A Framework for Automatic Simulated Accessibility Assessment in Virtual Environments

Nikolaos Kaklanis[1,2], Panagiotis Moschonas[1],
Konstantinos Moustakas[1], and Dimitrios Tzovaras[1]

[1] Informatics and Telematics Institute, Centre for Research and Technology Hellas
Thessaloniki, Greece
[2] Department of Computing, University of Surrey
Guildford, United Kingdom
{nkak,moschona,moustak,Dimitrios.Tzovaras}@iti.gr

Abstract. The present paper introduces a framework that enforces the accessibility of products and services by enabling automatic simulated accessibility assessment at all the stages of the development. The proposed framework is based on a new virtual user modelling technique describing in detail all the physical parameters of a user with disability(ies). The proposed user modelling methodology generates a dynamic and parameterizable virtual user model that is used by a simulation framework to assess the accessibility of virtual prototypes. Experimental results illustrate the use of the proposed framework in a realistic application scenario.

Keywords: accessibility evaluation; user modelling; task modelling; simulation; UsiXML; virtual user.

1 Introduction

The population is ageing: according to the World Health Organisation, the worldwide population of people aged 60 years and over will have increased from 580 million in 1998 to 1000 million in 2020. As global populations are growing older, together with an increase in people with disabilities, the moral and financial need for “designed for all” products and services becomes obvious.

The lack of non accessible products can cause large productivity losses, with many people being unable to fully participate at work, in education, or in a wide range of economic and social activities. People's choice of leisure activities may be narrower than it otherwise could be.

User studies involving users with disabilities often incur greater financial cost and complexity than those involving general populations. Consequently, accessibility issues may not be identified during the earlier phases of product/services design, when designs are still malleable. Additionally, it can be difficult to create controlled studies with multiple groups of very similar subjects due to the extremely heterogeneous nature of the impact of many motor and visual disabilities.

Accessible systems, that is, systems which can be used by users who are disabled as well as by the average user, are challenging to build. Like usability, accessibility is

V.G. Duffy (Ed.): Digital Human Modeling, HCII 2011, LNCS 6777, pp. 302–311, 2011.

a goal that requires iteration, customer-centered design, and significant expertise to be done right. Unfortunately, user testing with special populations often requires greater effort, time, and monetary commitments on the part of designers, developers, and their companies than user testing with the general population. Additionally, it may be difficult to find homogeneous groups of participants.

Effective evaluation of design concepts at an early design stage can help to achieve shorter lead time and reduced costs. With the advent of inexpensive high-speed computing, it has become feasible to verify a design based on a virtual prototype, using modelling and simulation technology. Even if there have been some limited and isolated attempts [10] to support accessibility testing of novel products and applications, there is a clear lack of a holistic framework that supports comprehensively virtual user modeling, simulation and testing at all development stages. The present paper presents a framework that performs automatic simulated accessibility evaluation based on a new user modeling technique. The great importance of such a framework lies to the fact that it enables the automatic accessibility evaluation of any environment for any user by testing its equivalent virtual environment for the corresponding virtual user. Moreover, the main innovation of the proposed framework lies in the fact that, the whole framework is based on a new virtual user modelling technique including physical, cognitive and behavioral/ psychological aspects and parameters of the user with disability. The proposed technique is not constrained in static modelling of the virtual user it but generates a dynamic and parameterizable user model including interactions that can be used directly in simulation frameworks, adaptive interfaces and optimal design evaluation frameworks.

2 Related Work

Virtual users with disabilities fulfill the role of standardized patients by simulating a particular clinical presentation with a high degree of consistency and realism and offer a promising alternative [14]. Different user model representations have been proposed in the literature using different syntaxes and implementations, varying from flat file structures and relational databases to full-fledged RDF with bindings in XML.

OntobUM [21], the first ontology-based user modelling architecture, was introduced by Razmerita et al. in 2003. A similar, but way more extensive approach for ontology-based representation of the user models was presented in [10]. GUMO, probably the most comprehensive publicly available user modelling ontology to date, is proposed in [9].

XML-based languages for user modelling have also been proposed. UserML has been introduced in [8] as user model exchange language. UserML is based on an ontology that defines the semantics of the XML vocabulary (UserOL).

The use of “personas”, which are empirically based abstract descriptions of people has gained popularity and has been developed into a well documented design practice. Pruitt et al. [20] claim that currently there are no explanations of why many prefer to develop personas, instead of focusing directly on the scenarios that describe the actual work processes that the design is intended to support. However, a potential candidate to a theory of using ‘personas’ in design is, according to Pruitt et al. [20] the ‘theory of mind' that says that we, as humans, always use our knowledge of other people's mental states to predict their behavior [2].

In the context of cognitive modelling, SOAR [22] offers a general cognitive architecture for developing systems that exhibit intelligent behavior. It defines a single framework for all tasks, a mechanism for generating goals and a learning mechanism. Another popular cognitive architecture is ACT-R [1], which is based on the distinction between declarative and procedural knowledge, and the view that learning consists of two main phases. In the first phase the declarative knowledge is encoded and in the second it is turned into more efficient procedural knowledge [18].

The idea of simulating disability has been applied to Web accessibility research in the past. For example, IBM's aDesigner [6] renders a Web page so that text or images that cannot be viewed by a person who is blind are impossible to see (covered in black). WebAIM (Web Accessibility In Mind) [27] provides scripted simulations with a number of capabilities, including those that illustrate screen reader use, those that demonstrate the use of screen-enlarging software for users with low vision, and those that simulate distractibility and the accompanying frustrations that a user with cognitive disabilities might experience.

Virtual User Models are also used in application areas such as automotive and aviation design. The RAMSIS human modelling tool [17] is currently used in the design of automobiles in about seventy percent of auto-manufacturers. SAMMIE is a computer aided human modelling system that represents a widely used tool to accommodate the needs of a broad range of differently sized and shaped people into the design of products [23]. HADRIAN [16] is another computer-based inclusive design tool that has been developed to support designers in their efforts to develop products that meet the needs of a broader range of users.

3 Proposed User Modelling Methodology

In order to support the automatic accessibility testing of ICT and non-ICT services and products, it is essential to create machine-readable descriptions of the disabled user and the tasks that the user is able to perform, as well as formal descriptions of the products and services to be tested. In [11] the concepts of Virtual User Model, Task Model and Simulation model (all expressed in UsiXML format) have been introduced. The aforementioned models are used by the core component of the proposed framework, the Simulation Module.

More specifically, the Simulation Module gets as input a Virtual User Model (describing a virtual user with disabilities), a Simulation Model that describes the functionality of the product/service to be tested, the Task Models that describe in detail the complex tasks of the user and simulates the interaction of the virtual user (as it is defined in the Simulation Model) within a virtual environment.

3.1 Virtual User Model

A Virtual User Model [11] describes an instance of a virtual user, including user's preferences, needs and capabilities and is modelled through an extension of UsiXML. It includes all the possible disabilities of the user, the affected by the disabilities tasks, motor, visual, hearing, speech, cognitive and behavioral user characteristics as well as some general preferences (e.g. preferred input/output modality, unsuitable input/output modality, etc.) of the user.

3.2 Task Model

Task models [11] describe the interaction between the virtual user and the virtual environment. User tasks are divided into primitive (e.g. grasp, pull, walk, etc.) and complex (e.g. driving, telephone use, computer use, etc.). For each complex task a Task Model is developed, in order to specify how the complex task can be analyzed into primitive tasks (as they have been defined by the designers/developers according to the functionality of the prototypes to be tested in terms of accessibility). Fig. 1 presents a schematic description of a complex task and the primitive tasks in which it could be analyzed.

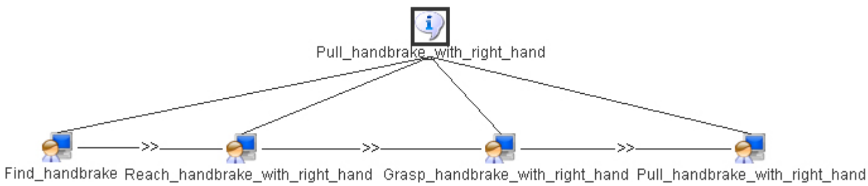


Fig. 1. Task Model example – The root node represents a complex task while the leaf nodes represent primitive tasks. The arrows between the primitive tasks define that the primitive tasks should be executed sequentially.

3.3 Simulation Model

A Simulation Model [11] refers to a specific product or service and describes all the functionalities of the product/service as well as the involved interaction with the user, including all the different interaction modalities supported.

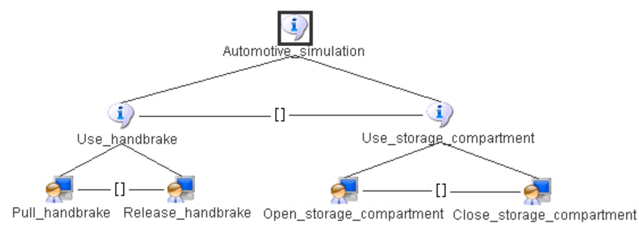


Fig. 2. A Simulation Model example representing some common tasks that can be performed in a car interior. The tasks of the same level are connected with choice relationship, which means that only one of the two can be executed at a time.

Fig. 2 presents a simulation model referred to a car interior. According to this simulation model, the interaction between user and the environment includes the use of the handbrake and the storage compartment.

4 Simulation Framework

4.1 Virtual Environment

The proposed framework performs automatic simulated accessibility evaluation in virtual environments, thus virtual environments have to be developed according to the specifications of the real ones to be tested. The virtual environment is used by the Simulation Module, which is the core module of the proposed framework and performs the evaluation process. Fig. 3 depicts a virtual environment representing a common car interior. It is obvious that the credibility and the accuracy of the simulation results lie on the detailed representation of the virtual environment, which will be given as input to the Simulation Module.

Fig. 3. Virtual environment example representing a common car interior

4.2 Simulation Module

The Simulation Module [11] gets as input a Virtual User Model (describing a virtual user with disabilities), a Simulation Model (describing the functionality of the product/service to be tested), one or more Task Models (describing in detail the complex tasks of the user) and a 3D virtual prototype representing the product/service to be tested. It then simulates the interaction of the virtual user (as it is defined in the Simulation Model) with the virtual environment.

The disabled virtual user is the main "actor" of the physically-based simulation that aims to assess if the virtual user is able to accomplish all necessary actions described in the Simulation Model, taking into account the constraints posed by the disabilities. Even if the proposed framework is designed to include cognitive, behavioral and physical disabilities, in the present work, first results on the modeling and simulation of motor arm disabilities are reported. Simulation planning is performed using inverse kinematics, while dynamic properties of the human limbs (e.g. torques and forces) related to the corresponding actions (e.g. grasping) are obtained using inverse dynamics.

The simulation module is parted of three sub-elements: a) the Scene module, b) the Humanoid module and c) the Task manager module. The scene module is responsible for the representation of the scene and the management of all the objects in it. The humanoid module represents the human model and controls the humanoid's movements. The task manager module defines the interactions between the humanoid and the scene objects in a specific manner defined by the task and simulation models. These sub-modules are further explained in the following sections.

4.3 Scene module

The scene module [11] is responsible for creating the scene, adding the objects in it and defining their special attributes and behavior. The scene is modeled by two sets of objects: static objects and moveable objects. Both types of objects have geometry (volume) and visual representation (textures). Static objects do not have mass and cannot be moved by the humanoid. On the other hand, moveable objects are modeled as rigid bodies, having properties like uniformly distributed mass over their volume (constant density), linear and angular velocity.

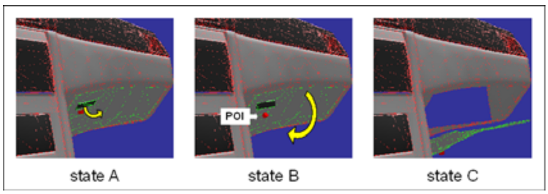


Fig. 4. The three states showing the car's storage compartment functionality. The arrows represent the rotational degrees of freedom. The red box shows the POI, used for interacting with the object. Three objects are presented in the screenshots: the handle (moveable), the storage compartments' door (moveable) and the car's dashboard (static).

For example, the car's storage compartment (Fig. 4) functionality described by two DoF-chains: one that connects the handle with the storage compartment and another that connects the storage compartment with the dashboard.

In the previous example, a scene rule is used to check at every timestep the state of the handle. If the angle to its parent (i.e. compartment's door) exceeds the predefined limit, the storage compartment opens by a spring force.

4.4 Humanoid Module

The humanoid [11] is modeled by a skeletal model that is represented by a set of bones and a set of joints. The skeletal model is following a hierarchical approach, meaning that each joint connects a child bone with its parent one. Basically, the bones are modeled by rigid bodies, having properties like mass, volume, position, velocity etc. The mass is distributed uniformly in the bone's volume. A primitive geometry shape (box, sphere or capsule) is attached to each bone, which is representing the human part flesh and is used for collision testing purposes.

The joints induce or restrict the movement of their attached bones and have properties such as: a) *rotational degrees of freedom*: define the rotational axis in which the attached child bone is able to rotate relatively to its parent. Every joint can have one to three degrees of freedom, b) *minimum and maximum angle per degree of freedom*: these two angles constrain the range of motion of the attached bones (and their body parts), c) *minimum and maximum joint torque*: an abstract representation of the muscles attached to the joint.

Currently, the model supports two basic modes: kinematic and dynamic. In kinematic mode, the humanoid moves by directly changing the position and orientation of each bone part per time step. In dynamic mode, the humanoid changes its state, by applying torques at each joint. As above, forward and inverse dynamic techniques [5] are supported.

Various high level motion planning and collision avoidance techniques are supported by the humanoid module. Configuration space [15] and structures that define and explore it, such as rapidly exploring random trees [13] and multi-query planners [24] are extensively used, in order to compute a non colliding path for the humanoid.

4.5 Task Manager Module

The task manager module [11] is responsible for managing the actions of the humanoid in order to provide a solution to a given task. After splitting the complex task to a series of primitive tasks, the task manager instructs the humanoid model to perform a series of specific actions in order to accomplish them. At every step, the task manager supervises and checks for task completion and reports to the system if something went wrong. The cycle pattern that is followed at each simulation step in the dynamic mode is presented in [11].

In order to decrease the complexity of each primitive task and its success/failure condition, the task manager uses a system of task rules. Each primitive task can have a set of rules that are checked at each timestep. Following the same rule model as in the scene module, each rule has two main parts: condition part and result part. When a condition is met, the rule's result part is applied. Conditions can check various simulation's elements and states, such as current distance of a specific bone from a POI, measure time since task started, count how many times a specific humanoid action was performed etc.

5 Experiments

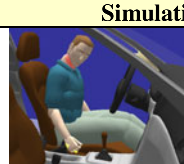
In order to show how the proposed framework could be used in practice, an evaluation scenario is presented in this section. According to the scenario, a car designer performs accessibility evaluation of a prototype for different virtual users with disabilities.

The designer initially develops the virtual workspace prototype presented in Fig. 3. Then, using the Virtual User Model Editor described in [11], the designer generates two Virtual User Models, whose problematic physical parameters are presented in Table 1. Two Task Models describing the handbrake use and the opening of the storage compartment, respectively, were developed in the context of the experimental scenario (in a similar way such as the Task Model presented in Fig. 1). Finally, a simple Simulation Model describing the interaction of the Virtual User in the specific virtual environment was developed (in a similar way such as the Simulation Model presented in Fig. 2).

Table 1. Virtual User Models - Details

Physical characteristics	**Normal Values** [4][12]	**Elderly (60-84)** [12][25][3][26]	**Spinal Cord Injury** [7]
Hand maximum pull force (N)	335	76.8	
Wrist radial deviation (°)	0 – 27.5	0 – 19	
Wrist ulnar deviation (°)	0 - 35	0 – 26	
Forearm supination (°)	0 - 85	0 - 74	
Forearm pronation (°)	0 - 85	0 - 71	
Elbow hyper-extension (°)	0 - 10	0 - 4	
Shoulder flexion (°)	0 - 160		0 - 86
Shoulder abduction (°)	0 - 85	0 - 67	0 - 21
Shoulder internal rotation (°)	0 - 80	0 - 63	
Shoulder external rotation (°)	0 - 45		0 - 12
Spinal column flexion (°)	0 - 90	0 – 23.6	
Spinal column extension (°)	0 - 30	0 - 17	
Spinal column left lateral flexion (°)	0 - 25	0 - 19	
Spinal column right lateral flexion (°)	0 - 25	0 - 20	

Table 2. Simulation results

Task	**Virtual User**	**Scenario**	**Simulation result**	
Pull handbrake	Elderly (60-84)	Handbrake resistance (torque) : 17Nm		Failure – Reduced hand pull force
		Handbrake resistance (torque) : 6Nm		Success
Open storage compartment	Spinal Cord Injury	Storage compartment that opens by pulling a handle		Failure – Reduced range of motion
		Storage compartment without handle that opens by pushing it		Success

The handbrake use was simulated for an elderly user (60-84 years old) in two different scenarios. In the first scenario, the handbrake (mass 200gr) had a resistance (torque) of 17Nm, which loosely speaking it can be translated into lifting a weight of about 4.8kgr, while in the second scenario the resistance was 6Nm (~1.7kgr).

Furthermore, the storage compartment's opening was simulated for a virtual user with spinal cord injury in two different scenarios. In the first scenario, the opening of the storage compartment could be performed by pulling a handle, while in the second scenario it could be performed by pushing the whole storage compartment.

The results of the simulation process clearly revealed accessibility issues of the specific car interior. More specifically, as depicted in Table 2, the elderly virtual user failed in handbrake use when resistance was 17Nm and the virtual user with spinal cord injury failed in opening the storage compartment with the handle, which means that the specific environment is inaccessible for them.

6 Conclusions

The present paper proposes a framework that performs automatic simulated accessibility testing of designs in virtual environments using a new user modelling technique. The great importance of such a framework lies to the fact that it enables the automatic accessibility evaluation of any environment for any user by testing its equivalent virtual environment for the corresponding virtual user.

Acknowledgment. This work is supported by the EU funded project VERITAS (FP7 – 247765).

References

[1] Anderson, J.R.: Cognitive Psychology and its Implications, 5th edn., p. 241. Worth Publishers, N.Y (2000)

[2] Astington, J.W., Jenkins, J.M.: Theory of mind development and social understanding. Cognition and Emotion 9, 151–165 (1995)

[3] Berryman-Reese, N., Bandy, W.: Joint Range of Motion and Muscle Length Testing. Saunders, Philadelphia (2002)

[4] Debrunner, H.: Orthopaedic Diagnosis. E & S Livingstone, Edinburgh (1970)

[5] de Jalon, J.G., Bayo, E.: Kinematic and Dynamic Simulation of Multibody Systems. In: The Real-Time Challenge, p. 440. Springer, New-York (1994); ISBN 0-387-94096-0

[6] Dvorak, P.: Looking for a User-Friendly Internet. The Wall Street Journal (September 16, 2004)

[7] Eriks-Hoogland, I.E., de Groot, S., Post, M.W.M., van der Woude, L.H.V.: Passive shoulder range of motion impairment in spinal cord injury during and one year after rehabilitation. Journal of Rehabilitation Medicine 41(6), 438–444 (2009)

[8] Heckmann, D.: Introducing situational statements as an integrating data structure for user modeling, context-awareness and resource-adaptive computing. In: ABIS 2003, Karlsruhe, Germany, pp. 283–286 (2003)

[9] Heckmann, D.: Ubiquitous User Modelling. Akademische Verlagsgesellschaft Aka GmbH, Berlin (2006); ISBN 3-89838-297-4, ISBN 1-58603-608-4

[10] Heckmann, D., Schwartz, T., Brandherm, B., Schmitz, M., von Wilamowitz-Moellendorff, M.: GUMO – The General User Model Ontology. In: Ardissono, L., Brna, P., Mitrović, A. (eds.) UM 2005. LNCS (LNAI), vol. 3538, pp. 428–432. Springer, Heidelberg (2005)

[11] Kaklanis, N., Moschonas, P., Moustakas, K., Tzovaras, D.: Enforcing accessible design of products and services by enabling automatic accessibility evaluation. In: CONFIDENCE 2010 (2010)

[12] Kumar, S.: Muscle Strength. CRC Press, Boca Raton (2004)

[13] Lavalle, S.M., Kuffner, J.J.: Rapidly-Exploring Random Trees: Progress and Prospects. Algorithmic and Computational Robotics: New Directions (2000)

[14] Lok, B., Rick, F., Andrew, R., Kyle, J., Robert, D., Jade, C., Stevens, A., Lind, D.S.: Applying Virtual Reality in Medical Communication Education: Current Findings and Potential Teaching and Learning Benefits of Immersive Virtual Patients. Virtual Reality 10, 185–195 (2006)

[15] Lozano-Pérez, T.: Spatial planning: A configuration space approach. IEEE Transactions on Computing C-32(2), 108–120 (1983)

[16] Marshall, R., Porter, J.M., Sims, R.E., Summerskill, S., Gyi, D.E., Case, K.: The HADRIAN approach to accessible transport. Work, A Journal of Prevention, Assessment & Rehabilitation 33(3) (2009) ISSN 10519815, doi:10.3233/WOR20090881

[17] van der Meulen, P., Seidl, A.: Ramsis – The Leading Cad Tool for Ergonomic Analysis of Vehicles. In: Duffy, V.G. (ed.) HCII 2007 and DHM 2007. LNCS, vol. 4561, pp. 1008–1017. Springer, Heidelberg (2007)

[18] Mitrovic, A., Koedinger, K.R., Martin, B.: A Comparative Analysis of Cognitive Tutoring and Constraint-Based Modeling. In: Proceedings of 9th International Conference on User Modelling 2003, USA, pp. 313–322 (2003)

[19] Palmon, O., Oxman, R., Shahar, M., Weiss, P.L.: Virtual environments as an aid to the design and evaluation of home and work settings for people with psysical disabilities. In: Proc. 5th Intl Conf. Disability, Virtual Reality & Assoc. Tech., Oxford, UK (2004)

[20] Pruitt, J., Grudin, J. (2003). Personas:Practice and Theory. In: Proc. DUX (2003), http://research.microsoft.com/en-us/um/people/jgrudin/

[21] Razmerita, L., Angehrn, A., Maedche, A.: Ontology-based user modeling for Knowledge Management Systems. In: Proceedings of UM 2003 User Modeling: Proceedings of the Ninth International Conference, USA (2003)

[22] Rosenbloom, P.S., Laird, J.E., Newell, A.: The Soar Papers: Research on Integrated Intelligence. MIT Press, Cambridge (1993)

[23] SAMMIE, SAMMIE CAD Ltd. (2007), http://www.sammiecad.com/

[24] Siciliano, B., Khatib, O.: Springer Handbook of Robotics. Springer-Verlag New York, Inc., Secaucus (2007)

[25] Steenbekkers, L.P.A., van Beijsterveldt, C.E.M. (eds.): Design-relevant characteristics of ageing users Series Ageing and Ergonomics, vol. 1. Delft University Press, Delft (1998)

[26] Trudelle-Jackson, E., Fleisher, L.A., Borman, N., Morrow Jr., J.R., Frierson, G.M.: Lumbar Spine Flexion and Extension Extremes of Motion in Women of Different Age and Racial Groups. Spine (Phila Pa 1976) 35(16), 1539–1544 (2010), doi:10.1097/BRS.0b013e3181b0c3d1.

[27] WebAIM Simulations, WebAIM, Center for Persons with Disabilities, Utah State University, http://www.webaim.org/simulations/