

iWorlds: Generating artificial control systems for simulated humans using virtual worlds and intelligent environments

Marc Davies* and Vic Callaghan

Digital Lifestyles Centre, University of Essex, Colchester, Essex, UK, CO4 3SQ

E-mail: {midavi,vic}@essex.ac.uk

Abstract. This paper highlights research into using virtual worlds, intelligent environments and mixed reality to create artificial control systems for simulated humans. Following a brief explanation of this project, the beneficial contribution provided by virtual worlds, intelligent environments and mixed reality systems is outlined. Throughout this project a series of experimental iWorlds, (virtual and mixed reality intelligent environments) have been used to profile the actions and decisions of humans to generate sets of realistic behaviours and test the project research hypothesis. These worlds consist of a mixed-reality household experiment plus two supporting game-like behaviour experiments, namely ‘Bar-World’ and ‘Battleships’ which were designed to explore and reinforce aspects of the underlying theory. As a core part of this work a description of a new profile generation mechanism is presented, which comprises a genetic programming mechanism that blends real user behaviour data to produce varied artificial profiles for non-player computer characters. In support of this a set of experimental results are presented that reveal the new artificial intelligence mechanism does indeed advance the project aim to create realistic simulated humans. A discussion is included that describes the potential benefits that could be gained by incorporating these methodologies into computer games and professional intelligent environment development systems.

Keywords: Artificial Intelligence, Genetic Programming, mimicry, mixed reality, virtual worlds

1. Introduction

1.1. Setting the context

Creating a human-level artificial intelligence is widely considered to be one of the ultimate destinations of modern computer science research, with countless projects around the world working towards this goal. However, in the interim, prior to the creation of this ultimate artificial intelligence, the standard of what is available frequently falls short of creating a widely considered replication of a realistic human-likeness. From a purely software perspective, one key issue promoting this lack of realism is the decision-making abilities of an A.I. program.

This project does not intend to create the ultimate human-level artificial intelligence, concentrating rather on researching a viable interim mechanism.

The aim is to augment an existing and widely-used artificial intelligence platform to allow control systems to be generated for a variety of simulated humans in different scenarios. The new artificial intelligence mechanism developed by this project, (UK Patent Application No. 1012243.0), combines elements of Genetic Programming with data gathered from real people using iWorlds, (virtual and mixed-reality intelligent environments). The intention of the new mechanism is to produce collections of realistic control systems for simulated humans, without extra programming being needed to add variability.

While a human-level artificial intelligence (or even a replica), would undoubtedly have potential applications across countless fields, this project has chosen to focus mainly on computer games and related technologies. Many computer games make use of numerous computer-controlled humanoid characters, each playing a specific role. Larger games commonly use collections of such avatars to create an immersive

* Corresponding author.

atmosphere, (e.g. the population of a town). It is the artificial intelligence system used by these ambient characters that provides the main focus of this project.

1.2. What is human-likeness?

One issue when attempting to create an artificial intelligence capable of producing human-like actions is identifying exactly what a majority of people would consider realistic. For the purposes of this project ‘realism’ of simulated humans refers to the actions they choose to perform, (i.e. decisions) rather than any physical attributes. However, as all individuals possess their own unique set of personality traits, there is no guarantee that a group of people would all respond to the same stimuli in an identical manner. For individual simulated humans this may not be that problematic, but in cases where multiple characters are being used in a group-based scenario, every individual performing the same set of actions could make an artificial intelligence seem much less human-like. Of course, there are also some special cases where it would be considered human-like for everybody to perform the same action simultaneously, for example during a fire evacuation [17].

1.2.1. Personality and personae

In addition to every individual possessing their own unique set of traits, most humans don’t behave in the same manner constantly but tend to alternate their behaviour (i.e. decisions and actions) according to factors such as their current situation and mood at the time. Therefore, in a controller for a mobile non-player characters (NPC) inhabiting a virtual world, its overall ‘Personality’ could potentially be comprised from a number of smaller behaviour-sets, (Personae) created from environment interactions during different situations, (e.g. working in an office, shopping, home-life). Figure 1 shows an example of a potential structure for an NPC controller when observed in this manner. The various categories shown



Fig. 1. An NPC controller comprised from several behaviour-sets.

are not intended to be exhaustive. In a true NPC controller different groups of behaviour-sets would be attached to the program according to scenarios experienced by the person(s) who created the original profile(s) used to build it.

Please note that in this paper the term ‘Personality’ is used loosely, as a means of describing a collection of different behaviour-sets. To this day psychologists and psychoanalysts are unable to collectively agree on what constitutes a personality [24,27].

1.3. Recording human-like personae

Intelligent environments provide an ideal platform to capture the natural decisions and actions of real people. To gather realistic usage data for use within the project mechanism a series of physical, virtual and mixed reality intelligent environments, (iWorlds) have been created. The aim was to immerse participants of evaluations within the test environment as much as possible, in an attempt to keep responses to any stimuli they encountered realistic.

1.3.1. Virtual worlds

The graphical quality of a virtual world can vary significantly depending on the number of dimensions used and the detail of surface and object textures.

Many virtual worlds representing intelligent environments typically employ a top-down viewing perspective with building features and other content being displayed using two dimensional graphics, as shown in Fig. 2 [11,29]. For simple operations this level of detail can be sufficient, although the lack of a height dimension can restrict the number of devices usable within the world. For example, any objects attached to walls or ceilings are displayed at the same level as those on the floor or resting on surfaces. This set-up can potentially create issues such as the con-



Fig. 2. Two dimensional world example.



Fig. 3. Three dimensional world example.

cealment of environment features, due to several overlapping objects distorting the true layout, which in turn could lead to inadvertent misinterpretations of the displayed scene by an unfamiliar observer.

Naturally, one way to resolve the lack of a height component is to create a virtual world modelled using three dimensional graphics, as shown in Fig. 3. With virtual content being modelled in all three dimensions the world could be observed from a first-person perspective by users, rather than using a top-down viewpoint [7]. If necessary the three dimensional world could also be viewed from a third-person perspective, with a camera angle set behind a virtual embodiment of the user, allowing the immersive impression of being ‘inside’ the environment to be preserved. This approach is commonly used in projects such as virtual meeting rooms, where it is necessary for a person to know the location of other world users relative to their own position [25].

1.3.2. *Virtual intelligent environments (iWorlds)*

Virtual worlds are already in common use as tools for researching intelligent environments and other aspects of pervasive computer science. In a sense, these are complementary developments of augmented reality that are already transforming real physical towns and cities with technologies such as broadband and smart homes [6]. In the case of virtual environments rendered with three dimensional graphics, this could provide added benefits to using the system as a research tool. By taking advantage of the immersive properties three dimensional simulations can provide users, a research project using intelligent environments could potentially benefit from more natural reactions and responses to test stimuli during human experiments. Participants using the virtual world could also find it easier to connect with the scene in which they’re placed and possibly even gradually

start to forget they’re taking part in an experiment when interacting with the environment. Of course, creating a virtual environment with immersive properties stands more chance of successfully influencing individuals if the world’s appearance is built to a realistic standard, (e.g. where a chair is easy to identify because it looks like a typical physical chair). Textures used for surfaces and devices should also be considered so nothing appears abnormal or obviously out of place in the virtual environment, with specific objects still being instantly identifiable by individuals. In the case of a Mixed Reality Environment, any simulated devices that also have a presence in the physical component, (i.e. *Dual Reality*), should be of a similar, (or preferably identical) design to their real counterparts.

Virtual worlds created for intelligent environments research projects can be utilised in a mixed reality context. Such systems can provide a convenient method for interfacing with devices in a physical environment and/or exhibiting the influence of agents operating within the test-space. Other benefits also emerge from virtual worlds not being subject to the same restrictions as physical environments. Temporal and climatic features of a virtual world can be customised to allow the conditions within an environment to suit the needs of individual experiments [10]. Features of physical devices can be augmented by virtual counterparts, revealing relationships or allowing actions to be performed that would ordinarily be impossible, (e.g. turning on a light using a switch physically wired to a completely different circuit in the real environment). Mixed reality environments could also allow the creation of virtual devices that have no physical presence, by combining features adopted from a number of real intelligent objects to provide new sensors and/or devices. The new device could be represented in any manner in the virtual component and be given its own UPnP control wrappers, making it indistinguishable from physical devices on the intelligent environment network.

With increasing capabilities of computer hardware and repeated advancements in graphics technologies, it is now possible for virtual worlds of varying complexity levels to be created. The overall graphical and/or technical quality of a virtual world designed to act as a tool for researching intelligent environments is often determined by its intended role in a project. In terms of mixed reality, the quality of a virtual component may be determined by the ratio of its presence in the resulting world, compared to that of the intelligent environment’s physical element [23]. Complexity may also vary depending on the defini-

tion of a virtual world held by the researchers of a particular project. Some projects may limit the virtual component of a mixed reality intelligent environment to a collection of windows displaying statistical and device data on a computer desktop [12]. Other projects such as the University of Texas MavHome [7,8] and the MiRTLE project [4] place a much greater emphasis on their virtual content, creating detailed three dimensional digital reproductions of the physical elements of their respective worlds, including simulated counterparts to real devices.

1.4. Building detailed virtual worlds

While the immersive properties of three dimensional virtual worlds could prove beneficial, the necessity of building the scene to a detailed standard creates several new issues that need to be overcome. Firstly, while researchers may be accomplished in the field of intelligent environments, they may lack the computer graphics programming skills necessary to build a bespoke three dimensional virtual world for their project. Cost could also be another issue, with a virtual world potentially requiring large amounts of resources (i.e. time, hardware, etc.), to construct, especially when modelling extensive or complex environments containing a large number of intelligent devices.

When you consider the complexity and resource requirements that accompany the design and implementation of an immersive three dimensional virtual world it becomes understandable why many research projects decide to use simpler two dimensional alternatives despite their inherent limitations. However, this project has discovered it is possible to overcome the aforementioned issues surrounding the creation of three dimensional virtual worlds.

As part of this ongoing project several three dimensional virtual intelligent environments have been created using minimal resources. The remainder of this section will describe the strategy used to create the different virtual worlds of this ongoing project and briefly highlight previous research, responsible for leading to this point.

1.4.1. Customising computer games

When it comes to three dimensional computer graphics, perhaps some of the finest examples can be found by observing recent projects of the computer games (a.k.a. videogames) industry. For some time now computer game developers have been striving towards the development of photo-realistic graphics for use in their titles. This goal has already led to the



Fig. 4. Sims-based virtual intelligent environment.

common usage of high-definition graphics in titles designed for current generation of games consoles (e.g. PS3, Xbox 360). With the arrival of next-generation hardware designed to display truly three dimensional graphics (e.g. Nintendo's 3DS), combined with the greater expectations of gamers who have used the previous platforms, graphical quality is only likely to continue improving further.

In the modern computer games industry, popular titles (e.g. those that are part of a series), often have multi-million dollar development budgets. Teams of developers spend years designing, coding and testing a game before it is finally released to the public.

During an earlier research phase, this project investigated whether it was possible to take an off-the-shelf computer game, (EA Games' Sims) and recode the software to create a virtual intelligent building, designed to represent a typical household environment [10,11,14].

An example of the intelligent virtual worlds created during this investigation is shown in Fig. 4. The intention was to take advantage of the resources committed by the development team when creating their product, greatly reducing the amount of programming required and allowing the intelligent virtual environment to benefit from the high-quality three dimensional graphics of the original game.

The investigation revealed that it was indeed possible to create a virtual intelligent environment using a modified off-the-shelf computer game. However, several drawbacks to using the strategy were also discovered that would need to be addressed by future research. Major obstacles included issues such as much of the original game code being encrypted for security reasons making editing certain sections of

the architecture impossible. Additionally, the game's creators wrote a portion of their code using a bespoke programming language, (SimAntics), which they developed specifically for that series of titles. As a result many features that would typically be found in a standard programming language were omitted.

1.4.2. Open source solutions

To resolve the problem of the limitations imposed on how an off-the-shelf computer game could potentially be modified, the project's research focus then turned to using open source graphics technologies to create a virtual intelligent environment [10].

Again it was decided to base the virtual world on a typical intelligent household environment. Eventually Project Wonderland, (now known as Open Wonderland) was selected as the new system foundation. Originally designed by Sun Microsystems, Wonderland is a complete client-server game system coded entirely using Java, allowing it to potentially be highly customisable.

Through expanding the Wonderland architecture and adding UPnP functionality limited success was achieved in creating a new virtual intelligent household environment. However, while using this software allowed the editing restriction issues highlighted by the previous investigation to be overcome, access was simultaneously lost to the professionally designed object models included in the complete system of the off-the-shelf game that was utilised by that experiment. A solution to this problem has also since been found.

Open Simulator (a.k.a. OpenSim) [26] like Wonderland is a complete piece of client-server massively multiplayer online (MMO) virtual world software. Originally created from a modified version of the SDK from popular online community Second Life, graphics are now rendered using the open source OGRE graphics engine. While OpenSim allowed the creation of virtual worlds containing a realistic landscape and avatars, the inbuilt tools available were inadequate to allow creation of detailed buildings and/or contents (i.e. furniture, devices, etc.) required in an intelligent environment.

An offshoot project from OpenSim known as RealXtend [28] also benefits from the same detailed graphics, (i.e. realistic avatars and landscaping) of the original Second Life architecture [22], combined with the added features of the OGRE graphics engine software, (e.g. particle scripts). Unlike its OpenSim parent, within RealXtend it is possible to import externally created three dimensional models, (formatted

as an OGRE mesh) into a virtual world and display them as objects. To overcome not being able to use the professionally designed models from a computer game Google's SketchUp graphics editing software was used to import three dimensional models, freely available from the Google 3D Warehouse, into a world created using RealXtend. Google's 3D Warehouse [15] is a vast online library of three dimensional models, many designed by professional graphics artists and students. Most of the models are free to use and can be converted to an OGRE mesh format using some third party Ruby scripts for SketchUp, designed by members of the RealXtend community, (several versions are available to download for free online). As RealXtend is also open source, like Open Wonderland it is possible to modify code from the client or server, (programmed mainly using C# and Python scripting).

1.4.3. The benefits of MMO technologies

Massive Multiuser Online (MMO) virtual worlds have become increasingly popular in recent times. Entire online virtual communities have established themselves around multiplayer architectures, several of which have attracted user populations running into several million. For example, as of January 2010 the previously mentioned online community Second Life possessed over 18 million registered account subscriptions for its services [22]. MMO worlds can also vary greatly in design. Online computer games such as World of Warcraft contain pre-established plot-lines to initially be played through by users. Social worlds such as Second Life tend to have no set plot with most content being generated by individual users usually not affiliated with the world developers.

The user customisation strategy of a community world like Second Life could potentially compliment a virtual world being used as a tool for researching intelligent environments. Such a strategy could allow users to create their own environments (i.e. specify their own layout and contents), potentially even 'owning' their own building(s) within the world. By using mimicry this user generated content could be stored and then recalled at a later date if repeat testing is required or as a quick method of creating a test environment. If a research group decided to use the stored environment in a mixed reality context then the physical component can be customised to suit the layout of the recalled virtual world. Such a system could potentially evolve into an online intelligent community, (similar to Second Life) where various researchers of pervasive computer science from

around the world can access the world and run their respective projects alongside each other in a collaborative environment. An MMO intelligent environment could be accessed by anybody in the world with a suitable computer and capability to use the internet, potentially allowing access to users from many different age-groups and/or cultures each providing their own contributions.

Using MMO architectures to make an intelligent virtual world accessible via internet could potentially lead to numerous benefits for intelligent environments research. Depending upon the overall size of the virtual world, several hundred users could access the environment simultaneously. In the case of this project each user of the virtual world could provide their own unique sets of data as they move around and interact with different features and any intelligent devices it contained. Both Open Wonderland and RealXtend possess MMO functionality, which in turn was inherited by the virtual worlds created by this project that used either of those systems as a base platform. For details of how this project intends to exploit these MMO capabilities when gathering data from users please refer to the following section and 'Next Steps' at the end of this paper.

1.4.4. The current platform

Ultimately this project has discovered that there is always a trade-off regardless which of the aforementioned platforms is used as the base for a virtual intelligent environment system. Creating a bespoke virtual world, even using simple two dimensional graphics, requires a significant programming effort. Off-the-shelf computer games contain better graphics and a three dimensional environment but also possess editing restrictions imposed by their developers. Open-source computer game technologies are more accessible by their nature, but most of the systems evaluated during this project are in a perpetual state of development meaning the stability and usability of certain features is not always guaranteed. As a result of these findings throughout this project several different platforms have been used to create virtual worlds. Platforms were selected on a case-by-case basis comparing its strengths and weaknesses to the requirements of the current investigation.

1.5. Populating virtual intelligent environments

While the focus of this project is primarily on computer games, another area of research is how the new artificial intelligence mechanism can potentially be re-deployed back into the intelligent environments

used to obtain the original user data from real people. In a virtual intelligent environment built using MMO technologies it is possible to have multiple users inhabiting the world simultaneously. This creates potentially benefits for aiding intelligent environments research, by allowing the introduction of computer-controlled (NPC) avatars into the virtual world.

In a virtual world NPC avatars can interact with the environment and its contents in exactly the same manner as a human-controlled character (PC). By extension, this means that in a virtual intelligent environment an NPC avatar could 'use' any installed smart devices. If the virtual world was being used in a mixed reality context it would therefore allow an NPC to control physical devices in a real intelligent environment. Therefore, in the case of an intelligent household NPC avatars could give the impression of being a real person actually living in the environment. This could clearly benefit researchers of intelligent environments as potential test subjects for projects could be stored and recalled from a computer hard-drive as required rather than it being necessary to go through the procedure of bringing in real people. Additionally, as NPCs don't require payment or any other type of luxury, a research project could include more in an evaluation and transport them to be used anywhere in the world if necessary.

1.5.1. Limitations of NPCs

Of course this is all fine in theory but in reality current limitations in artificial intelligence make potentially replacing living test subjects with NPCs difficult at best. If a researcher wanted to use NPCs to test their intelligent environment they potentially would need to spend large amounts of time programming behaviours for each avatar individually then customising it to suit a particular project scenario, (e.g. adding path data and obstacle avoidance protocols to allow the avatar to move around the world). Not only does this take away time from the rest of the project, but the more complex an NPC's behaviours need to be the more computer resources are likely to be required to run the A.I.

Even in the computer games industry artificial intelligence quality has typically been overlooked in favour of better graphics, due to the limited amount of memory available on the cartridge/CD/DVD/Blu-Ray disc for the game and the available resources of the hardware on which it must be playable. This is why when playing computer games it is often easy to identify NPC avatars when compared against human-controlled PCs, purely by observing their actions. If an artificial intelligence system has been pro-

grammed badly, NPCs can exhibit strange behaviours, (e.g. trying to walk through walls or closed doors), or make stupid decisions, (e.g. walking into the line-of-fire of another avatar shooting a gun).

1.5.2. Overcoming NPC limitations

While the extensive programming requirements for creating NPC users may seem to indicate they're not worth the bother, it may be possible to overcome the limitations highlighted above by taking advantage of the intelligent environment itself.

Perhaps the best way to prevent NPC avatars from performing 'stupid' actions is to get them to copy the entity they're being used to personify. Several A.I. projects already in existence make use of mimicry to allow the actions of a real person to be recorded then subsequently used in the controller of a human NPC avatar, which simply repeats the performance. This technique could be used to create NPC avatars for use in virtual intelligent environments by recording the devices a real person uses along with a timestamp of when the action occurred and any specific settings selected (e.g. on/off, levels, etc.).

While mimicry would most likely work, aside from the ability to playback test sessions of real users in an environment there is really no benefit gained by using NPCs in this manner for a researcher of intelligent environments. Human test subjects would still be required in the first instance to generate mimicry controllers for an identical number of NPCs and if the environment layout was altered, (e.g. devices were changed), it could make stored records useless unless extra code is added, (as seen in [17]).

A mechanism developed during this investigation seeks to overcome the limitations of artificial intelligence in human NPC avatars and those of the mimicry technique. By using Genetic Programming the mechanism allows the generation of multiple useful NPC controllers from a single set of mimicry data.

2. Project evaluation iWorlds & simulations

To evaluate the new project mechanism for generating artificial human-like control systems a multi-directional investigation was required. Firstly, before new control systems could be generated it was necessary to 'profile' a number of real people, obtaining a series of behaviour-sets by observing and recording their decision-making and actions. Using mimicry the gathered usage data profiles would then be applied to the new artificial intelligence mechanism.

While a number of two dimensional virtual worlds have also been created and used by this project for evaluating the new artificial intelligence mechanism, when gathering the initial usage data profiles it was essential for the environment to appear as realistic as possible, so participants unfamiliar with the layout would be able to easily interpret specific stimuli and other events, as they occurred in real-time. The three dimensional virtual worlds created by this project have been used to represent a number of different scenarios, including a system designed to function in a mixed reality context alongside a physical intelligent building test-bed.

As the project has chosen to focus primarily on how the new artificial intelligence mechanism can be applied to computer games, it is necessary to evaluate the performance of artificial control systems, (profiles) generated by the system in a variety of different contexts relating to how a computer controlled non-player character (NPC) could potentially be used. As was mentioned during the last section the focus of this project is on ambient NPCs, (i.e. background or supporting characters) rather than those who interact with human-controlled players directly. However, for the purposes of completeness an investigation was also created to assess the performance of artificial profiles generated using the project mechanism when used in a competitive scenario.

2.1. A mixed reality intelligent environment

2.1.1. World design

A mixed reality intelligent environment was constructed to gather usage data from people inhabiting a typical household. For this environment RealXtend was chosen as the base platform for the virtual world component, due to the superior graphics quality it possesses over Open Wonderland. The bespoke buildings and certain devices featured in the iWorld were created using Google SketchUp, while some other objects were built by modifying models from the Google 3D Warehouse.

The virtual world was designed to approximately replicate the layout and features of the physical component of the mixed reality system, specifically the University of Essex iSpace, (an intelligent household environment purpose-built for pervasive computer science research) [2,3,16]. The iSpace (Fig. 5) is a four room environment (i.e. lounge, study, bedroom and bathroom), containing furniture and devices commonly found in a typical household. Being an intelligent building the iSpace is outfitted with



Fig. 5. The University of Essex iSpace environment.



Fig. 6. Mixed reality intelligent environment virtual component.

hollow walls and ceilings, which conceal a myriad of sensors, effectors and other embedded computer-based technologies. All intelligent devices and sensors installed within the iSpace are wrapped into a generic OSGI UPnP framework for computer control.

While the virtual component of the mixed reality intelligent environment, (Fig. 6) was intended to resemble the layout of the iSpace, some simulated devices used were a different design to their physical counterparts and/or had their position in the world slightly altered. Most changes were made either to allow smooth rendering or improve the clarity and appearance of the virtual world. Some changes were necessary due to the availability and quality of suitable Google 3D Warehouse models.

It is worth noting that in a mixed reality intelligent environment it isn't strictly necessary for virtual objects to have an appearance that is anything like their physical counterparts. Virtual models are merely markers in the world that can be potentially either static, or associated with a particular fragment of code built into the system to make them interactive on some level, (similar to a GUI button). A model could be designed to closely mimic the exact appearance and performance of a real counterpart object, but could also just as easily be used to represent a collection of devices that have a completely different physical appearance. For example, a physical lamp's counterpart virtual device could be represented by nothing more than a glowing orb or perhaps a square box with 'lamp' written on the side, but the system would still function in the same way as if a realistic model was used. Likewise, a virtual model of a switch could be used to control the light level in an environment by being used to open/close some motorised curtains and turn on/off a selection of lamps

and ceiling lights. Of course, if virtual models don't attempt to reflect their physical counterparts to some extent it could be difficult for users unfamiliar with the world to identify specific devices, which is why during this investigation it was decided to keep content in both worlds as similar as possible.

2.1.2. Connecting the worlds

After outfitting the physical space and building a similarly styled virtual household, it was necessary to link together any intelligent devices that existed in both worlds, so that they could behave as single entities in the system. Virtual worlds created using RealXtend can have their functionality augmented through the addition of Python scripts. These sub-programs, contained within the RealXtend server, are run automatically when the world is started. Alternatively, each model (prim) contained in the world can have a single script program attached to it, which amongst other features can contain a method that handles events created whenever a user touches (clicks on) the associated object. In addition to all the typical libraries found in the Python language RealXtend scripts can also use a number of converted LSL (Linden Scripting Language) methods retained from the original Second Life program code. The RealXtend community and development team have previously created several subprojects using Python scripts, (most of which are included in the program download files), including methods for allowing X-10 modules to be controlled from within the world. The University of Essex iSpace does contain X-10 modules, but like every other intelligent device in the environment they are wrapped into and controlled by our generic OSGI UPnP framework, therefore a new bespoke device control system needed to be created.

To create the mixed reality intelligent environment a new piece of middleware was developed, which would act as a bridge between the RealXtend virtual world and the OSGI UPnP wrappers linking the physical devices on the iSpace network to the system. For this a Socket based client-server framework, (programmed in Java) was used. The bridge program involved virtual Python and physical Java device clients sending request messages to the server, which would then return the required state values. It was designed to ensure that device that existed in both worlds had the same settings simultaneously.

As the physical devices in the iSpace was controlled using the inbuilt OSGI UPnP wrappers, rather than a direct interface, the bridge could easily handle several different types of sensor or intelligent device,

controlled by various third-party operating systems and software, (e.g. X-10s, Echelon LonWorks, Phidgets, etc.). Multidirectional communication was also possible using the bridge, allowing physical devices to control their virtual counterparts in addition to vice-versa.

2.1.3. A mixed reality functionality example

Figure 7 provides views from a case-study that took place within the operational mixed reality intelligent environment, seen from the perspective of a user standing in the iSpace lounge area. The larger virtual world image in Fig. 7 is an extrapolation of the virtual component scene being displayed on the real touch-screen television shown in the smaller pictures. The example is of a scenario where four dimmable spotlights in the iSpace ceiling are being controlled by interaction through the virtual world. By clicking on the individual light models the brightness level of the associated physical counterpart device could be adjusted. Alternatively, as is shown in the Fig. 7 example, the lights could be controlled collectively by clicking on the virtual light switch model, which would change the state of each device simultaneously. Even when manipulated collectively



Fig. 7. A mixed reality intelligent environment example.

the system would preserve the individual brightness settings set for each light. In the virtual world, the simulated lights have their current brightness level and state (i.e. on/off) displayed above each device model in red text. This information is updated automatically whenever a state change occurs.

The virtual light switch model shown in Fig. 7 also has a physical counterpart in the real iSpace. While the virtual model is used to change the state of the dimmable spotlights and a number of X-10 enabled lamps located elsewhere in the lounge area, the physical light switch is hard-wired on a circuit only containing itself and a pair of paper globe binary ceiling lights, (both of these are visible in the virtual screenshot and one is partially visible at the top right corner of both real-world views in Fig. 7). The physical light switch is not directly wired to the real dimmable spotlights or any X-10 enabled lamp so can exhibit no influence over these devices. Only within in the realm of the mixed reality world can the physical switch can have its functionality augmented to make controlling both sets of lights possible.

Virtual or soft appliances [5], created by combining groups of embedded computer services from the physical iSpace, can also be given a presence in the environment using the virtual component. While such appliances don't usually possess a monolithic manifestation within a physical intelligent space, they could be assigned a model in the virtual world and a set of OSGI UPnP wrappers written for the new device could be added to the physical iSpace system. The virtual appliance could then be accessed and controlled in the same way as any other device on the mixed reality intelligent environment network, albeit only be executable from the virtual side.

Third party programs (e.g. agents) can also be attached to the bridge software, allowing researchers to easily customise the world functionality at a code level for the bespoke requirements of their projects. For example, in the investigations of this project the original middleware bridge software is modified by adding a logging system to the server code, to allow user device usage data to be collected.

2.1.4. MMO functionality

Having been modified from *Second Life*, the *RealXtend* software has inherited attributes typically found in MMO virtual worlds. By taking advantage of the *RealXtend*'s MMO features each inhabitant of the physical iSpace environment can be allocated their own independent avatar character to allow them to fully interact with the mixed reality world. Several users could simultaneously 'inhabit' the mixed real-

ity iSpace, either in person within the physical environment or remotely via the virtual world. As MMO worlds are designed to be used over networks or the internet, remote users could potentially be located anywhere in the world, running the virtual world software from their personal computer and using it as a gateway to access and interact with the devices of the physical iSpace. Remote users could also create their own devices from their current location and upload them to the virtual world to be included in the iSpace design, (although only within the virtual component) and interacted with by other mixed reality intelligent environment inhabitants.

2.2. Bar iWorld

2.2.1. World design

In addition to the mixed reality intelligent household environment, a number of other virtual worlds were created to allow the gathering of user data from a variety of different scenarios. The next iWorld create for this project was designed to model a simple bar environment. Unlike the household iWorld which was designed with a high level of realism in an attempt to coax users into providing natural responses to stimuli, in this virtual world a more basic approach was adopted. This change was intentional to allow usage profiles, which can be used as mimicry data by the project mechanism, to be gathered by a variety of different methods.

The entire virtual world was programmed from scratch using the standard Java SDK. The number of different intelligent devices featured in this iWorld was significantly less than the previous household environment, allowing the quality of the simulation to be reduced while still avoiding misinterpretation by participants. As the system was entirely bespoke, unlike in the mixed reality world the logging system to gather the usage data could be implemented as part of the main program rather than as an addition grafted to the virtual world. Because the entire world was being created from nothing the implementation strategy required the design and functionality to be kept as simple as possible, to minimise the programming effort required, but still obtain useful sets of usage data from participants. With this in mind it was decided to use two dimensional graphics for the virtual world, as the only overlapping objects were drinks placed on a table, which could easily be explained at the start of an evaluation. Figure 8 shows a screenshot of the final design for the bar iWorld, with a number of served drinks on one table.

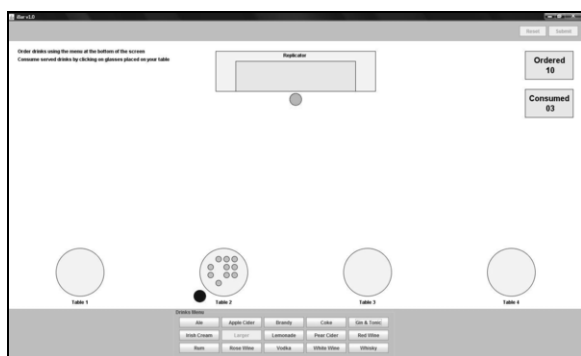


Fig. 8. The bar iWorld.

2.2.2. Bar iWorld functionality

The entire iWorld for the bar environment only consisted of four tables and a ‘bar’ at the back of an otherwise empty room. As the simulation was built using two dimensional graphics the avatars were represented by different coloured circles on the screen, (blue for a customer, green for a waiter). Drinks are collected from the bar area by the waiter NPC and served to the customer’s table. Evaluation participants played the role of the customer. They could order drinks via an interactive menu or consume served drinks by clicking on individual units. The interactive drinks menu at the bottom of the screen contained fifteen different options displayed on an interactive button. If a specific drink became ‘out of stock’ during the evaluation the associated button in the menu was deactivated, preventing any further orders from a user.

A tally of the number of served drinks and consumed drinks was included on the right of the screen. At the bottom of the screen a menu containing fifteen different drinks was displayed. If a specific drink became ‘out of stock’ during the evaluation the associated button in the menu was deactivated preventing any further orders from the user.

The bar in the world was an intelligent device similar to a vending machine, which monitored stock levels and processed orders from the waiter, dispensing a requested drink if it was in stock. The drinks contained within the bar were also intelligent, capable of telling the world when their contents were consumed by a customer.

2.3. Battleships game

As was mentioned earlier, this project is mainly focusing on the artificial intelligence of NPCs used as ambient, (or background) characters in computer

games. However, for completeness an investigation into how the artificial profiles created by the project mechanism perform in a competitive scenario was also carried out.

For this investigation a fully implemented game inspired by the classic ‘Battleships’ (a.k.a. Battleship), board game was created. The system was completely bespoke, created using the standard Java SDK and included a logging system to gather user data such as the positions selected for different ships and the targets selected on an opponent’s grid. A screenshot of the system in the middle of a tournament is shown in Fig. 10.

3. The Profile Mechanism

The new artificial intelligence mechanism created by this project, (hence forth referred to as the Profile Mechanism) is a multi-tiered strategy for the generation of artificial behaviour-sets (profiles) for NPC avatars in a virtual world. Artificial profiles created by the mechanism are intended to provide sets of realistic (human-like) behaviours that when used by simulated human NPCs, allow them to blend in with PC avatars controlled by real people. The aim is for the artificial profiles to provide a sufficient level of realistic behaviour control to prevent NPCs from being instantly identifiable from PCs in the same scenario when their actions are compared collectively.

Using the mimicry to record real user interactions with iWorlds allows the creation of one NPC controller behaviour-set per person observed. The Profile Mechanism takes one or more standard mimicry profiles then applies genetic programming to evolve multiple artificial behaviour-sets, usable by an NPC controller. As the usage data contained within the mimicry profiles was originally collected from real people, it was believed that the human-likeness of the recorded actions would remain relatively preserved within the artificial controller behaviour sets.

3.1. Proof-of-concept testing

Initially this project sought to determine whether it was possible to evolve suitably realistic behaviour sets using the profile mechanism, maintain at least some element of the human-like characteristics of the original user data that was being mimicked. When attempting to answer this question, the mixed reality intelligent household system was used to generate sets of device usage profiles, by having participants

guide an avatar ‘inhabitant’ around the virtual component of the iWorld.

3.1.1. Household experiment outline

A single experiment would involve observing the actions of the human-controlled avatar within the iWorld over a timescale of one day. Note that the speed of time can be changed at will in a virtual world so all references in this section should be taken as referring a period in the simulated environment, unless otherwise stated.

During this experiment the genetic program of the profile mechanism was inspired from the original designs of Koza [18–21]. The system was programmed to evolve the original mimicry profiles, (gathered device usage datasets) over one hundred generations, each with a population limit of two hundred profiles. Mimicry data selected for its fitness by the program could have one of the traditional genetic operators, (i.e. reproduction, crossover or mutation) applied to it to create one or more artificial output profiles for the population of a subsequent generation.

To assess fitness and rank individual profiles for each population generated by the genetic program, the virtual intelligent environment was augmented with new attributes, inspired by EA Games’ Sims computer games. Each avatar in the virtual intelligent environment was assigned a set of four ‘needs variables’ (i.e. boredom, hunger, hygiene and tiredness). These variables would increase gradually over time (creating a need) or as a consequence of performing certain actions within the world, (e.g. working increases the tiredness level). The value of each needs variable could be replenished by directing the avatar to use an appropriate intelligent device, (e.g. if the avatar was tired then sleeping in a bed would accommodate that need). The value of each needs variable for an individual avatar was represented as a percentage figure. The values of each variable were recorded after each hour has passed in the virtual world. The fitness of an individual profile was measured by the level of ‘happiness’ its actions created when performed by an avatar during the simulated day. The happiness level was calculated by adding together the average values for each of the four needs variables during the simulated day and subtracting that total from one hundred, (the maximum possible value being one hundred percent).

In addition to providing a means of evaluating the fitness of profiles the needs variables also served as stimuli to encourage the users controlling the avatar to perform a varied group of actions over the course

<p>Input Profile A</p> <pre> monday.21.47=loungeTelevision DefaultTelevision setState 1 monday.21.48=loungeTelevision DefaultTelevision setChannel 1 monday.22.3=loungeTelevision DefaultTelevision setChannel 3 monday.22.47=loungeTelevision DefaultTelevision setState 0 monday.22.49=stereo DefaultStereo setState 1 monday.22.50=stereo DefaultStereo setStation 3 monday.23.3=loungeTelevision DefaultTelevision setState 1 monday.23.4=loungeTelevision DefaultTelevision setChannel 1 tuesday.2.47=loungeTelevision DefaultTelevision setState 0 tuesday.2.49=toilet DefaultToilet tuesday.2.52=bathroomSink DefaultSinkBathroom tuesday.2.54=bed DefaultBedDouble </pre>
<p>Input Profile B</p> <pre> monday.21.10=loungeChair2 DefaultChair monday.22.31=studyChair DefaultChair monday.22.33=studyComputer DefaultComputer setState 1 monday.22.34=studyComputer DefaultComputer Work monday.23.8=studyComputer DefaultComputer setState 0 monday.23.9=toilet DefaultToilet monday.23.23=bathroomSink DefaultSinkBathroom monday.23.25=bath DefaultBath monday.23.43=bedroomChair DefaultChair monday.23.45=bedroomWardrobe DefaultWardrobe monday.23.47=loungeTelevision DefaultTelevision setState 1 </pre>
<p style="text-align: center;">Crossover Point = Monday 22.54</p>
<p>Output Profile A</p> <pre> monday.21.47=loungeTelevision DefaultTelevision setState 1 monday.21.48=loungeTelevision DefaultTelevision setChannel 1 monday.22.3=loungeTelevision DefaultTelevision setChannel 3 monday.22.47=loungeTelevision DefaultTelevision setState 0 monday.22.49=stereo DefaultStereo setState 1 monday.22.50=stereo DefaultStereo setStation 3 monday.23.8=studyComputer DefaultComputer setState 0 monday.23.9=toilet DefaultToilet monday.23.23=bathroomSink DefaultSinkBathroom monday.23.25=bath DefaultBath monday.23.43=bedroomChair DefaultChair monday.23.45=bedroomWardrobe DefaultWardrobe monday.23.47=loungeTelevision DefaultTelevision setState 1 </pre>
<p>Output Profile B</p> <pre> monday.21.10=loungeChair2 DefaultChair monday.22.31=studyChair DefaultChair monday.22.33=studyComputer DefaultComputer setState 1 monday.22.34=studyComputer DefaultComputer Work monday.23.3=loungeTelevision DefaultTelevision setState 1 monday.23.4=loungeTelevision DefaultTelevision setChannel 1 tuesday.2.47=loungeTelevision DefaultTelevision setState 0 tuesday.2.49=toilet DefaultToilet tuesday.2.52=bathroomSink DefaultSinkBathroom tuesday.2.54=bed DefaultBedDouble </pre>

Fig. 9. Profile crossover example.

of the simulated day. Using this strategy ten mimicry profiles were collected using the mixed reality intelligent environment, and then used as the initial population of the genetic program. Both the mimicry profiles and subsequent artificial versions each consisted of a set of action rules. Each rule was recorded along with a timestamp stating when the action was performed. If a profile was selected by the genetic program it was these action records that were subject to modification by the operators, to evolve new artificial offspring. An example of the layout of a profile can be seen from the fragments shown in Fig. 9.

3.1.2. Household experiment results

This initial experiment revealed that the creation of artificial behaviour-sets using a combination of mimicry and genetic programming was possible. As can be seen from the example shown, during early generations the genetic program was able to use crossover to effectively recombine the behaviour-sets

of two mimicry profiles (Fig. 9). The structure of the original mimicry behaviour-sets was preserved on either side of the crossover point, allowing realistic actions to still be performed by many of the artificial profiles generated by the system.

The mutation functioned on various levels in the genetic program, capable of modifying the timestamp associated with a single selected rule, the action itself or changing several randomly selected actions and/or timestamps in the entire behaviour-set. The type of mutation used was also randomly chosen on occasion.

Despite showing that generating NPC controllers using this method was possible, the investigation also revealed that while using the current genetic programming design there was no guarantee of evolved profiles maintaining the a structure similar to that of a mimicry profile beyond approximately fifteen generations. As the genetic program began using artificial profiles from a previous generation as the parents for a new populations, the structure of the original mimicry profiles quickly became distorted as records were mutated and recombined at various points, (as seen by the orphaned computer action of Output Profile A in Fig. 9). By approximately the fifteenth generation profiles had dramatically grown in size, often containing long chains of actions all occurring within a minute of each other. Stupid actions also began to emerge, with avatars performing tasks such as only going to bed for a few minutes during the entire day. Clearly the actions included in these later artificial profiles could no longer be considered to possess a realistic human-likeness.

This experiment was designed purely to evaluate whether it was possible to generate realistic control systems for human NPCs, using a combination of mimicry and genetic programming. Therefore, it was not entirely unexpected when it emerged that behaviour-sets evolved in later generations gradually lost their ability to perform human-like actions in a realistic way. The genetic program used was a basic design, (especially in terms of the data being used for individual nodes). The population size for each generation was also low, which was likely to have caused the profile distortion to occur more quickly, (due to the small number of profiles available for potential modification), although similar results would have likely eventually emerged even if the limit been set higher.

Having shown that it was possible to generate realistic controllers for human NPCs with the project's profile mechanism, the next step was to refine the system architecture and investigate potential methods of resolving the issues discovered. For the next inves-

tigation, it was also decided to move away from the complex mixed reality intelligent household environment to a simpler program, (with fewer options). This would allow updated profile mechanism designs to be evaluated under simpler conditions, potentially making it easier to identify any recurring problems.

3.2. Competitive performance evaluation

As this project is focussing on applying the new artificial intelligence mechanism to computer games as well as intelligent environments, it was decided that the next evaluation should use a game platform. The Battleships game simulator (Fig. 10), mentioned in the previous section was created specifically for this investigation. The project also decided to use this opportunity to evaluate the performance of artificial profiles generated by the new mechanism when applied to a competitive scenario. In computer games simulated humans are also used as opponents to player, so for completeness it was important to assess this potential usage method. Likewise, in an intelligent environment there are occasions where multiple individuals can inhabit the same smart space simultaneously. In such situations there will almost inevitably be instances where the preferences of one individual conflicts with those of another. While the experiment used by this investigation is aimed more towards the computer games aspect of this project, it is possible that a similar tournament solution could be applied to intelligent environments to decide which individual has precedence in conflict situations.

3.2.1. Profile Mechanism revisions

The largest change in the revised profile mechanism design was allowing the genetic program to generate its own random initial population of behaviour-sets for profiles, rather than automatically using

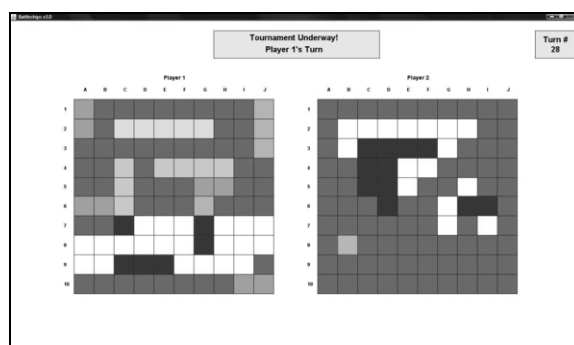


Fig. 10. The Battleships game world.

those collected using mimicry. The fitness of the evolved profiles for each generation was measured by comparing the performance produced by a new profile, to that of a pre-selected benchmark, taken from the obtained usage datasets, gathered from users who previously played the game. Using this strategy, for each experiment only a single mimicry profile, per NPC, was needed to provide a benchmark.

Another change made to the genetic programming portion of the profile mechanism was to the selection method used when choosing profiles to become parents of new samples for a subsequent population. In the new system a fitness-proportional approach was selected based on the normalised fitness value of individual profiles. It was also possible to visualise the programs evolved by the system using the more traditional tree-structure for this investigation as function and terminal sets were simpler.

3.2.2. Battleships experiment outline

For this investigation a limit was placed on the size of an acceptable evolved profile. The behaviour trees produced evolved by the genetic program were limited to a depth of three levels. This threshold could be exceeded by overrunning branches being added as a result of crossover or mutation. However tree-nodes selected as the application points for genetic operators were required to be located within the specified limit. The population limit for a generation was increased to one thousand and the maximum number of generations also increased to one thousand. The genetic program was given a thirty percent chance of using the mutation operator and twenty percent chance of reproducing selected evolved profiles. The crossover operator would be used for creating any other new artificial profiles.

The function set used by the genetic program to generate or evolve profiles was as follows:

- IF_PLAYER_SCORES_HIT
- IF_PLAYER_SCORES_MISS
- IF_THREE_MISSES_IN_A_ROW
- IF_FIVE_MISSES_IN_A_ROW

The terminal set used by the genetic program when generating or evolving profiles was as follows:

- MOVE_NORTH
- MOVE_SOUTH
- MOVE_EAST
- MOVE_WEST
- MOVE_NORTH_EAST

- MOVE_NORTH_WEST
- MOVE_SOUTH_EAST
- MOVE_SOUTH_WEST
- SELECT_A_RANDOM_SQUARE

The first two functions are dependent on results produced by a player targeting a particular enemy grid location. The additional miss functions were added to encourage variety for the structure of the produced behaviour-sets. As is probably already clear, the terminal nodes each relate to the different potential options a player could choose from when selecting their next target. The move options automatically select the next untargeted square in the relevant direction on the grid. The remaining option was to randomly select an untargeted location.

The investigation itself firstly involved using real people to play the game in competition against various NPC enemies. As shown in Fig. 10 the world used a pair of square grids with a height and width of ten spaces, each providing one hundred different locations potential targets for a player during a game.

A new artificial profile was generated by the mechanism for the NPC controller at the start of each game. The intention was to evaluate whether artificial behaviour-sets evolved by the profile mechanism were significantly different, in terms of performance output (i.e. the number of moves required to win a game), when compared to other actions generated using other strategies and the live performance of a real person reacting to the changing game scenario.

The other strategies used in this investigation generated behaviour sets by pure mimicry (i.e. the same recorded usage data used by the profile mechanism), and a non-augmented genetic program following Koza's original design. For mimicry profiles the opponent's ships would be in different positions on the grid to when the behaviour set for the profile was originally created. As a result it was highly likely that the NPC controller would reach the end of the mimicry profile behaviour set before succeeding in sinking the entire enemy fleet. Whenever this issue occurred, for the remainder of the game targets were chosen by a random selection system. The non-augmented genetic program was designed to evolve entirely artificial profiles customised to suit the current grid it was presented with. Obviously to do this in the game scenario, the genetic program needed to 'cheat' by seeing the layout of the opponent's grid prior to the start of the tournament. The genetic program used the same function and terminal sets and had the same parameter settings, (i.e. population size,

Table 1
Results data summary for the Battleships investigation

Test	Number of moves performed – (Max. 100)			
	Human player	Mimicry profile	Genetic program	Profile mechanism
1	78	85	87	84
2	68	100	89	97
3	74	99	80	98
4	85	100	85	85
5	85	98	97	96
6	80	99	90	96
7	92	98	97	99
8	77	91	91	96
9	91	89	69	92
10	89	93	95	90
Average:	81.9	95.2	88	93.3
5% Thresholds			High = 75.93 Low = 24.07	
U value	12.5	66.5	30	

generation limit, etc.), as the profile mechanism. The fitness measure used by this genetic program system was the number of battleships sunk by a controller using each evolved behaviour-set after targeting eighty possible locations. Each of the strategies were compared against real people, as shown in Table 1.

3.2.3. Battleships experiment results

The investigation began with ten tournament rounds where human players competed against a randomly selected artificial intelligence to create a cache of stored usage data for mimicry profiles. Of course as at the beginning of the investigation there were few mimicry profiles to be used the human player was almost guaranteed to win the game. This was an intentional step taken to ensure complete behaviour sets were recorded from the real participants, (i.e. every action leading to an eventual victory).

A further ten rounds of the tournament were then performed for each of the three artificial intelligence strategies. For each run both players were controlled using different profiles generated from the same artificial intelligence strategy. The number of moves required for the winning player to sink their opponent's fleet of ships was recorded.

In addition to the results (i.e. number of required moves) recorded from the victorious player for each tournament, Table 1 also shows values of a statistical analysis. Using a Mann-Whitney U-Test analysis the results gathered from real people, pure mimicry and non-augmented genetic programming were compared with those of the profile mechanism. The outcome of this analysis is discussed in further detail in the evaluation. However as was to be expected the raw

results from the experiment show that in terms of performance, the behaviour-sets generated by the profile mechanism on average are at a level between those created using pure mimicry or genetic programming and perform much worse than the reactive responses of a real human player.

3.3. Evolving human-likeness

In both the previous investigations the project's Profile Mechanism successfully created new sets of artificial behaviours based on mimicry data collected from real people. However, neither investigation has yet conclusively proven that any of the artificial profiles generated by the project mechanism would allow an avatar to exhibit a human-likeness through its actions. To assess whether the evolved artificial profiles do contain behaviour sets that maintain the human-like characteristics of the original mimicry profiles a new more qualitative analysis was required.

3.3.1. A Turing-inspired evaluation strategy

The Turing Test, created by renowned computer scientist Alan Turing was designed as a response to the question 'Can machines think?' The issue that prompted the creation of the test was Turing's problem with the word 'think,' as there is no clear answer to the question 'What exactly is thought?' The Turing Test consists of human participants typing a series of questions into a computer. Based on the responses the participant receives for their questions they can decide whether they are being answered by another human or a machine with some sort of artificial intelligence [1,9].

In the same way as Turing had difficulty clearly defining the word thought, in the case of this project the term human-likeness is equally ambiguous. As the Turing Test was designed to overcome a similar vague definition, it can potentially be used as inspiration for a strategy suitable for use by this project.

A new evaluation method was created for this project inspired by the Turing Test strategy for artificial intelligence. The strategy could be adapted to suit the needs of this project by creating a ‘spot-the-difference’ type game for evaluation participants to play. A number of different instances of simulated human would be generated by the system and run simultaneously in a demonstration iWorld, (or each avatar could have their own instance of the iWorld if more appropriate). Each simulated human instance would be assigned a unique artificial intelligence control system generated using the project profile mechanism, random selection, or from one of the other previously used strategies, (i.e. pure mimicry or non-augmented genetic programming). While the number of avatars in each experiment that use a control system derived from a particular strategy would be pre-determined, which of the instances would be assigned each strategy would be through random selection. This would mean even the person giving the test would be unaware which avatar was using each strategy. Additionally, in cases where more than four avatars are used there would logically be more than one avatar using some strategies.

The task for the evaluation participant is to simply observe the performance of each of the avatars, (i.e. watch their actions as they interact with the iWorld). After a specified time interval the participant is then requested to identify which of the simulated humans is using an artificial control system, (i.e. not an unaltered mimicry profile, which is the equivalent of a real person). However, at no point are participants told the ratio of ‘real’ to artificial control systems used in the experiment. If the avatars using artificial profiles generated using the project artificial intelligence mechanism are not distinguished by every participant of the evaluation those profiles could be considered to possess at least some level of realistic human-likeness, in terms of their actions and decisions.

3.4. Bar experiment

Having completed an analysis of the performance of the new artificial intelligence mechanism when applied to a competitive scenario, the project returned to the original source of intelligent environments for gathering user data for mimicry profiles.

As it was still believed a simpler virtual world with fewer options and stimuli than the mixed reality intelligent household environment was needed, a new virtual intelligent environment was created, designed to model a typical bar.

3.4.1. Bar experiment outline

The experiment used for this investigation was multi-staged so as to accommodate the new qualitative evaluation strategy. To allow participants to first become accustomed to the bar iWorld the evaluation started by placing them in the minimalistic environment with a single task to be completed, (i.e. ordering drinks to be delivered by a waiter NPC). A stock limit of three units was imposed on each drink type available to order. The table to which the drinks were served had a maximum capacity of sixteen units. Additionally, to keep the profile behaviour sets simpler a restriction of only one drink per order was imposed, (i.e. players could order a drink but must wait until it was delivered before requesting another).

The genetic program used by the profile mechanism was again modified, now with no limit imposed on the number of generations it could produce before producing a result. The population size was kept at one thousand, with a thirty percent chance of mutation and twenty percent chance of reproduction.

The function set used by the genetic program to generate or evolve profiles was as follows:

- IF_N_DRINKS_CONSUMED
- IF_N_DRINKS_ORDERED

Thirty function nodes were actually used with each of the above having an instance with N being replaced by an integer between one and fifteen, (i.e. IF_10_DRINKS_CONSUMED). To save space the complete list has been omitted from this paper.

The terminal set used by the genetic program when generating or evolving profiles was as follows:

- ALE
- APPLE_CIDER
- BRANDY
- COKE
- GIN_AND_TONIC
- IRISH_CREAM
- LARGER
- LEMONADE
- PEAR_CIDER
- RED_WINE
- ROSE_WINE

- RUM
- VODKA
- WHITE_WINE
- WHISKY

The terminal set simply contains a node to order a single unit of each of the available drink types. If a program attempts to order a drink that is currently ‘out-of-stock’ a null value is returned, otherwise a new instance of the object is returned.

As with the Battleships investigation, the Bar iWorld also generated some artificial behaviour sets using non-augmented genetic programming, to allow comparison place with the profile mechanism. An additional node ‘RANDOM_DRINK’ was added to the terminal set for this genetic program to give evolved trees an option to prevent repetitive actions.

For both the profile mechanism and non-augmented genetic program, the generated behaviour sets had their standardised fitness evaluated by adding the totals of the number of drinks consumed subtracted from the total ordered and the number of different drink types subtracted from the total ordered.

3.4.2. Profile Mechanism strategies

During the previous investigations the mimicry profiles created from real usage data have been applied to the genetic programming in different ways by the profile mechanism. In the mixed reality intelligent household investigation mimicry profiles were used as the initial population of the genetic program. For the Battleships simulation a single mimicry was selected at the start of the program then used as a benchmark to evolve a new population from scratch. To assess which of these strategies produced the best artificial profiles, during the Bar investigation both were used.

3.4.3. Bar experiment results

As with the other investigations ten experiments were performed using evaluation participants. Each experiment consisted of a single ‘game’ run where users ordered drinks to create a mimicry profile. Figure 11 shows the ‘game’ mode Bar iWorld design.

Following the game mode, four ‘evaluation’ runs were performed using the new strategy inspired by the Turing Test. Each run used a combination of pure mimicry and one other mechanism to create control systems for the NPC customers. The systems used were non-augmented genetic programming, random selection and the profile mechanism. The profile mechanism was used for two run so that both methods of applying mimicry data could be evaluated.

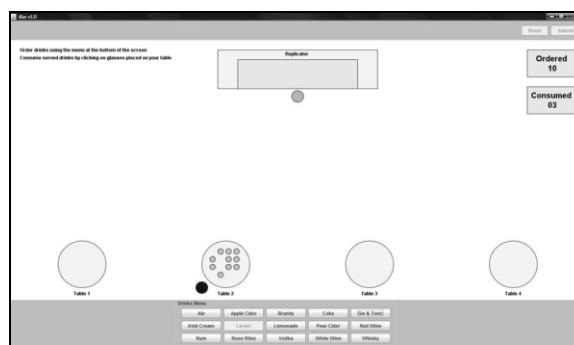


Fig. 11. The bar iWorld in game mode.

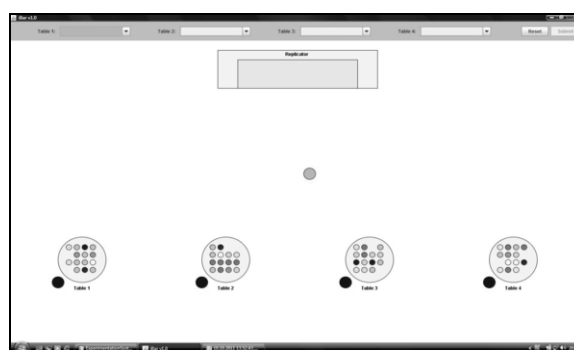


Fig. 12. The bar iWorld in evaluation mode.

The four tables in the iWorld were occupied by an NPC customer, with the waiter serving requested drinks to each in turn. Two of the NPC customers used a control system generated by the selected artificial intelligence mechanism for that run. Likewise, the other two NPC customers used a pure mimicry profile, (from the stored user data) to replicate a real person. Participants were required to watch the performance of the different avatars simultaneously then select which of the four customers they believed to be artificial at the end of the run. Participants were told the ratio of real to artificial control systems used by the NPC customers during the evaluation. Figure 12 shows the slightly modified design of the Bar iWorld for the evaluation mode.

Table 2 shows the results from the Bar iWorld investigation, with a breakdown of how many of the different artificial control systems used were identified by participants. For the profile mechanism the table shows the different results gathered from using both of the mimicry data application strategies outlined earlier.

After experiments involving five participants it was noted that for many of the runs at least one artificial profile was being identified, regardless of the

Table 2
Results summary for the Bar iWorld investigation

Test	Control system identification breakdown – (total artificial controllers used / total found)				
	User 1	User 2	User 3	User 4	User 5
1	2 / 1	2 / 2	2 / 2	2 / 2	2 / 2
2	2 / 1	2 / 2	2 / 1	2 / 1	2 / 1
3	2 / 1	2 / 1	2 / 1	2 / 1	2 / 2
4	2 / 1	2 / 0	2 / 1	2 / 2	2 / 0
Test	User 6	User 7	User 8	User 9	User 10
	1	1 / 1	1 / 0	1 / 1	1 / 1
2	1 / 1	1 / 0	1 / 0	1 / 1	1 / 0
3	1 / 0	1 / 0	1 / 0	1 / 0	1 / 0
4	1 / 0	1 / 0	1 / 0	1 / 1	1 / 0

Test	Key
	Control system generation mechanism
1	Non-augmented genetic programming
2	Random drink selection
3	Profile mechanism (direct mimicry profile application)
4	Profile mechanism (mimicry profile benchmark application)

artificial intelligence strategy used to generate the control systems. Several participants admitted to guessing when providing their results, which suggested that some correct identifications could be the result of luck when choosing NPCs as there was a fifty percent chance of guessing a correct NPC. Therefore for the final five participants it was decided to switch to a ratio of one artificial profile and three pure mimicry (real) behaviour sets for the customer control systems, reducing the chance of a lucky guess to a mere twenty-five percent. As can be seen from the results in Table 2, there were far more runs in the second half of the investigation that resulted in the artificial control system not being correctly identified.

Overall the results from the Bar iWorld investigation suggest that in the case of the profile mechanism, evolving new artificial behaviour sets using a single mimicry profile as a benchmark may produced better results with regard to the level of human-likeness. While in the second half of the evaluation the direct application method did marginally outperform the benchmark strategy, in the earlier segment there were two separate instances where neither of the artificial profiles were identified. Nevertheless the results gathered reveal that both methods of applying mimicry profiles to the project mechanism's genetic program can produce more realistic, (in terms of human-likeness) sets of behaviours than non-augmented genetic programming and random selection, when compared directly against actions from real people.

4. Evaluation

The results obtained from the three investigations detailed in this paper seem to support this project's attempts to devise a new artificial intelligence mechanism to produce artificially generated sets human-like behaviours that reflect the decisions and actions performed by a real person. The aim of the project is to apply the artificial behaviour sets generated by the new artificial intelligence mechanism to background NPCs in computer games and as tools for intelligent environments research, potentially providing an alternative to using real people as participants in evaluations.

By taking advantage of computer game technologies, the various sources of freely available graphics and the global communities commonly found associated with open-source software, this project was able to create a three dimensional virtual intelligent environment, (similar to the MavHome's bespoke ResiSim world [7,8]), at a very low cost in terms of time and resources. Using the virtual world this project was also able to augment the University of Essex iSpace household environment, to create a mixed reality test-bed. Partially inspired by the virtual classroom of the MIRTLE project [4], the virtual component of the mixed reality intelligent environment enabled remote users to access the intelligent environment, along with real inhabitants.

Using the mixed reality intelligent environment as a platform, the first investigation showed conclusively that it was possible to use a combination of

mimicry (to gather usage data from real people) and genetic programming to produce new datasets, whilst preserving the essence of human-like behaviour captured from the intelligent environment. However, it was also revealed that in complex environments containing many stimuli and different intelligent devices was more difficult than in simpler worlds with a more limited set of possible actions.

The second investigation focussed more on the computer games applications of the new artificial intelligence mechanism. For completeness it was decided to test the artificial behaviour sets produced in a competitive scenario, specifically a tournament on a bespoke Battleships-inspired simulation. To better assess how significant the difference was between the performance of the profile mechanism and three other sources of controller data, a statistical analysis in the form of a Mann-Whitney U-Test was applied to the sets of results in Table 1. Using a five percent significant level, the U-Test analysis revealed that as expected the performance of the profile mechanism behaviour-sets was significantly different to that of real humans. However, the analysis also showed there was no significant difference between the profile mechanism behaviour-sets and those created using pure-mimicry or non-augmented genetic programming. When analysed alongside the average scores obtained by each category in the experiment these results are encouraging. They reveal the profile mechanism performed slightly better than the pure mimicry strategy on average, whilst producing many more usable behaviour-sets. It also didn't need to cheat to function, as was the case with the non-augmented genetic program, yet still produced similar performance results.

The final investigation featured in this paper returned to intelligent environments. By combining the knowledge gained from the research of the previous two investigations the project created a new *iWorld* modelling a simple bar design. Using a two-fold strategy, participants became familiar with the world by first playing a short drink-ordering game, (simultaneously creating a new mimicry profile by the system recording their usage data). Then the system applied a new evaluation strategy developed by the project, (inspired by the Turing Test) to assess whether the artificial profiles produced by the new mechanism were producing behaviours that could be considered human-like. From the results of the investigation shown in Table 2, it is reasonable to suggest that behaviour sets produced by the profile mechanism do display human-like characteristics. When directly compared to pure mimicry profiles (replicat-

ing the performance of real people) the profile mechanism generated behaviour sets were only correctly identified on nine of thirty usages, (30% of instances). Of the nine cases where the profiles were correctly identified the results also suggest that up to six instances may be due to guesswork by the participant as another artificial profile generated using the same method remained undetected. Later when the ratio of artificial profiles to pure mimicry profiles was reduced only in one instance out of ten (10%) was a correct identification made. These results compared to non-augmented genetic programming (87% correctly identified) and random drink / consumption selection, (53% correctly identified) show that the profile mechanism is capable of producing far more believable sets of behaviours, in the opinion of a selection of real observers.

The results generated by the profile mechanism during the last investigation suggest the human-likeness of the behaviour sets to be almost on par with that of real humans. This is encouraging as the project now continues to move forwards towards the aim of using the profile mechanism as a tool for providing alternatives to real humans in intelligent environments research. A key benefit aiding the use of the profile mechanism as an intelligent environments research tool is that no additional coding is required by a programmer in order to produce uniqueness between behaviour sets, as has been required in some projects in the past [17]. As long as some mimicry profiles exist that contain behaviour sets acceptable to a particular scenario, the mechanism will always be able to generate countless new designs that are similar in style but contain enough subtle difference to create a unique performance. The level of similarity between the original mimicry profiles and any generated by the mechanism can be easily altered by changing the mutation and reproduction frequencies of the genetic program or by defining a more specific fitness measure for identifying acceptable profiles.

As was seen in the final investigation the behaviour sets of the profile mechanism outperformed those generated by non-augmented genetic programming during almost every evaluation. Even when performing a specific task a real person would be influenced by their personality traits [27] and external features, (e.g. the ambience in the test environment). Factors such as these are practically impossible to measure and use as function or terminal nodes in a genetic program, which results in poor performance caused by the limited options available. By using intelligent environments and computer games to capture realistic usage data from real people the ge-

netic program of the profile mechanism can overcome these limitations. Although specific nodes are still almost impossible to define they would still be contained within the collective rules of the mimicry profile behaviour set so may create options that wouldn't otherwise have been considered.

5. The next steps

5.1. Multi-instance virtual intelligent environments

Another possibility this project is keen to explore with MMO *iWorlds* is using the multi-user capabilities to allow a single world to include multiple instances of the intelligent environment. Each instance could possess a unique layout, have an altered inventory or be running with different third-party programs (e.g. agents) in the background. Figure 13 shows an early example of a multi-instance virtual intelligent environment created as part of this investigation. Five intelligent households are present, each based upon a modified version of the virtual component from our original mixed reality world.

Users of a mixed reality intelligent environment containing a multi-instance world could select which household they wished to use as their virtual component, (e.g. by walking to the relevant building). The physical intelligent space would then automatically link with the selected virtual environment, adding any unique or dual-reality devices and agent programs installed to the system network. A multi-instance *iWorld* could also potentially also be used to observe and compare several instances of an intelligent environment running simultaneously.

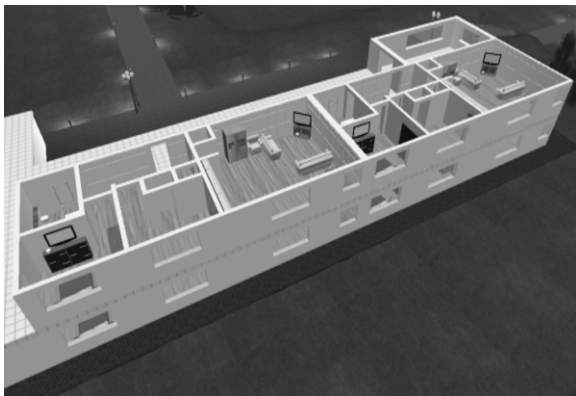


Fig. 13. Multiple household environment instances.

5.2. The iTown

This project is also keen to explore other potential benefits of applying MMO game technologies to intelligent environments, (both in a stand-alone and mixed reality context), and using artificial control systems generated by the profile mechanism as a research tools that provide a viable alternative to real people during evaluation sessions. To pursue these investigations a next-generation *iWorld* we refer to as the iTown, is currently under construction.

Built with the previously used technologies provided by RealXtend and Google but on a scale several times larger than any of the previous *iWorlds*, the layout of the iTown is designed to encourage users to explore the virtual environment by walking around and interacting with the various intelligent objects they encounter. The layout and contents are intended to reflect that of a typical small township with designated sectors for residential, commercial and industrial themed buildings. Some screenshots from the early design are shown in Fig. 14.

The residential sector of the iTown consists of twenty-four households, (remodelled from the earlier virtual worlds), which can be outfitted with static or intelligent devices and used in a stand-alone or mixed reality context. Some of these household instances will be used by the project for researching multi-instance virtual intelligent environments.

The commercial sector includes businesses such as, small independent shops, cafes, restaurants, a bank, utility company offices and a supermarket. Several civic buildings are also featured including, a health centre, theatre and library. Businesses each offer goods and/or services that can be purchased or used, offering a broad spectrum of potential interaction options and access points, to prompt the collection of user behaviour-sets that are both rich and variable.

Finally, the industrial sector consists of a 'factory' building, where users will be able to play a number of mini-games built into the world, (such as the Battleships simulation) allowing them to earn fictional currency rewards to spend on the items and services available from the commercial sector.

The Profile Mechanism will continue to be refined using the iTown world. One intention is to use artificial profiles generated by the mechanism to control a number of NPC inhabitants, who can interact with the world in the same manner as real users. This would allow a new phase of the Turing-inspired evaluation to be performed in a more detailed virtual environment. The human-likeness demonstrated can

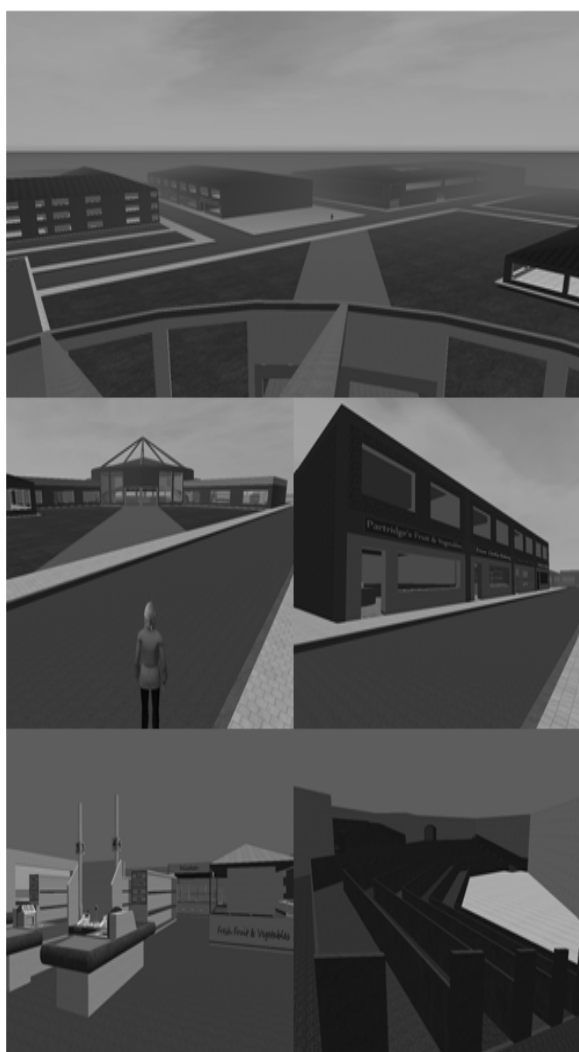


Fig. 14. The iTown virtual world.

then be compared to the results presented in this paper, gathered from the Bar iWorld. With its different zones and buildings the iTown world also provides a convenient platform for further investigation of the Personality / Personae concept mentioned earlier.

5.2.1. MMO Bar iWorld

The intelligent bar iWorld presented in this report is a prototype for a more elaborate three dimensional MMO environment designed for an international competition. Once deployed, the new virtual world will also be used to perform further evaluations for the next stage of this project [13]. Through large numbers of people accessing this world simultaneously from around the world can potentially quickly

produce large collections of usage data for use by the profile mechanism. Additionally, each user is another participant that can be used to evaluate the human-likeness of artificial profiles produced by the project mechanism.

6. Conclusions

This paper has presented ongoing research into using virtual and mixed reality intelligent environments as part of a mechanism to generate artificial control systems for simulated humans that produce realistic ‘human-like’ actions. Intelligent environments have been used as they provide an ideal means of monitoring the behaviours of real human users, and creating rule-sets based on their actions and decisions. These user profiles were subsequently used by a new artificial intelligence mechanism developed by this project, designed to generate new control systems for simulated humans. This ‘Profile Mechanism’ functions by using a combination of mimicked user data gathered from intelligent environments (iWorlds) and genetic programming. As the new artificial profiles were generated from data mimicked from real people, it was possible to preserve the structure of realistic actions and decisions within the new behaviour-sets.

To allow the initial usage data from real human users of intelligent environments to be gathered, a number of iWorlds simulating different scenarios had to be created by this project. This paper has discussed the various issues surrounding using three dimensional graphics to create virtual worlds for use as intelligent environments research tools. Previous research leading to the current focus was also discussed, focussing specifically on modifying computer games and related technologies to build virtual intelligent environments. After highlighting several key issues, a methodology has been presented which allows high-detail three dimensional virtual worlds to be created with minimal financial cost and during a timeframe likely to be compatible with an intelligent environments research project. By taking advantage of existing technologies, the presented methodology allows complex and flexible worlds to be created by individuals who may be lacking three dimensional graphics programming skills.

In addition to the creation methodology, a strategy used by this project was presented, for augmenting a virtual world with physical smart devices and other embedded technologies to produce a mixed reality intelligent environment. Subsequent sections dis-

cussed advantages of a using mixed reality in intelligent environments research and benefits that could be introduced by incorporating MMO features inherited from software used to build the virtual component.

With the *iWorlds* gathering usage data from real humans, the project then focused on researching how obtained behaviour-sets could be best applied to the genetic programming component of the Profile Mechanism. It soon became clear that a number of different strategies were possible, so a series of evaluations were created to assess which strategy produced the most 'human-like' replicas of people. Each evaluation was designed to test a different strategy for applying mimicked profiles to the genetic programming of the Profile Mechanism. The platform used as a test-bed was also changed to observe how the mechanism performed in different scenarios.

Of the three evaluations featured in this paper the first used a mixed reality household environment to gather initial usage data from real users. This initial 'proof-of-concept' testing found that while the strategy was viable, operators used by the genetic program of the Profile Mechanism could quickly distort the realistic integrity of original mimicked profiles. This was especially true for the mutation operator, which if left unrestricted could prompt the creation of artificial profiles containing unrealistic actions, such as only sleeping for a few minutes each day.

The second evaluation focused on the potential competitiveness of profiles created by the project mechanism when used in computer games. After making some modifications to the genetic program and choosing a different application strategy for mimicked user profiles, the updated mechanism was installed into a bespoke simulation, based on the classic 'Battleships' board game. Following this testing phase a Mann-Whitney U-Test analysis showed that the improved mechanism profile was still not capable of producing artificial profiles that could match the reactive performance of a real person. However, the same analysis did show that there was no significant difference between the performance of the Profile Mechanism and other artificial intelligence systems, (pure mimicry and non-augmented genetic programming).

Taking the knowledge gained from the previous two evaluations, the final investigation introduced a new testing strategy for evaluating the performance of the artificial profiles. The results from the first evaluation showed that if no restrictions were placed on mimicked data being used by the genetic program the artificial profiles created would soon become unrealistic when compared to the actions and deci-

sions of real humans. However, when restrictions were imposed on how the mimicry data could be used during the second evaluation a statistical analysis showed there was still a significant difference between the performance of artificial profiles and real people when playing the Battleships game. Additionally, the restrictions placed on the profiles of the Battleships game meant that it was almost always very easy to identify a real user compared to an artificial player, mainly due to the limited number of possible moves available to the artificial intelligence. As the ultimate aim of this project was to create artificial control systems to allow simulated humans to 'appear' realistic, a statistical analysis may not provide the best means of evaluating human-likeness. Therefore a new evaluation mechanism inspired by the 'Turing Test' was created to assess profile performance in the final investigation.

The final evaluation consisted of a simple Bar-World containing a drink collection point and four tables. Each table had one customer assigned and a waiter character collected orders and delivered requested drinks. The objective of the evaluation was for users to observe the collective performance of all four customer avatars, (i.e. the types of drink they ordered and how many they actually consumed). After observing the scenario evaluation participants were asked to identify which of the four customers they believed were artificial people, (rather than a replay of some data collected from a real user). Initially the evaluations used an even distribution ratio with two 'real' users and two artificial. This was later changed to three real and one artificial as the results collected suggested that many positive identifications may be a result of a lucky guess by participants. Collectively the results from this evaluation show that the Profile Mechanism installed in the Bar World was capable of generating profiles that could produce a human-like performance. For the revised distribution ratio only ten percent of the artificial profile controlled avatars used in evaluation runs were correctly identified by participants.

Following the presentation of results gathered from each of the three evaluations and a summary analysis, the remainder of this paper revealed the future intentions and directions of this project and its research focus. The intention is to eventually be able to combine both research branches of this project with a virtual intelligent environment, (potentially using a mixed reality setup) that could be inhabited by collections of simulated humans each performing unique and realistic (i.e. human-like) actions, by following behaviour-sets generated by the profile

mechanism. The results gathered from the Bar-World evaluation provide encouragement as this project continues its research into the Profile Mechanism and potential applications in both computer games and intelligent environments research.

Acknowledgements

The authors would like to thank Intel and the Creative Science Foundation, (www.creative-science.org) for their ongoing contributions for the Bar iWorld experimentation phase of this project.

References

- [1] H. Brighton and H. Selina, *Introducing Artificial Intelligence*, Icon Books UK, Cambridge, 2003, ISBN 1-84046-463-1.
- [2] V. Callaghan, G. Clarke and J. Chin, Some socio-technical aspects of intelligent buildings and pervasive computing research, *Intelligent Buildings International Journal* **1**(1) (2007).
- [3] V. Callaghan, G. Clark, M. Colley, H. Hagra, J. Chin and F. Doctor, Intelligent inhabited environments, *BT Technology Journal* **22**(3) (July 2004), Kluwer Academic Publishers, Dordrecht, Netherlands.
- [4] V. Callaghan, M. Gardner, B. Horan, J. Scott, L. Shen and M. Wang, A mixed reality teaching and learning environment, in: *Proc Hybrid Learning 2008 (ICHL'08)*, Hong Kong, LNCS, Vol. 5169, Springer, 2008, pp. 54–65.
- [5] J. Chin, V. Callaghan and G. Clarke, Soft-appliances: A vision for user created networked appliances in digital homes, *Journal of Ambient Intelligence and Smart Environments (JAISE)* **1**(1) (2009), 69–75, ISSN 1876-1364, IOS Press.
- [6] G. Clarke and V. Callaghan, Ubiquitous computing, informatization, urban structures and density, *Built Environment* **33**(2) (May 2007), 196–212, ISSN: 0263-7960, Alexandrine Press.
- [7] D.J. Cook, M. Youngblood, E. Heierman, K. Gopalratnam, S. Rao, A. Litvin and F. Khawaja, MavHome: An agent-based smart home, in: *Proc. IEEE International Conference on Pervasive Computing & Communications*, 2003, pp. 521–524.
- [8] D.J. Cook and S.K. Das, *Smart Environments: Technology, Protocols, and Applications*, Wiley Inter-Science, 2005, ISBN 0-471-54448-5.
- [9] B.J. Copeland, ed., *The Essential Turing: The Ideas that Gave Birth to the Computer Age*, Oxford University Press, UK, 2004, ISBN 0-19-825080-0.
- [10] M. Davies, V. Callaghan and M. Gardner, Towards a mixed reality intelligent campus, in: *Proc. 2008 International Conference on Intelligent Environments (IE'08)*, Seattle, USA.
- [11] M. Davies, V. Callaghan and L. Shen, Modelling pervasive environments using bespoke & commercial game-based simulators, in: *Proc. of the 2007 International Conference on Life System Modelling and Simulation (LSMS'07)*, Springer, Shanghai China, September 2007.
- [12] M. Duller, R. Tamosevicius, G. Alonso and D. Kossmann, XTream: Personal data streams, in: *Proc. of the 2007 ACM SIGMOD International Conference on Management of Data*, ACM, Beijing, China, 2007, pp. 1088–1090.
- [13] S. Egerton, M. Davies, B. Johnson and V. Callaghan, Searching for free-will (a competition), in: *The 2nd International Workshop on Creative Science – Science Fiction Prototyping for Technology Innovation (CS'11)*, Nottingham, UK, 2011.
- [14] K.D. Forbus and W. Wright, *Some Notes on Programming Objects in the Sims™*, Northwestern University, 2001.
- [15] Google, 3D Warehouse, <http://sketchup.google.com/3dwarehouse>, Retrieved: 21st December 2010.
- [16] IIEG, iSpace, <http://dces.essex.ac.uk/Research/iieg/idorm2/index.htm>, Retrieved: 20th February 2010.
- [17] T. Ishida, Y. Nakajima, Y. Murakami and H. Nakanishi, Augmented experiment: Participatory design with multi-agent simulation, in: *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI-07)*, 2007.
- [18] J.R. Koza, Genetically breeding populations of computer programs to solve problems in Artificial Intelligence, in: *Proc. of the Second International Conference on Tools for A.I.*, IEEE Computer Society Press, Herndon, Virginia, November 1990, pp. 819–827.
- [19] J.R. Koza, Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems, Stanford University Computer Science Department Technical Report STAN-CS-90-1314, 1990.
- [20] J.R. Koza, Evolution and co-evolution of computer programs to control independent-acting agents, in: *From Animals to Animats: Proceedings of the First International Conference on Simulation of Adaptive Behavior*, J. Meyer and S.W. Wilson, eds, Paris, September 1990, pp. 366–375.
- [21] J.R. Koza, Introduction to genetic programming, in: *Proc. of the 2007 GECCO Conference Companion on Genetic and Evolutionary Computation*, London, UK, 2007, pp. 3323–3365.
- [22] Linden Lab., Second Life, <http://secondlife.com/>, Retrieved: 27th November 2010.
- [23] P. Milgram and A.F. Kishino, Taxonomy of mixed reality visual displays, *IEICE Transactions on Information and Systems* **E77-D**(12) (Dec. 1994), 1321–1329.
- [24] D. Nettle, *Personality: What Makes You the Way You Are*, Oxford University Press, Oxford, UK, 2007, ISBN 978-0-19-921142-5.
- [25] A. Nijholt, J. Zwiers and J. Peciva, Mixed reality participants in smart meeting rooms and smart home environments, *Personal and Ubiquitous Computing* **13**(1) (January 2009), 85–94, Springer-Verlag, London.
- [26] OpenSim, Main Page – OpenSim, <http://opensimulator.org>, Retrieved: 30th December 2010.
- [27] D. Pennington, *Essential Personality*, Arnold, Hodder Headline Group, London, UK, 2003, ISBN 0-340-76118-0.
- [28] RealXtend, realXtend – Open source platform for interconnected virtual worlds, <http://www.realxtend.org>, Retrieved: 30th December 2010.
- [29] T. Van Kasteren, A. Noulas, G. Englebienne and B. Kröse, Accurate activity recognition in a home setting, in: *Proc. 10th Conference on Ubiquitous Computing*, Seoul, Korea, 2008.