore, the process to retrieve situated knowledge can be described as the following steps: 1) interpret a context \( e \) from the state \( s(t) \); 2) inquire all situations which contain the current context, \( S'_i = \{ s_{i1}, s_{i2}, \ldots, s_{ir} \}| s_i \) contains \( e, 1 \leq i \leq r \}; 3) retrieve all knowledge representations used in the current situations, \( K'_i = \{ k_{i1}, k_{i2}, \ldots, k_{ir} \}| s_i \in S'_i \land (s, k) \in SKB \}, 1 \leq i \leq r \}; 4) create an integrated knowledge representation based on the retrieved knowledge representations \( K' \) and the user model.

5. Implementation Issues

Since 1996 [4, 6], research and development work on web-based simulation has exploded. By using Java-applets, complex and collaborative simulation tools can be implemented on the Web. DRIVE employs Tomcat as Servlet/JSP engine, Xindice server to manage data and knowledge, VRML/Java3D to represent and render 3D models, ISDT [8] to support synchronous cooperation. DRIVE is a distributed multi-user system with server/client architecture. Because of the limited space, this paper focuses on discussing the implementation issues concerning situated knowledge.

A simulation client runs in each user web-browser. The perception module, community module, and instruction module run locally as well. Only the situated knowledge base locates in the server site. This centralized nature allows knowledge to be globally available to all clients and allows to add, delete, and modify knowledge bases in a manner that makes such changes available to all clients and keeps consistency. We developed tools to create, delete, and modify a factor, a situation, and knowledge representation, and to define a context space and bindings between situations and knowledge representations. Rather than some forms of if-then hard code, knowledge and user information are represented in DRIVE as XML files with link to multimedia elements available on the Web. The X-tree [2] is used as index structure for high-
dimensional data objects, which can improve the performance to inquire situations. All functions to represent and manage situated knowledge are implemented as generic facilities, and all domain knowledge is stored in databases and as files. Therefore, the situated knowledge base can be constructed dynamically. The main advantage of this data-centered approach is an ability to change situated instruction without re-compiling the system. For example, if replacing encoded actions in knowledge representation, the system can simulate the behavior of a drunk driver dynamically.

6. Conclusion

Based on situated learning theory, we developed an approach to support situated learning in a web-based, multi-user, 3D simulation learning environment. This paper describes how the principles of situated learning are embodied in the design and implementation of a virtual learning environment. As a prototype system, DRIVE can be characterized as a “community” and a “coach”, rather than a “simulator” and a “teacher”. It helps and assists learners to obtain experience and skills in “authentic” context and “realistic” activities. While performing the activities, learners can receive situated instruction as needed to complete tasks and understand aspects of the domain knowledge, and may collaboratively construct shared knowledge with other community members. In addition, the data-centered approach to represent and manage situated knowledge makes it possible to construct situated knowledge based dynamically.

A simplified version of DRIVE has been developed, which can simulate simple driving places. Our current work is still focusing on testing the usability and feasibility of the system. One test has been conducted by running 18 clients at the same time and the system worked well. In the future, more serious evaluation will be conducted and then a complete version will be implemented. We need to develop a systematic method to decompose the context space into lower-dimensional data spaces in order to improve the performance of inquiring situations. We will use the approach to develop systems to support situated learning in other application domains.

References


