



City Research Online

City, University of London Institutional Repository

Citation: Papakonstantinou, S. and Brujic-Okretic, V. (2009). Prototyping a Context-Aware Framework for Pervasive Entertainment Applications. Paper presented at the Games and Virtual Worlds for Serious Applications.

This is the unspecified version of the paper.

This version of the publication may differ from the final published version.

Permanent repository link: <https://openaccess.city.ac.uk/id/eprint/207/>

Link to published version:

Copyright: City Research Online aims to make research outputs of City, University of London available to a wider audience. Copyright and Moral Rights remain with the author(s) and/or copyright holders. URLs from City Research Online may be freely distributed and linked to.

Reuse: Copies of full items can be used for personal research or study, educational, or not-for-profit purposes without prior permission or charge. Provided that the authors, title and full bibliographic details are credited, a hyperlink and/or URL is given for the original metadata page and the content is not changed in any way.

Prototyping a Context-Aware Framework for Pervasive Entertainment Applications

Stelios Papakonstantinou¹ Vesna Brujic-Okretic²
Dept. of Information Science
CITY University
London, UK
stelios@soi.city.ac.uk¹, vesna@soi.city.ac.uk²

Abstract — This paper presents research and development of dedicated system architecture, designed to enable its users to interact with each other as well as to access information on Points of Interest that exist in their immediate environment. This is accomplished through managing personal preferences and contextual information in a distributed manner and in real-time. The advantage of this system is that it uses mobile devices, heterogeneous sensors and a selection of user interface paradigms to produce a socio-technical framework to enhance the perception of the environment and promote intuitive interactions. Representation of the real-world objects, their spatial relations and other captured features are visualised on scalable interfaces, ranging from 2D to 3D models and from photorealism to stylised clues and symbols. The framework is fit for use in unknown environments and therefore suitable for ubiquitous operation. The presented prototype is multifaceted and capable of supporting peer-to-peer exchange of information in a pervasive fashion, usable in various contexts. The modalities of these interactions are explored and laid out particularly in the context of entertainment.

Keywords - mobile devices, context-awareness, virtual reality, augmented reality, pervasive entertainment, social interaction

I. INTRODUCTION

People, who cope with hi-tech solutions during their work, require technology to offer more in terms of satisfaction, when introduced in personal and social scenarios. Computer assisted entertainment allows people from any age, gender or background to find a game type that will cover their needs. Since the early video games in 1970s, like *Pong*, we have observed a dramatic advance of how technology was applied to entertainment, not only in terms of AI engines and computer graphics, which receive special focus from the community, but also how it affects the nature of game playing. New sets of technologies did not only manage to simulate games that were traditionally played in the real world, but also reached a point, which made them capable of being the actual game, which provides joy to the participants. The main reason is the intriguing puzzles that have been applied on inventive scenarios, which aided towards the expansion of the target audiences. Recent applications, like *Crysis*, have utilised advanced graphic engines in order to provide high interactivity and immersion with the virtual world, but also proved to be detrimental in terms of the physical and social activities of

their users [1], in comparison to traditional games. Similarly, mobile versions of such games have been developed, with reduced processing demands, but with evident potential to become mainstream.

The introduction of concepts like *Ubiquitous Computing* [2] and *Mixed Reality* [3] contributed towards the development of new methodologies for entertainment and provided the foundation to bridge the gap between independent and social user behaviour [4]. Consequently, there has been a natural expansion towards the spatial, temporal and social connotations that any solution needed to address, including their advantages and disadvantages [5]. The acquisition and management of quantifiable user-related and environmental parameters (i.e. context) was found to be a means for achieving this kind of functionalities and for providing the resources to connect the real with the artificial world, where the game takes place. Real-time, context-sensitive frameworks have operated for long time in business and military applications, but have just recently started serving purposes related to commercial user entertainment. An analytical review has been published by Rashid et al. [6], who describe applications that employ location context in mobile gaming scenarios. The examined applications run on mobile devices, rendering them operational in any physical environment or while in motion. Similar systems that can explore virtual environments [7], augmented environments [8], or both [9] have been introduced with application in navigation. The advantage of these engines is that they can combine virtual and physical space and assist the decision making process of each user through advanced user interfaces.

Several attempts have been made to classify entertainment solutions, which are partially persistent in a computer generated environment and partially deployed on the real surroundings. Most of these classifications use loosely the term *Pervasive Game* as discussed by Nieuwdorp [10]. Interesting reviews of such applications have been given by Magerkurth et al. [1], who expand and define various systems with main goal the amusement of their users. The term *Ambient Game* has been used to define solutions that conform to a specific balance between the commitment required by the user and the distance that is travelled [11]. *Trans-Reality* has

been used to describe a subset of ubiquitous applications that use MR techniques for implementing various genres of games [12]. All examined types fit to the conceptual frameworks drawn by Walther [13] and Hinske et al. [14], as well as the applicability concerns presented by Capra et al. [15]. A description of several issues in the underlying technical infrastructure, in terms of the interfaces, is provided by Broll et al. [16] and in terms of subject localisation by Benford et al. [17]. Ultimately, an evaluation platform capable of examining ubiquitous entertainment solutions has been released by IPerG, an EU-funded project [18].

In this paper we are going to present a technical framework that can support pervasive entertainment functionalities by making use of real-time context variables and by registering the user, in a number of immersive environments, which can simulate and enhance the real surroundings. Physical and some limited social interactions that occur in the real world can be modelled and presented through the visualisation interfaces, which are complementing each other and include a 2D map representation, a VR interaction engine and a custom-tailored AR interface. The system runs on a mobile device platform that enables and promotes user activity in the real world. The design of the system architecture was envisaged in order to support ubiquitous operation. It does not require any special characteristics to exist in the environment and it can present its functionality through the use cases described later in this paper. Although this framework can support applications in various domains like wayfinding and exploration of unknown places, in this paper we are going to focus on the potential it has, when operated under the governing rules and mechanics that competitive or goal-oriented, pervasive gaming instructs.

II. ENABLING CONCEPTS

A. Ubiquitous Computing

The pioneer who formed the basis for the concept of ubiquitous computing systems was undoubtedly Mark Weiser. He believed that the most important technologies are those that work transparently from the user and provide contextualised services through devices and sensors that are distributed in the physical environment [2]. These interconnected devices assemble the hardware layer, which enables ubiquitous system operation. Recent commercial trends have pinpointed that the most usable and accepted by the public solutions tend to be mobile communicating devices like handheld PCs, Personal Digital Assistants (PDA) and mobile phones. In this paper we are examining medium to small-sized devices, with battery capacity close to 1500 mAh, unrestricted, native programmable memory and a mobile operating system with multimedia and networking capabilities. This type can reflect current technological progress and produce a good performance to size ratio. Furthermore, state-of-art products embed context-sensitive sensors, which can generate additional data and offer improved functionalities. Despite the advantages of such devices there are several limitations, which can affect the gaming experience of the

users. The small display size, for example, needs to be efficiently managed in order to accommodate any visualisation and interaction requirements.

In an attempt to disseminate the principal components of ubiquitous systems, Saha and Mukherjee identified the networking layer as the next core element of pervasive systems [19]. Networks ranging from Bluetooth Piconets to wide scale 3G infrastructures can form the platform, which enable remote clients to transparently connect and share context with other remote entities. These entities may be other actors sharing similar interests or participating in a group task, or a centralised system that may expect real time data in order to return suitable information. In our framework we regard the network as a valuable source of context, which will provide the resources to the context management engine and consequently to the visualisation interfaces. Even though there is a large number of available networking stacks to select from, each implementation introduces its own boundaries (e.g. operational range) that may restrict the omnipresence of mobile applications. Although several telecommunication protocols can satisfy the requirements, service providers apply restrictions on arbitrary data exchange. Specifically, P2P interactions and service hosting on mobile devices is complicated and costly to implement because telecommunication proxies masquerade network addresses, thus rendering the distribution of content limited to request/response mode and not proactive. Other issues that pervasive system developers need to handle, while developing multi-user Collaborative Virtual Environments (CVE) are those inherited by distributed networking research. Namely, these include high latency and low bandwidth management, heterogeneity of infrastructure, scalability issues, synchronisation and failure handling, security and privacy.

B. Real-time Context Adaptation Issues

In order to satisfy the requirement of invisibility [19], which pervasive system architectures define, a middleware component has been introduced in the proposed framework. Purpose of this entity is to establish a connection between the network nodes, which act as source of context, and the context management layer. Communication on this channel is bidirectional. Data received from remote entities is transformed and fused into structures that can be queried by the application and vice versa. This way, the upper layers and effectively the user, do not need to hold any specific information about the remote entities or how to contact them.

Many researchers who have tried to group context in distinct categories observed the matter from different research perspectives. The result was domain-specific classifications. Each classification scheme focuses on the satisfaction of certain user information needs. Ambiguities between context categories may rise. As described by Mostefaoui et al. [20], context can be fused to the engine through *sensed*, *derived* and *explicitly provided* means.

Main challenge for our context management layer was the smooth execution of the process that involves perceiving

context, processing it and changing behaviour towards the user. To depict the result of this process, augmentation of the user interface with descriptive information about the current, personal and environmental situation is provided. The developed context engine conforms to the classification, which examines context in a way that is relevant to the retrieved source. This scheme can distinguish data that was either collected from hardware sensors, or explicitly from the user, through the UI, and lastly, by the internal mechanisms that provide generation and enhancement of locally stored information. In this case, ambiguity of context variables can also be noticed. Orientation of a moving object is accurately obtained by querying a digital compass. Furthermore, for a specific span of time, orientation can be measured by calculating former and present location coordinates, sensed by a GPS. Thus, we promote the utilisation of redundant context management techniques in order to support context retrieval from secondary available sources as a fail-safe mechanism.

Common ground that most mobile context-aware applications share can be found in the use of location context. Mobility of devices in the environment is the main reason that triggers exploitation of such information. Proximity of Points of Interest (POI) and other participating entities (i.e. users, virtual and physical objects) can cause potential collaboration and interaction. Additionally, location sensors, like GPS receivers, have been widely adopted, even in out-of-the-box solutions for mobile devices. Accuracy of measurements by these sensors is increasingly evolving and can prove satisfactory not only for sensed context (e.g. location) but for derived (e.g. speed) as well.

III. FRAMEWORK ARCHITECTURE

Pervasive games significantly affect how computing devices are deployed and how people interact with the resulting interface paradigms, because ubiquitous information access is the main goal. The challenge for software developers is to create applications that continuously adapt to dynamic changes in the environment even if people move in the real world and even if the underlying network architecture can offer only limited services. The essential elements for the global design of pervasive systems have been identified and modelled into four categories by Saha and Mukherjee. These are *mobile devices*, *networking*, *middleware* and *applications* [19]. Conceptually, interaction interfaces belong to the application subsystem. However, they functionally depend on the output of all layers. Figure 1 shows the elements that compose the implemented prototype, with emphasis on the interaction platform.

The software platform is based on Windows Mobile 5/6 PocketPC edition. The application was developed in native code (C++), which provided accurate and resource-efficient handling of low-level performance issues. Windows Mobile devices do not conform to a standard specification for Java Profiles (JSR), making applications incompatible and device specific. In contrast, our system can be characterised as platform-specific.

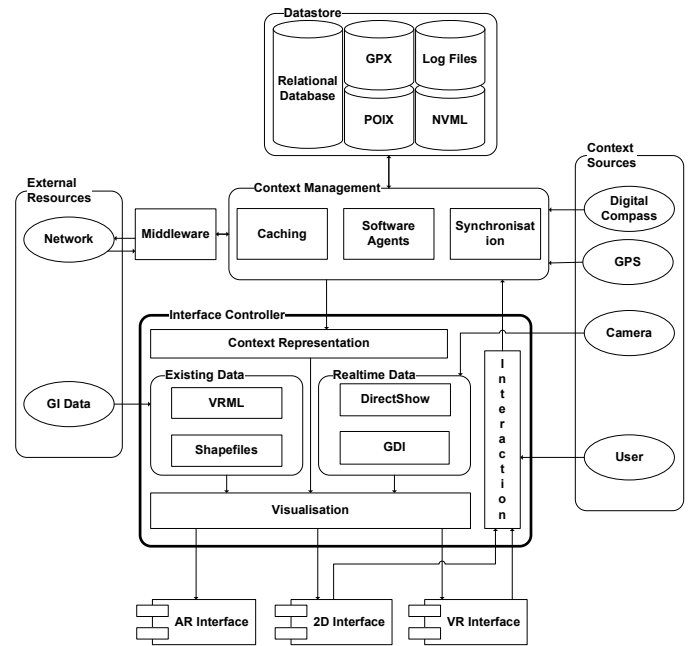


Figure 1. Prototype Client Architecture

A. Context Management System

The core structure is formed out of two principal subsystems, which receive input from the external sources and are capable of rapidly controlling the information flow between each other. The first one is the Context Management System (CMS) and the second one is the Information Presentation System (IPS). The former unit is a low-level subsystem which receives input from the peripheral resources and is responsible for keeping the geometric model that governs the application updated with relevant information either from local or remote entities. Input is considered the real-time data generated by the sensors or network updates and any relevant locally-stored information. This subsystem is also responsible for implementing all transformations in order to keep the model coherent and for exchanging information with the remote entities. Additionally, movement of Non-Player Characters (NPC) that may exist in the scope of the game is accommodated by this structure. During configuration and debugging of the application, when the network infrastructure is not fully operational, the CMS generates and simulates artificial movement of remote objects. In this subsystem algorithms, which enhance the accuracy, performance, prediction and interpretation requirements of the application are executed.

B. Information Presentation System

Following next, the IPS is used for reflecting any changes provoked by the CMS and for accepting explicit user input. Its main purpose is the visualisation and interaction with the environment and any relevant objects in it. To accomplish accurate visualisation functions in all interfaces (i.e. 2D, VR and AR), this subsystem manages input from 3 sources. Two of them are used to simulate the real surroundings and the other interconnects objects within the real world with objects in the virtual world. The latter is an interface to the CMS,

which receives numerical and textual descriptions (i.e. metadata) of objects and creates virtual representations of them. These can be interrogated in any UI and can vary depending on the type of interrogated entity (e.g. avatar for human in 3D or icon in 2D). For the 2D and VR interface the environment is modelled out of existing data. These can be either shapefiles for 2D or VRML models for 3D, both generated out of the same spatial dataset, by using ESRI ArcGIS. In contrast, the AR interface processes real-time data from the camera. More information about interactions can be found in subsequent paragraphs.

C. Modes of Operation

In the following context diagram of our system, the reader can observe that except the core entity, which includes the IPS and DRS subsystems, the user is considered as active source of context as well. This occurs because potential interactions are interpreted and influenced by the actual situation that the user is experiencing. The data sources can be software or hardware based. The substantial meaning of these data sources signifies them as independent subsystems.

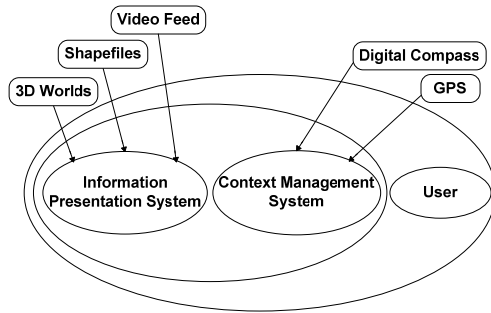


Figure 2. Context diagram of the developed system

Fundamentally, interaction in the 2D and 3D environments is supported by the two operational modes of the system: the *sensor-controlled* mode and the *user-configurable* mode. This way there is a potential for the system to meet a variety of user needs, such as naïve search, primed search and exploration. These modes of operation provide means to form a multiple level-of-immersion application - visible from absolute egocentric to any allocentric perspective. In the sensor-controlled mode, interaction takes place by considering context input and placing the user in the appropriate position, with analogous orientation that corresponds to real world behaviour. This mode is designed for real-time game playing, whereas the manual interaction mode assists in the exploration of remote locations and the enhancement of the decision-making process. Manual interactions are triggered explicitly by the user. This way, any place in the virtual world can be rapidly examined and its surroundings evaluated. In user-configurable mode, any viewpoint is supported. In contrast, the sensor-controlled mode supports first-person, oblique and bird-eye views of the scene. In this mode, each perspective complements the other and is up to the user to select the one that is more familiar and comfortable with. Additional elements that could enhance user experience depend on user personalisation preferences and involve further technical

issues. For instance, the user perspective and the current orientation may be identical to the real, surrounding scene, or may vary depending on whether the user is interested in some remote site features. In terms of pervasive operation, sensor and manual control of the interfaces can reflect the active and passive context reactivity of the system.

IV. USE OF SENSORS FOR SELF-LOCALISATION

The developed system utilises three sensors, which fuse the context acquisition engine with valid measurements. Real-time location information can be stored in the form of GPX, POIX, NVML and human-readable log files. Furthermore, a local database implementation is under development for future application of KDD and Data Mining techniques. In real-time pervasive applications the use of geometric location models is suggested. Although these models provide only simple spatial reasoning, in contrast to symbolic and hybrid models, their main advantage is that the measurements can be fused directly to influence interactions, without the need to introduce additional transformations. This way, expensive device calculations are kept to a minimum.

The accuracy of context received by the GPS is of utter importance. During debugging, we tested several internal and external commercial devices with embedded GPS chipsets. Mostly, these were SiRF Star II/III solutions with average declination in dense urban areas, lying between the expected boundaries of up to 15 meters. Some mobile devices supported A-GPS implementations with partial indoor functionality (HP iPAQ hw6915). Newer implementation based on Qualcomm's integrated gpsOne chipset provided higher accuracy and more stable readings (HTC Touch Diamond). The context retrieval engine can translate GGA, GLL, GSV and RMC NMEA0183 sentences, passed over a serial link (i.e. Bluetooth) between the external sensor and PDA communication ports.

The second sensor, which can provide real-time orientation information leading to gesture interactions, is a digital compass. The product was based on Honeywell HMR3300 magnetic sensor chipset. Similarly, it connects to the PDA via serial protocol and produces three value sentences. They are in ASCII format and correspond to yaw, pitch and roll information. The manufacturer reports that achieved heading accuracy is in the region of 0.5 degrees with 0.1-degree resolutions and tilt ± 40 degrees.

It was found obligatory for the AR interface, to handle streams of video generated by the camera of the mobile device. Almost every new commercial device embeds an imaging sensor. Concurrently encoding and compressing video is a demanding procedure for current mobile processors. This is why this context type is not stored by the system for future reuse and is employed only for real-time processing. The mobile OS delivers functions to initialise the camera drivers and DirectShow offers source, transform and rendering filters that can form a media-streaming layer for the application. Image quality was between 1.3 and 5.0 MP. It is important to note that static output formats are in higher resolution, when compared to animating content, as most

cameras provide less quality for streaming configurations. The maximum resolution that was received from the imaging sensor was 2 MP, in video mode.

V. SUPPORTED INTERFACE PARADIGMS

Visualisation and interaction in mobile Mixed Reality applications, which offer ubiquitous operation and environmental representation, functionally depend on the precise registration of the subject (i.e. user, sensors and mobile device) to the available interfaces, in relation to real world conditions. The described framework analogously registers the user to the map, VR and AR frameworks by examining 6 Degrees of Freedom (DOF), 3 from GPS and 3 from compass. Reactivity of the application is triggered after detecting changes on retrieved context. Presenting information about objects in the real world can take place by querying a pool (POIX or DB) for location information (longitude, latitude, sea-level height, type and description). Demonstrating synthetic information on each interface can occur by comparing context from the viewpoint of the subject and the remote resource. As a result, descriptive information can be visualised in relation to the actor and the real world. The benefit of this methodology is that it can simulate and enhance the cognitive map of a person. This is accomplished by matching the user's cognitive Frames of Reference (FOR) with the registration parameters of the represented scene [21]. Having multiple, concurrent perspectives of the real world is invaluable for functional pervasive entertainment applications. Additionally, binding advanced UI with physical interactions encourages the transition from traditional computer games to pervasive entertainment [22]. The following paragraphs describe some special characteristics and functionalities of each user interface and the advantages that pose over the rest.

A. 2D Maps

Although traditional map representations are becoming dated, commercial applications are reluctant to discard them and try to boost their functionality by coupling them with supplementary enhancements. The reason that this kind of interface is still invaluable springs from the fact that it can satisfy certain user needs. It is easier for somebody to estimate the distance between 2 points by observing a map, enabling better decisions. Furthermore, scalable maps have been a productive tool for the GIS community, because they can hold overlaid phenomena and features of the real world, in a controlled fashion. The developed application can present vector visualisations of the environment by processing geo-referenced shapefiles. This is the standard format for exchanging geographic data, supported by ESRI. The user has a vertical perspective of the represented space and all interactions take place in abstract 2D scenes. Real-time context is classified into categories and each type is placed on a specific layer. The Z-order of each layer and the icons that are included depend on user preferences and current situation. Visualising the movement of the subject requires the context-controlled mode of the system to be enabled. Interaction is triggered when the user moves in the real world and consequent position updates occur. Zooming on the digital

map is accomplished explicitly by the user, with the stylus or the navigation button of the device. Automatic zooming takes place, when the distance between the user and the target reaches a specific threshold. Furthermore, rotation can occur either explicitly or via compass input. Furthermore, each actor can create own POIs by clicking on any part of the map and providing relevant metadata. These are appended in new or existing POIX files and can be used as game locations, if shared with other users.

B. 3D Virtual Environments

The introduction of 3D technologies added interactivity and dynamism in the visualisation of spatial information. The realism, which 3D representations of the real world provide, is exponential and complementary to what map visualisations have been offering. 2D representations, although efficient and popular, are fairly limiting regarding the amount of information that can present, compared to 3D representations that carry much more information. Controlled overlaying of objects in the virtual world, allows more information to float in a mobile device display, without increasing the complexity of the interface. Thus, navigation in a 3D environment appears more realistic and engages users for further interactions. The proposed framework uses VRML to establish the Virtual Environment (VE). Although it does not provide advanced functionalities like those found in Direct3D, OpenGL ES and M3G, this format can hold geo-referenced information (GeoVRML) and is still a common standard for exchanging 3D content over the web. The 3D models have been developed and enhanced with semi-automatic techniques, out of spatial datasets, originating from the shapefiles described in the former paragraph. As a result, the detailed 2D and 3D environments conform to the same location model and coordinate systems, which makes interaction design for both interfaces easier to capitalise.



Figure 3. a) Oblique & b) Allocentric VR perspectives

The user may observe the environment from an egocentric, an allocentric, or from a birds-eye perspective at any time, as seen in figure 3. It was found that information overload and occlusion could be effectively minimised by implementing all views. Interaction is driven by the context-controlled method (sensors) or manually, by the user. The sensor mode verifies alterations in position and orientation context and presents them to the display. Movement and gestures in the real world can be simulated in the VE. In the user-controlled mode, interaction with the device touch screen is required. The user can have a variable viewpoint of the scene so that he can explore the 3D world effortlessly. When the egocentric

perspective is selected, the viewpoint is parallel to the ground and the height is accumulated by the GPS. The oblique perspective raises the viewpoint by 50 meters and produces an inclination of 45° towards the ground. Birds-eye view is particularly useful if the actor moves in urban canyons where 3D positioning (≥ 3 GPS satellites in line-of-sight) is not available. Relevant features of the real world are presented by adding distinct 3D elements, like avatars or billboards and descriptive text annotations. Furthermore, the realism of the VE can be enhanced by examining temporal context as well. The colour of the sky and the position of a spotlight that represents the sun can vary depending on the time of day.

C. Augmented Reality Interface

Recent advance in the field of computer vision produced new set of technologies that provide photorealistic images, which can be queried and adopted by real-time context-sensitive engines. Traditional AR has been applied on pervasive systems as a mean to register the user in the synthetic environment, in accordance to real position and orientation. Most implementations utilise either a fiducial tracking solution or a natural feature recognition method. The fiducial tracking approach depends on the detection of predefined patterns in the real world and the estimation of the camera pose compared to those patterns. Most patterns come in the form of planar, multi-coloured, symmetrical textures, with some distinct characteristic for easier differentiation. This method makes the operating algorithms vulnerable to image quality, light conditions and texture quality, which means that in certain scenarios, tracking efficiency is severely affected. Moreover, because of the scarcity and difficulty of locating markers and populating them to the real world, this method restricts operation to confined environments. Exceptional work in the field of natural feature detection has been done by Reitmayr [23]. A model-based hybrid tracking system has been utilised, which combines an edge-based tracker, gyroscope readings, measurements of gravity and magnetic field and a back store of reference frames. The numerous sensors that are queried, effectively, make use of this system exclusive to research purposes. Another factor that makes the application of the previous 2 methods prohibitive for mobile phone operation, is the processing power required by the CPU and graphic subsystem of the device.

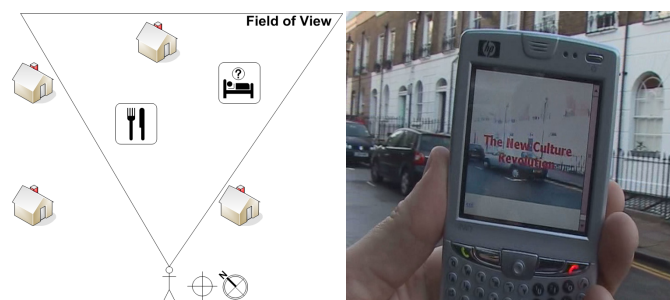


Figure 4. The use of AR: contextualising location & orientation information

For ubiquitous service operation of such technology, a new approach is required. The Augmented Reality interface that

was developed during the course of this project is capable of amalgamating the scene and selected surroundings that are currently observed by an actor, with spatially referenced context descriptions, by utilising a mobile device and a number of context-sensitive sensors. Similar to the VR interface, for ubiquitous AR, the location and orientation parameters of the device are the essential information that must be available to the CMS layer. Furthermore, the proposed AR implementation assumes that the underlying engine has knowledge about the spatiotemporal coordinates of specific objects or POIs in the real world. By being aware of the local spatial and remote spatiotemporal context, the system is capable of superimposing digital information on precise locations on the device display. The specified implementation can be easily ported into most kinds of mobile devices, as the interface is decoupled from the functionality and the only hardware requirement is the physical existence of the sensors. To prove that, we implemented the AR interface in two versions. The first was embedded in the standalone application (Aura) and the second was a reusable ActiveX component (ARIE), which can be integrated in diverse host applications (LOCUS). Both solutions enable the incorporation of graphical information with the media streaming layer. The valid use case flow for delivering ubiquitous AR functionality is presented below.

1. Retrieve position & orientation for local entity.
2. Retrieve position (& orientation) for remote entity.
3. Calculate point coordinates of Field-of-View polygon.
4. Check if remote entity lies inside the FOV boundaries.
5. Calculate distance between local and remote entities.
6. Get coordinates for tangent side of FOV polygon.
7. Map the tangent side on the device display.
8. Calculate the image coordinates for the remote entity.
9. Make the transformations from world to camera and from camera to image coordinate systems. (Camera Modelling)
10. Superimpose text, image or a 3D model over the specified pixels, on the device display. (Rendering)

The measurements retrieved by the sensors have proven to be sufficiently accurate for representing features of interest, of the real environment, in an estimated location relative to the device's actual orientation and position. The frequency of updates and the small derivation from the genuine location poses limitations, when the requirements instruct for more advanced functionality that will assist the game play. An example can be found in [23], where alternative textures are mapped on the face of real world buildings, which in our case may be used for simulating the after-effects of an artificial game battle. The aim of the AR interface is to augment a live camera feed of the real world scene with virtual information in real time, achieving registration between the virtual and real, while providing a convincing mixed reality experience for the users of the system. Most virtual information has been confined to textual annotations and descriptive, 2D symbols

providing information about a user's surroundings, or navigational information. Although not already achieved, the final deliverable will rendered virtual, 3D objects in the real world scene.

Hand gestures are used to interact with the real world. The user can pan the device around in order to interrelate with the environment and retrieve information about objects in it. For locating game objectives, directional aids are applied. This includes textual descriptions about the target, distance calculations and arrow representations that point towards the objective. For exploration scenarios information about entities in close proximity is offered. This includes self-explanatory symbols and textual descriptions, which let the users comprehend and decide dependant upon their preferences. In any pervasive scenario, directed guidance is provided towards the next goal of the game. This may be either locating an object or person, reaching a checkpoint or unlocking the next level of interaction and visualisation functionalities.

VI. PERVASIVE SCENARIOS FOR ENTERTAINMENT

The main purpose of our framework is to provide users with a common platform that will promote further interactions. These interactions can take place either between the user and relevant objects of the simulated environment or between the actors of the system. In the latter case, the system acts as a mediator that provides the *rules of engagement* between the participating entities. These rules are based on user-specified criteria (i.e. user context) and are established through the IPS. There are several potential capabilities of the implemented interaction and visualisation environments, but we examine those that can literally or metaphorically, *bring people together*. This can happen by applying pervasive game-like scenarios, which will trigger social interaction between the participating entities. Based on each scenario further activity in the real world is necessary. Physical activity, with the guidance of a computing device that handles context parameters, is the basis of pervasive computing. Engaging users becomes more effective, when the application is executed in multiuser environments, rather than in single-user modes (e.g. LBS). In comparison to independent modes of play, social interactions can intensify user engagement with the system.

The following paragraphs, describe a set of pervasive scenarios, which are currently under development. All of them conform to specific requirements but comprise several differences in functionality and purpose. These use cases need novel interfaces for nominal operation and specifically they utilise VR but extend to AR, as well. All established communications should conform to explicit personalisation and privacy rules.

A. Confined Space

The first scenario takes place in a restricted environment (i.e. Formula 1 track). The advantage of a restricted environment is that the LOD of the virtual world can be enhanced to the maximum. The stakeholders, who host the service, are responsible for sustaining a dedicated, low-latency

network infrastructure and for distributing contextual information to the participants. Promotion of this service targets the full-exploitation of the available space. The use case includes one or more actors and at least an F1 car that will embed a positioning device and a transmitter. This way car location is continuously available to the public. The system enables competitive racing between the users and any driver, while the race is still taking place. This is accomplished by accumulating almost real-time data. Except real-time competition, the modes of play may include observation of the actual race from any point of the track and exploration of the surrounding space. Effectively, this can prove helpful because actual tracks are enormous and several hours are needed in order to fully explore them. The quality of the modelled environment is a determinant factor towards the satisfaction of the user. Additionally, an AR implementation may be useful for retrieving information about each car in the race, when it passes in front of the user. Sensed, derived and explicit data can be used. Advantage of this scenario is that anybody with a compatible mobile device can participate in an event, while it is taking place.

B. Open Space

The second scenario is mostly applicable to urban environments because people have to be geographically close to each other, to trigger interactions. It can have several modes of play and specific extensions can be defined that would make it more intriguing for the users. In the action mode, AR can be used to *seek and destroy* an opponent. In adventure, location-based context is invaluable for finding and manipulating any objects of the environment. Furthermore, in adult mode social behaviour is triggered through personalised context, fed to the application explicitly by the user. A 3D urban model can be automatically generated out of spatial data and downloaded to the device, or accessed in a distributed fashion. Optimally, information about the participating characters should be modelled, based on independent privacy concerns. By implementing both VR and AR interfaces, detailed texturing of the VE is no longer a necessity but optional, because a photorealistic representation of the environment is made available by the onboard camera. Some of the advantages of this scenario are that it can run in variable-sized areas and can trigger social interaction between users. In the literature, this kind is referred to as pervasive MR games.

C. Evocative Space

The last scenario focuses on virtual observation and remote exploration rationale and can take place in any space with interesting context (i.e. marketplace). In this use case one or more actors need to participate. The first actor can efficiently navigate with the help of the application and the rest can observe their colleague or friend moving in real time and in real space from a remote location. The 3D representations of the environment must be automatically generated and sent to both parties. Moreover, specific elements of the virtual world (e.g. shop) can be manually optimised by the stakeholders or created in real-time by the first actor and distributed to the

rest. These places may contain geo-bookmarks or hotspots and can provide interaction with other parties (e.g. shop stock) for further engagement. Additionally, this scenario can provide MR representations of previous and current user trajectories and several interactions with POIs. Furthermore, collaboration can be triggered between actors, in case that one needs guidance in an unfamiliar environment and the other acts like a source of information. The Main advantage of this scenario is virtual and ubiquitous presence of anybody, in a place that is visited by somebody else.

VII. CONCLUSION

In this paper, the system architecture and core functionality of the client side of a pervasive application has been described. The points that need to be underlined include the smooth presentation of contextual information and provision of required interaction services to the user. Execution of entertainment services has been investigated, based on user applied personalisation options. An advantage of the proposed system architecture is found in the retrieval of contextual information in order to create groups of users that may collaborate in an ambient environment. In extent, actual collaboration in the real world, based on their proximity and personal preferences occurs. Communication patterns on how users can ubiquitously exchange personal information and immerse in digital environments, evolve from the operation of pervasive entertainment services. Lastly, business opportunities emerge, worthwhile for further investigation.

REFERENCES

- [1] C. Magerkurth, A. D. Cheok, R. L. Mandryk, and T. Nilsen, "Pervasive Games: Bringing computer entertainment back to the real world," *ACM Computers in Entertainment (CIE)*, vol. 3, p. 19, July 2005 2005.
- [2] M. Weiser, "Some computer science issues in ubiquitous computing," *Communications of the ACM*, vol. 36, pp. 75-84, July 1993 1993.
- [3] P. Milgram and F. Kishino, "A Taxonomy of Mixed Reality Visual Displays," *IEICE TRANSACTIONS on Information and Systems*, vol. E77-D, p. 15, 01 December, 1994 1994.
- [4] A. D. Cheok, K. S. Teh, T. H. D. Nguyen, T. C. T. Qui, S. P. Lee, W. Liu, C. C. Li, D. Diaz, and C. Boj, "Social and physical interactive paradigms for mixed-reality entertainment," *Computers in Entertainment (CIE)*, vol. 4, p. 20, April 2006 2006.
- [5] M. Montola, "Exploring the Edge of the Magic Circle. Defining Pervasive Games," in *Proceedings of Digital Experience: Design, Aesthetics, Practice Conference (DAC)* IT University of Copenhagen, 2005, p. 4.
- [6] O. Rashid, I. Mullins, P. Coulton, and R. Edwards, "Extending Cyberspace: Location Based Games using Cellular Phones," *ACM Computers in Entertainment (CIE)*, vol. 4, p. 4, January 2006 2006.
- [7] S. Burigat and L. Chittaro, "Location-aware visualization of VRML models in GPS-based mobile guides," in *Proceedings of the 10th international conference on 3D Web technology* Bangor, United Kingdom: ACM Press, 2005, pp. 57-64.
- [8] G. Reitmayr and D. Schmalstieg, "Collaborative Augmented Reality for Outdoor Navigation and Information Browsing," in *Proceedings of the Symposium of Location Based Services and TeleCartography* Vienna, Austria, 2004, pp. 31-41.
- [9] F. Liarokapis, V. Brujic-Okretic, and S. Papakonstantinou, "Exploring Urban Environments Using Virtual and Augmented Reality," *Journal of Virtual Reality and Broadcasting (JVBR)*, vol. 3, pp. 1-13, February 6th, 2007 2007.
- [10] E. Nieuwdorp, "The Pervasive Discourse: An Analysis," *Computers in Entertainment (CIE)*, vol. 5, p. 13, August 2007 2007.
- [11] M. Eyles and R. Eglin, "Entering an age of playfulness where persistent, pervasive ambient games create moods and modify behaviour," in *Third International Conference on Games Research and Development 2007 (CyberGames 2007)* Manchester Metropolitan University, U.K., 2007.
- [12] C. A. Lindley, "Trans-reality gaming," in *Proceedings of the 2nd Annual International Workshop in Computer Game Design and Technology* John Moores University, Liverpool, UK, 2004, p. 10.
- [13] B. K. Walther, "Reflections on the methodology of pervasive gaming," in *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology* Valencia, Spain: ACM, 2005, pp. 176 - 179.
- [14] S. Hinske, M. Lampe, C. Magerkurth, and C. Roecker, "Classifying Pervasive Games: On Pervasive Computing and Mixed Reality," in *Concepts and technologies for Pervasive Games - A Reader for Pervasive Gaming Research*. vol. 1, C. Magerkurth and C. Roecker, Eds. Aachen, Germany: Shaker Verlag, 2007, p. 20.
- [15] M. Capra, M. Radenkovic, S. Benford, L. Oppermann, A. Drozd, and M. Flintham, "The multimedia challenges raised by pervasive games," in *Proceedings of the 13th annual ACM international conference on Multimedia* Hilton, Singapore: ACM, 2005, pp. 89-95.
- [16] W. Broll, J. Ohlenburg, I. Lindt, I. Herbst, and A.-K. Braun, "Meeting technology challenges of pervasive augmented reality games," in *Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games* Singapore: ACM, 2006, p. 28.
- [17] S. Benford, W. Seager, M. Flintham, R. Anastasi, D. Rowland, J. Humble, D. Stanton, J. Bowers, N. Tandavanitj, M. Adams, J. R. Farr, A. Oldroyd, and J. Sutton, "The Error of Our Ways: The Experience of Self-Reported Position in a Location-Based Game," in *Proceedings of the 6th International Conference on Ubiquitous Computing (UbiComp)*. vol. 3205/2004 Nottingham, UK: Springer Berlin / Heidelberg, 2004, pp. 70-87.
- [18] S. Benford, M. Capra, M. Flintham, A. Crabtree, A. Drozd, L. Oppermann, U. PankokeBabatz, A. Waern, L. Ermi, A. Jäppinen, and M. Montola, "A review of design and evaluation methods for IPerG," European Project (FP6 - 004457) Deliverable 13 April 2005 2005.
- [19] D. Saha and A. Mukherjee, "Pervasive Computing: A Paradigm for the 21st Century," *IEEE Computer*, vol. 36, pp. 25-31, March 2003 2003.
- [20] G. K. Mostefaoui, J. Pasquier-Rocha, and P. Brezillon, "Context-Aware Computing: A Guide for the Pervasive Computing Community," in *Proceedings of the The IEEE/ACS International Conference on Pervasive Services (ICPS'04) - Volume 00* American University of Beirut (AUB), Lebanon: IEEE Computer Society, 2004, pp. 39-48.
- [21] R. L. Klatzky, "Allocentric and egocentric spatial representations: Definitions, distinctions, and interconnections," in *Spatial Cognition, An Interdisciplinary Approach to Representing and Processing Spatial Knowledge*, F. Christian, H. Christopher, and W. Karl Friedrich, Eds. London, UK: Springer-Verlag, 1998, pp. 1-18.
- [22] C. Roecker, C. Magerkurth, S. Hinske, and M. Lampe, "Designing User-Friendly Interfaces for Pervasive Gaming Applications," in *Concepts and technologies for Pervasive Games - A Reader for Pervasive Gaming Research*. vol. 1, C. Magerkurth and C. Roecker, Eds. achen, Germany: Shaker Verlag, 2007, p. 12.
- [23] G. Reitmayr and T. Drummond, "Going out: robust model-based tracking for outdoor augmented reality," in *Fifth IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '06)* University of California, Santa Barbara, USA 2006, pp. 109-118.