A Multi-Touch Based Edutainment Demonstrator for Children

Hans C. Arlt[1], Alexander Ehrlich[2], Peter Eschler[2], Cornelius Malerczyk[1]

[1] Technische Hochschule Mittelhessen, University of Applied Sciences, Friedberg

[2] New Media Yuppies, Kreativagentur für Neue Kommunikation, Frankfurt am Main

Abstract

This paper describes the concept and its prototypical implementation of a multi-touch based edutainment demonstrator designed especially for young computer users. Assuming that children learn the most effective, if learning and knowledge transfer is hidden behind game interaction, we propose a game application, that force fun and joy during interaction with a virtual character. Especially interacting with an avatar leads to the possibility of an intuitive communication between user and application. As the narrative element a virtual character of a space traveler is chosen that is visiting our solar systems and learning about Earth, Moon and Mars. Both 3D world and character are designed in a classical comic style to attract children especially between the ages of six and twelve. The usage of motion captured animation data and real-time motion blending ensures realistic looking and credible movements of the avatar. While this paper describes the concept and implementation of the demonstrator only, a later following evaluation phase will have to proof the learning effect of the demonstrator.

COMPUTER GRAPHICS, MULTI-TOUCH, EDUTAINMENT, GAME DESIGN, VIRTUAL CHARACTER, 3D REAL-TIME RENDERING

Introduction

In opposite to a large number of adult people, children are more often impartial towards using new technology. Therefore, the idea of this demonstrator is to use a multi-touch table as an eye catcher to attract young users. But using an interesting technology should not be the main goal of an edutainment application. Moreover, children should use the technology intuitively and learn while playing a game.

Figure 1: Space traveler *Newtrix* waving to the user

Especially the use of a virtual character in the application ensures a high level of communication between human and computer due to the fact that a human like avatar brings interactions closer to communication with a human than with a technical interface. Also the use of multi-touch technologies in edutainment applications allows the users (in our case children) a more natural interaction than traditional desktop computer applications [Hourcade09].

The implementation of multi-touch control in edutainment games is nothing new. Just to name a few examples, the project "SynergyNet" of Durham University is engaged in the design and implementation of new forms of user interfaces for educational systems integrating several multi-touch surfaces into a classroom [AlAgha10]. Another approach for an edutainment game was developed in the project "Magic Garden" at the University of Duisburg-Essen, where several young users in the age between six and eight work together at one tabletop with the aim of taking care of a garden and preserving its beauty [Masuch09]. The project “JoyDesk-ExhibitJoy” of the Media University Stuttgart aims at the development of a multi-touch serious game in the context of an interactive exhibition room in a history museum using inter alia several different comic-like virtual 2d characters [Strohmeier10].

Figure 2: Using a multi-touch gesture for interaction with the space traveler

Requirements

Goal of this application demonstrator is an indirect knowledge transfer of information about our solar system and especially about Earth, Moon and Mars. The interface between user and computer is a virtual character that is reacting to user's gestures performed directly on the table's surface. Therefore, the most important technical requirements are real-time 3D rendering and realistic looking animations of the avatar at any time of the application to ensure smooth movements of the character. Furthermore, it is necessary that the application is able to blend between different predefined animations in real-time. As a consequence there are different ideas and technologies that have to be linked into one single application:

- Meaningful usage of touch- and multi-touch gestures linking simple gestures to meaningful system reactions.
- Virtual character and motion captured animations
- Motion Blending to ensure smooth and realistic looking transitions between different recorded animations.

- Real-time 3D rendering with a rendering rate of 25 frame per second and above and with a system's response time less than 120 ms [Miyasatu97].
- Hardware independence to ensure compatibility with various operating systems.

Story, Style and Character Design

The information to be provided to the user is embedded into a simple story of a space traveler called *Newtrix* (see figure 3) that is coming from outer space and visiting our solar system to learn more about it. The traveler decides to study closely the planets Earth and Mars and the moon. The proportion between character and celestial bodies are chosen highly unrealistic to achieve a comic style for the application. *Newtrix* is landing directly on a planet's surface (see figure 4, left) and scanning it. He from now on acts as the interface between the user and the content provided by the application. When the user is pressing an information button for example, the virtual camera automatically zooms at the traveler's visor, in which a short movie clip of the actual chosen planet is played back (see figure 4, right). Another example of this indirect interaction is the way of switching from one planet to another: The user is not choosing a new planet directly, but by selecting a new destination from a planet menu, he/she is inducing the character to lift off the current planet and to fly to the new designated one. Furthermore, some idle animations are also part of the story to keep the user interested such as an UFO visiting *Newtrix* while he is standing on Mars or a satellite passing by when standing on Earth.

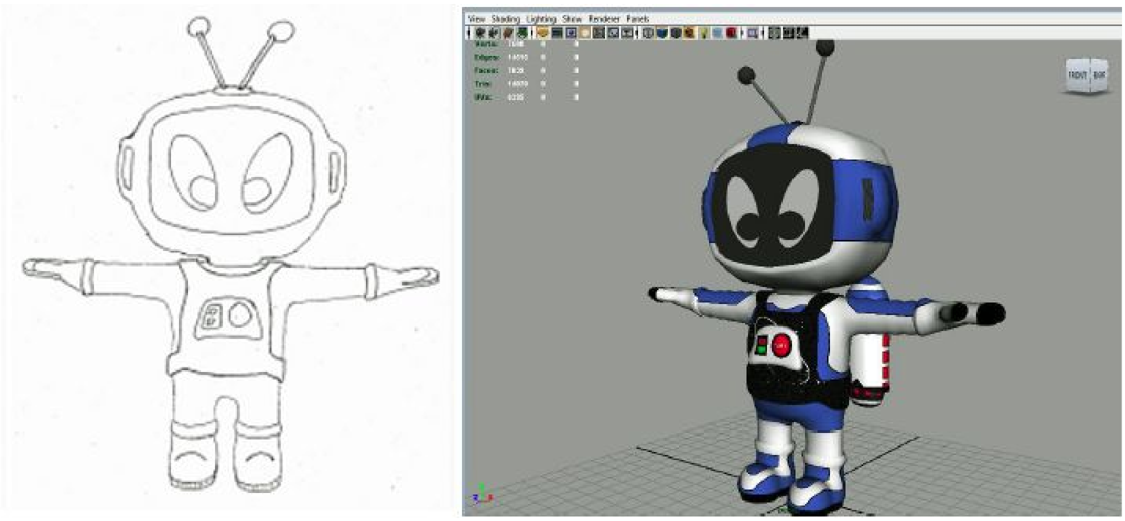

Figure 3: Sketch and modeling screenshot of the space traveler character *Newtrix*.

The virtual world the user is interacting with is designed in a classical comic look, which is highly accepted especially by young users. The protagonist of the demonstrator is a space traveler, which is designed using the typical characteristics of the small child pattern ("Kindchenschema") such as proportionally large head and eyes and short arms and legs with respect to the torso. Both comic style and cuteness are used to attract potential users [Tintel09] to play with the application even if they are technically unversed and do not have used a multi-touch table before.

Interaction Concept

Goal of the demonstrator is to provide touch and multi-touch gestures beyond the classical gestures used with multi-touch tables such as translating, rotating and scaling of 2d elements like digital photographs. With the story of the space traveler in mind, there are several events possible that are implemented into the story. We have to distinguish between direct interactions, which are performed by directly touching the character and indirect interactions, which are performed somewhere else but influence the animation of the avatar.

Figure 4: Information display of the sun and two planets (left) and detailed information about a specific planet displayed within the visor of the astronauts helmet (right)

Direct Interaction:

- Touching the start button on the chest of the avatar activates the travelers jetpack and he lifts of slowly.
- Touching the head or the visor of the character irritates him and he will wipe the occurred finger print away.
- Dragging with two fingers over the character will lead to loosing his balance and he will fall.

Indirect Interaction:

- The character will start to walk, when rotating the planet he is standing on. Depending on the rotational speed the avatar will switch from walking to running and vice versa.
- Pressing the thrust button an the thrust slider displayed left and right of the avatar (see figure 2) changes the altitude of the avatar while it is in flight.
- Rotating the whole 3D scene by a dragging gesture causes the character to turn his head, so that he is looking at the user at any time.
- Without any gestural interaction over a period of time the character will be randomly animated from a set of idle animations such as waving, yawning or looking around.
- Pressing an information button starts a video sequence, which will be displayed within the visor of the avatar (see figure 4, right).
- Changing the location (Earth, Moon or Mars) is realized by a simple menu, which is displayed in the right lower corner of the screen. If a new location is se-

lected, the avatar lifts off and flies to the designated celestial body. After arriving on a new planet, the character first starts to scan the planet.

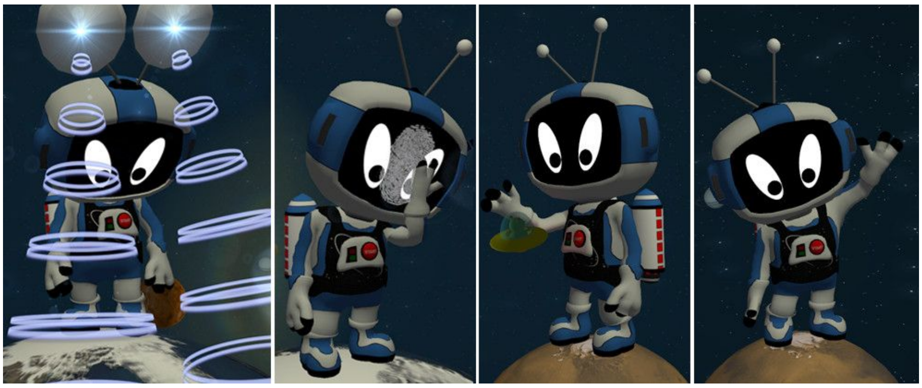

Figure 5: Four animations triggered by interaction: *Newtrix* is scanning a planet, wiping away a fingerprint, touching an UFO and waving to the user

Implementation and Results

Considering the requirements for the demonstrator mentioned above, we have chosen the game engine "Unity" by Unity Technologies in San Francisco as the real-time rendering engine (http:// www.unity3d.com) [Goldstone09]. Unity is an integrated authoring tool for creating interactive 3D video games and real-time 3D animations. Unity fulfills the requirement of hardware independence by supporting the deployment of the application on multiple platforms such as the operating systems Microsoft Windows or Mac OS X, game consoles such as Nintendo Wii or Microsoft's Xbox 360, web via a player plugin and handhelds such as iPhone and iPad.

Furthermore, it supports scripting with a JavaScript-like programming language, which allows to implement missing features of the engine. Both the game engine and its scripting API are well documented with user manuals, tutorials and open example projects. Additional open source implementations such as *uniTUIO* (http://www.xtuio.com) or the Google Code project *Unity3d-TUIO* (https://code.google.com/p/unity3d-tuio) allow an easy interfacing of various multi-touch input devices based on the TUIO protocol. Beside this, the most important advantages of Unity are that first it is capable of real-time motion blending, which allows smooth transition between different animations of the avatar during game play and second that it supports integration with Autodesk Maya (http://www.autodesk.com/maya), which was used for modeling and texturing of the character itself. The character including all accessories such as the jetpack was modeled as a low poly mesh consisting of approximately 7000 polygons. The mesh was bound to a simple skeleton rig to allow the mapping of motion capturing data (see figure 6).

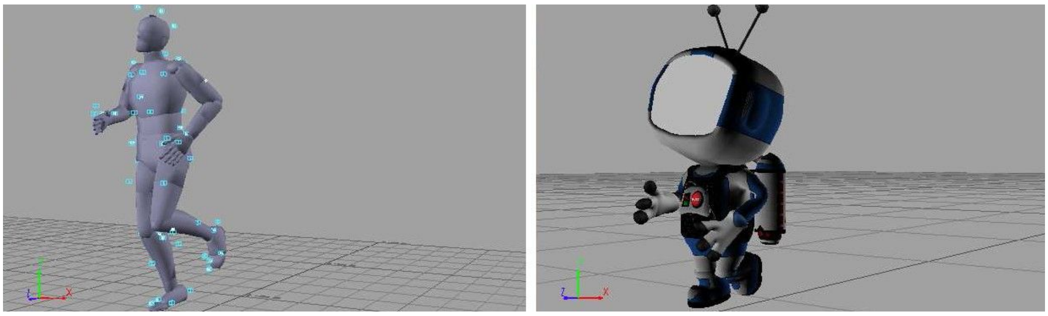

Figure 6: Mapping of motion capturing data onto the space traveler in Autodesk Motion Builder

Overall, the whole 3D virtual world that has to be rendered in real-time by the game engine consists of less than 20.000 polygons including the avatar itself and all other additional objects such as the planets, menu and buttons and explanations labels. Therefore, the demonstrator is rendering in real-time at UXGA resolution with 1600*1200 pixels using a standard PC with off-the-shelf components only. First user tests could proof that the demonstrator is fulfilling all technical requirements for a 3D real time demonstrator with multi-touch support. For future work, it is planned to carry out an evaluation of the concept of using the comic-style avatar in combination with multi-touch technology to enhance the learning effect of the underlying story.

References

AlAgha, I., Hatch, A., Ma, L., & Burd, L. (2010). "Towards a Teacher-centric Approach for Multi-touch Surfaces in Classrooms". ACM International Conference on Interactive Tabletops and Surfaces 2010, Saarbrucken, Germany, ACM.

Goldstone, W., (2009). Unity Game Development Essentials, Packt Publishing, ISBN 978-1-847198-18-1, Birmingham, UK

Hourcade, J.P., Hansen, T.E. (2009). Multitouch Displays to Support Preschool Children's Learning in Mathematics, Reading, Writing, Social Skills and the Arts. Proceedings of IDC - 8th International Conference on Interaction Design and Children, Como, Italy.

Kaltenbrunner, M. et al. (2005). TUIO - A Protocol for Table-Top Tangible User Interfaces", Proceedings of the 6th International Workshop on Gesture in Human-Computer Interaction and Simulation, Vannes, France

Marr A. C. (2010). Serious Games für die Informations- und Wissensvermittlung - Bibliotheken auf neuen Wegen. Wiesbaden: Dinges & Frick GmbH. (B.I.T.online - Innovativ; 28).

Masuch, M., Schild, J. (2009). Entwicklung eines Multi-Touch-Lernspiels für Kinder in XNA. Duisburg: Universität Duisburg-Essen.

Miyasato, T., Nakatsu, R. (1997). Allowable delay between images and tactile information in a haptic interface. Proceedings of the 1997 International Conference on Virtual Systems and MultiMedia, p. 84, IEEE Computer Society

Sostann, K., Tolks, A., Fischer, M., Buron, S. (2010). Serious Games for Health: Learning and healing with video games? GMS German Medical Science.

Strohmaier, T., Isella, C. (2010). Project joydesk-exhibitjoy, Website http://www.joydesk-exhibitjoy.de retrieved June 2011.

Neuss, N., et al. (2009). Über Medien reden. Bundeszentrale für politische Bildung, Bonn.

Tintel, M. (2009). Character Design - Die Methoden und Mechanismen hinter dem Charakter Design. GRIN Verlag, Munich.