Citation: Papakonstantinou, S. & Brujic-Okretic, V. (2009). Framework for context-aware smartphone applications. The Visual Computer, 25(12), pp. 1121-1132. doi: 10.1007/s00371-009-0391-8

Link to published version: https://doi.org/10.1007/s00371-009-0391-8

Editorial Manager(tm) for The Visual Computer
Manuscript Draft

Manuscript Number:

Title: Framework for Context-Aware Smartphone Applications

Article Type: Special Issue Article

Keywords: context-awareness; virtual and augmented reality; mobile interactions

Corresponding Author: Mr. Stelios Papakonstantinou,

Corresponding Author's Institution: CITY University

First Author: Stelios Papakonstantinou

Order of Authors: Stelios Papakonstantinou; Vesna Brujic-Okretic

Framework for Context-Aware Smartphone Applications

Stelios Papakonstantinou[1] Vesna Brujic-Okretic[2]

Dept. of Information Science

CITY University, London, UK

Tel: +44 20 7040 3914 Fax: +40 20 7040 8584

stelios@soi.city.ac.uk[1], vesna@soi.city.ac.uk[2]

Abstract — This paper presents research and development of a 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 conceptual design and implementation of our location and orientation based algorithm for mobile augmented reality is presented in detail. 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 and urban navigation.

Keywords — context-awareness, virtual and augmented reality, mobile interactions

1 Introduction

Dramatic advances in technology revolutionised the way in which computers, including mobile devices recently, allow the users to acquire and manipulate complex, multifaceted information in real-time and to interact with each other at various levels. The range of appealing applications is increasing rapidly and it spans across urban navigation, sudden events management, cultural heritage information – through to entertainment and peer to peer communication. For example, since the early video games in 1970s (e.g. Pong) technological progress in artificial intelligence engines and computer graphics, also affected the nature of users' interactions, which is particularly reflected in game playing patterns. New sets of technologies did not only manage to simulate games that were traditionally played in the real world, but also enabled new types of games based on the technology itself, which proved equally enjoyable to the participants. 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 products.

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 the consideration of their advantages and disadvantages [5]. Acquisition and management of quantifiable user-related and environmental parameters - that we refer to as *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. 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 expands on the concept and defines various systems with the main goal set to be 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 genders of games [12]. All examined types fit to the conceptual frameworks drawn by Walther [13] and Hinske et al. [14], as well as the concerns regarding the applicability 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 Virtual Reality (VR) interaction engine and a custom-tailored Augmented Reality (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. The AR mode is independent on any markers or fiducial points that exist in the environment, rendering it advantageous over the majority of similar applications. Although this framework can support applications in various domains like way-finding and exploration of unknown places [9], in this paper we are going to focus on the potential it has when operated under the governing rules of pervasive gaming following its mechanics and the competitive or goal-oriented nature.

2 Enabling concepts & technologies

Mark Weiser 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. The developed software platform works on portable devices, based on Windows Mobile 5/6 PocketPC edition with battery capacity close to 1500 mAh. This type can reflect current technological progress and produce a good performance to size ratio, which is an essential element for modern Commercial Off-The-Self (COTS) tools. Compatibility was verified by executing the framework on HP iPaq hw6915 and HTC Touch Diamond. Despite the advantages of such devices there are several limitations, which can affect user experience. The small display size, for example, needs to be efficiently managed in order to accommodate any visualisation and interaction requirements.

Furthermore, state-of-art devices embed context-sensitive sensors, which generate valuable data and offer improved functionalities. The developed system utilises three sensors; a GPS receiver, a digital compass and an imaging sensor. The positioning sensors that have been evaluated are based on older SiRF Star II/III chipsets, as well as on newer integrated solutions, which support Assisted GPS (A-GPS). These implementations offer more stable readings, with higher precision and partial indoor functionality.

A digital compass provides real-time orientation information to the system that may lead to gesture interactions. For the purpose of this research we developed a custom made solution, based on Honeywell HMR3300 magnetic sensor chipset. The hardware sensor produces three value sentences, which are in ASCII format and correspond to yaw, pitch and roll information.

Most commercial devices embed at least one imaging sensor, which is indispensible for the operation of the AR interface, because it generates video streams. Concurrent encoding and compressing video is a demanding process for current mobile processors, which effectively makes this type of context available only for real-time processing. Although the supported quality for static images is between the range of 1.3 and 5.0 MP, for streaming configurations the maximum resolution is up to 2 MP.

Wireless networks infrastructures can form the platform, which enable clients to transparently connect and share context with remote entities. These entities may be other actors sharing similar interests or participating in a group task, or a centralised system that distributes suitable information. Our framework regards the network as a valuable source of context, which provides resources to the context management engine and consequently to the visualisation interfaces. Each networking stack introduces its own boundaries (e.g. operational range) that may restrict the omnipresence of mobile applications. Furthermore, wide-range 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 mask network addresses, thus rendering the distribution of content limited to request/response mode and not proactive.

Real-time location information is 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 Data Mining techniques. In real-time 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.

2.1 Concept of context-awareness in real time

Many researchers who 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 variables can be fused to an engine through *sensed*, *derived* and *explicitly provided* means.

Main challenge for our context management layer is to perceive context, process it and change 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 generate and enhance stored information. Even in this case, ambiguity of context variables is noticed. Orientation of a moving object can be accurately obtained by querying the digital compass. In addition, for a specific span of time, orientation can be measured by calculating former and present GPS location coordinates. Thus, we promote utilisation of redundant context management techniques in order to support retrieval from secondary available sources as a fail-safe mechanism.

Common ground that most mobile context-aware applications share is found in the use of location context. Accuracy of sensor measurements is increasingly evolving and can prove satisfactory not only for sensed context (e.g. location) but for derived (e.g. velocity) as well. Mobility of devices in the environment is the main reason that triggers exploitation of such information. Proximity of Points of Interest (POI) and of other participating entities (i.e. users, virtual and physical objects) can cause potential collaboration and interaction with the user.

3 Framework architecture

Pervasive applications significantly affect how computing devices are deployed and how people interact with the resulting interface paradigms, because one of the main goals is considered to be ubiquitous information access. 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.

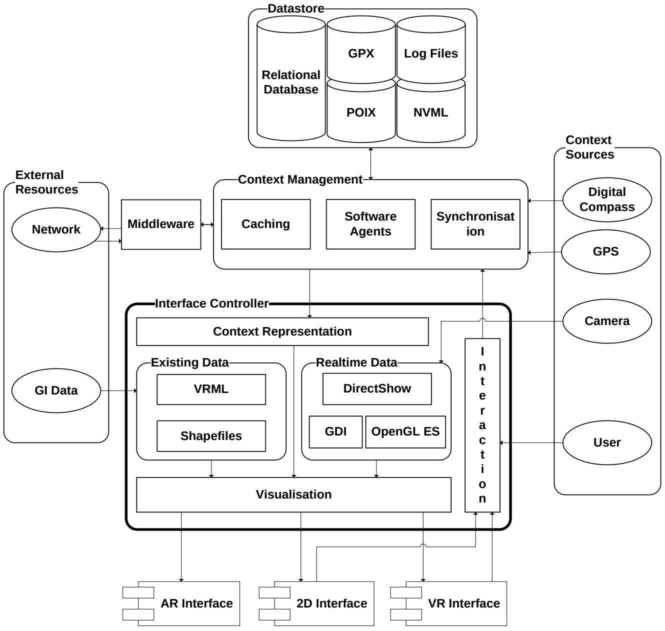

Fig. 1. Prototype Client Architecture

In order to satisfy the requirement of invisibility [19], 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 local 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 keep any specific information about the remote entities or how to contact them.

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) as indicated in Figure 1.

3.1 Context Management System (CMS)

The CMS unit is a low-level subsystem which receives input from the peripheral resources and is responsible for keeping the location 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 responsible for implementing all geometric 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 a game is accommodated by this structure. During configuration and debugging, the CMS generated and simulated artificial movement of remote objects. In this subsystem algorithms, which enhance the accuracy, performance, prediction and interpretation requirements of the application are being executed.

3.2 Information Presentation System (IPS)

Next in sequence is the IPS unit, 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 their virtual representations. These can be interrogated through all user interfaces and can vary depending on the type of interrogated entity (e.g. avatar for human in 3D or icon in 2D). For 2D and VR interface the environment is modelled out of existing data. These are 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.

3.3 Modes of operation

Except the core system, which includes the CMS and IPS subsystems, the user is considered as an 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 significance of these data sources makes them, essentially, independent subsystems.

Fundamentally, interaction in the 2D and 3D environments is supported by two operational modes: the *sensor-controlled* and the *user-configurable* mode. This way the system has the potential 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 simulating real-time interactions, whereas the manual 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. 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.

4 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 interface 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 takes 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 elements. 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.

4.1 2D Interface

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. Each actor can create own POIs by clicking on any part of the map and by providing relevant metadata. These are appended in new or existing POIX files and can be used as game locations, if shared with other users.

4.2 3D Interface

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 (i.e. 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.

Fig. 2. a) Oblique and 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 on figure 3. It was found that information overload and occlusion could be effectively minimised by implementing all views. Interaction is either driven by the sensors or manually, by the user. The sensor mode verifies adjustments 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 select the preferred viewpoint to 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 ($\geq$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.

4.3 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.

For ubiquitous service operation of such technology, a new approach is required. Functional goal is the estimation of the pose, of the device, in accordance to a point in the real world. In addition, synchronisation of all sensory input, as well as real-time retrieval of data generated by local and remote entities need to be accurately managed. Similar to the other interfaces, the location and orientation parameters of the device constitute the essential information required by 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 local spatial and remote spatiotemporal context, the Augmented Reality interface that was developed during the course of this project is capable of amalgamating the scene and selected surroundings with spatially referenced context descriptions by superimposing digital information on precise locations on the device display. The specified implementation can be easily ported on a variety of mobile devices, as this interface is decoupled from the functionality and the only hardware requirement is the physical existence of the sensors. To prove this, we implemented two versions. The first was embedded in a standalone application (Aura) and the second was a reusable ActiveX component (ARIE), which can be integrated in diverse host applications (LOCUS, EPSRC Project). Both solutions enable the incorporation of graphical information with the media streaming layer. The conceptual analysis of the proposed series of algorithms is presented in section 5.

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 subsequent objectives, directional aids are applied. These include 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. Self-explanatory symbols and textual descriptions allow users to comprehend and decide dependant upon their preferences. In any pervasive scenario, directed guidance is provided towards the next goal of the scenario. This may be either locating an object or person, reaching a checkpoint or unlocking the next level of interaction and visualisation functionalities.

5 Coordinate transformations: Algorithm and implementation

The process that we describe in this section was found to be competent in terms of performance, resource-efficiency and reliability for representing features of interest, of the real environment, in an estimated location relative to the device’s actual orientation and position. Although, calculating the pose in the real world by examining a single point in space is not efficient for advanced AR operation, like that found in [23], it was proven to be adequate for the objectives of this research. The reason is that camera resolution, environmental conditions and quality of natural features or markers, do not directly affect the efficiency of the interface. This makes our solution robust and ready to work in unknown environments. Ultimately, we are going to examine pose tracking based on a sum of points. The next figure shows how location context of remote objects is utilised by the AR interface.

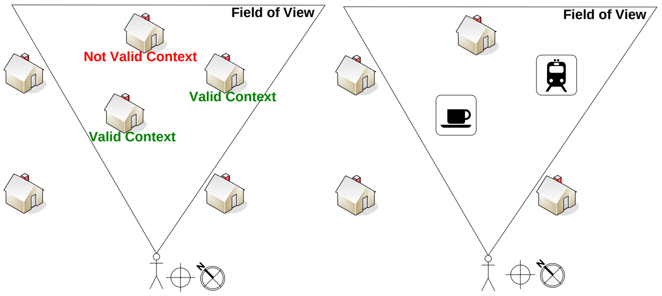

Fig. 3. The use of AR: contextualizing location & orientation information

5.1 Sensing the context

For mobile mixed reality systems, a preferred alternative to tracking the observer’s viewpoint is the use of sensors which can measure the observer's position and orientation. Position can be calculated by using a GPS device, which can report coordinates in three-dimensions to an acceptable level of accuracy, anywhere on earth provided that four or more satellites are within the line of sight of the receiver. Real time orientation information became increasingly available as a result of rapid advanced in micro-electro mechanical systems (MEMS) technology. Digital magnetic compasses can report yaw angles to within 1 degree or better, and accelerometers can measure pitch and roll to similar levels of accuracy. With nominal positional and directional accuracy, digital content can be overlaid, in the real world scene, when appropriate. On the other hand, with almost absolute location accuracy, synthetic content is superimposed exactly over the physical object (i.e. door, person), on the device screen. There are cases that the user may not want to point the camera directly over the target, but have the ability to retrieve only the contextual information about the entity. In such instance only yaw is calculated from the compass and similarly altitude is discarded from GPS readings.

5.2 Calculating the Field-of-View Polygon

The device position and current viewpoint enable calculation of all polygon elements, which simulate the actor's Field of View (FOV). After applying certain trigonometric algorithms, we can retrieve the coordinates of points *B* and *C* in real time, as seen in the following figure. The mobile device position is represented by point *A* and its direction by vector *AD*. In the current implementation, the size of *AB* and *AC* is equal to a predefined value (50m), which can be easily altered from the UI. Changing this constant will provide greater accuracy when reduced and least when maximised. The difference, in terms of functionality, is that we can query entities lying at variable distance. The use of calculus in the Cartesian coordinate system provided the solution for calculating the geographic coordinates of the two unknown points (*B* & *C*), which are needed to accumulate all parameters of the polygon.

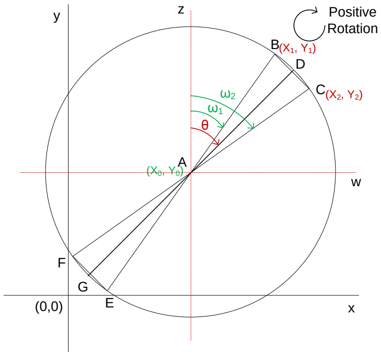

Fig. 4. Line & circle intersection for calculating the FOV

The problem that we face is defined as follows.

- Direction of *AD*, the line of sight, or angle θ is known by retrieving orientation context from the digital compass.
- The coordinates of a point $A(x_0, y_0)$, in the world coordinate system is known from the GPS receiver.
- The length of *AB*, *AC*, *AD* and *r* are equal to a manageable constant (i.e. 50m)
- The angles of *BAD* and *DAC* are equal to a predefined constant (i.e. camera focal length/2)

To calculate the field of view polygon, the required outputs are the geographic coordinates of points $B(x_1, y_1)$, $C(x_2, y_2)$ and $D(x_3, y_3)$. To find the intersection between lines *AB*, *AC* and *AD* the following process has to work out.

We define the coefficient of direction λ of a given line *AD*, as the tangent of angle θ ($\lambda = tan\theta$). Intercept *c*, is the point of the line, which intersects with the *y-axis* and can be found by solving the equation of a line:

$$c = y_0 - \lambda x_0 \qquad \text{(Eq. 1)}$$

Similarly, the equation of a circle with a known centre $A(x_0, y_0)$, an overlapping point $D(x, y)$ and a given radius *r* is given by the equation:

$$(x - x_0)^2 + (y - y_0)^2 = r^2 \qquad \text{(Eq. 2)}$$

The substitution of *y*, in equation 2, with $\lambda x+c$ from equation 1, after expanding and grouping the terms leads to:

$$(1 + \lambda^2)x^2 + (2\lambda c - 2\lambda y_0 - 2x_0)x + (x_0 + y_0 + c^2 - 2cy_0 - r^2) = 0 \qquad \text{(Eq. 3)}$$

Now, if we set

- $a = 1 + \lambda^2$
- $b = 2\lambda c - 2\lambda y_0 - 2x_0$
- $d = x_0 + y_0 + c^2 - 2cy_0 - r^2$

we observe that equation 3 is a quadratic formula conforming to the following format.

$$ax^2 + bx + d = 0 \qquad \text{(Eq. 4)}$$

As a result, we get the solution for obtaining *x*.

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \qquad \text{(Eq. 5)}$$

Similarly, there are two results for *y*, when *x* is substituted with the results of equation 5. This process has to be implemented three times in order to get the coordinates of points *B*, *C*, and *D*. This is achieved by replacing θ with ω_1 and ω_2, which are equal to $\pm \frac{\text{focal length}}{2}$.

5.3 Verify candidates for processing

Next step is to calculate if a point, which represents the coordinates of a remote entity, lies inside the boundaries of a given polygon, with a Point-in-Polygon or Polygon-in-

Polygon theorem. For the purpose of this research we developed a custom algorithm, which implemented the Crossings count method. The simplest way to achieve that was by using the *Jordan Curve Theorem*. This technique evaluates that a point is inside a polygon by sending a ray, starting from the origins of the point, towards infinity. If the number of times, that the ray crosses the polygon edges, is an odd integer, then the point is inside the boundaries. If the number is even, the point is considered to be outside the polygon. There are some cases, in more complex polygons, which this algorithm has proven not to be appropriate. In contrast for a simple pyramid-like polygon, applied in our case, calculation has proven quite efficient.

5.4 Camera modelling

The framework requires the internal parameters of the camera lens because we need to get the screen coordinates of a point that exists in the real world, in relation to the actual orientation and position of the camera that captures it. Accurate modelling of these parameters is accomplished by non-linear models, whereas linear transformations do not model the distortion of the lens. Thus, non-linear approaches are far more accurate and applied on systems with the requirement of maximum precision [24]. The simplest approach is considered to be Hall's [25] but, generally, all linear transformations, including Tsai's [26] are following the same 4-phase procedure.

Inputs for the first step are the coordinates (*x, y and z*) of a POI from the real world. The transformation from the world coordinate system, of the given point, to the 3D camera coordinate system is accomplished by executing the following equation. The transformation is modelled by utilising the rotation matrix $\mathrm{R}_{\mathrm{W}}^{\mathrm{C}}$ and the translation vector $\mathrm{T}_{\mathrm{W}}^{\mathrm{C}}$.

$$\begin{pmatrix} \mathrm{X}_{\mathrm{W}}^{\mathrm{C}} \\ \mathrm{Y}_{\mathrm{W}}^{\mathrm{C}} \\ \mathrm{Z}_{\mathrm{W}}^{\mathrm{C}} \end{pmatrix} = \mathrm{T}_{\mathrm{W}}^{\mathrm{C}} + \mathrm{R}_{\mathrm{W}}^{\mathrm{C}} \begin{pmatrix} \mathrm{X}_{\mathrm{W}}^{\mathrm{W}} \\ \mathrm{Y}_{\mathrm{W}}^{\mathrm{W}} \\ \mathrm{Z}_{\mathrm{W}}^{\mathrm{W}} \end{pmatrix} \qquad \text{(Eq. 6)}$$

For the projection of the 3D point on the image plane, we have to consider the optical sensor as a pinhole camera. This means that the image plane is located at a distance *f* from the optical centre O_{C} and is parallel to the plane defined by the coordinate axis X_{C} and Y_{C}. If an object point $\mathrm{P}_{\mathrm{W}}^{\mathrm{C}}$, related to the camera coordinate system, is projected through the focal point O_{C} the optical ray intercepts the image plane at the 2D image point $\mathrm{P}_{\mathrm{U}}^{\mathrm{C}}$. This is presented in the following equations. Note that commercial camera manufacturers provide the focal length with the product specifications, which is not the case for mobile phone cameras. This means that it has to be calculated by using additional software , which examines pictures taken by the camera.

$$X_U^C = f\,\frac{x_W^C}{z_W^C} \qquad \text{and} \qquad X_U^C = f\,\frac{x_W^C}{z_W^C} \qquad \text{(Eq. 7)}$$

The third step is used for modelling the distortion of the lens. Basic reason that causes radial distortion is the potential flaws found on the curve of the camera lens. In the following equation we can see the transformation between the undistorted point $\mathrm{P}_{\mathrm{U}}^{\mathrm{C}}$ and the distorted point $\mathrm{P}_{\mathrm{D}}^{\mathrm{C}}$, where δ_x and δ_y represent the involved distortion.

$$X_U^C = X_D^C + \delta_x \qquad \text{and} \qquad Y_U^C = Y_D^C + \delta_Y \qquad \text{(Eq. 8)}$$

Although there are two kinds of potential distortion factors, radial and tangential, Tsai [26] has noted that only radial has to be taken under consideration, because modelling both may produce numerical instabilities during the calculations. The displacement given by radial distortion δ_r can be modelled in the following equation, which considers only k_1.

$$\delta_{xr} = k_1 X_D^C\,({}^{C}X_D^2 + {}^{C}Y_D^2) \quad \text{and} \quad \delta_{yr} = k_1 Y_D^C\,(X_D^{\mathrm{C}2} + Y_D^{\mathrm{C}2}) \quad \text{(Eq. 9)}$$

The last step deals with the change from the camera image to the screen image coordinate system. This is accomplished by conveying point $\mathrm{P}_{\mathrm{D}}^{\mathrm{C}}$ with respect to the screen image plane, which is constituted by pixels. The next two equations explain how to accomplish this transformation.

$$X_D^I = -k_u X_D^C + u_0 \qquad \text{and} \qquad Y_D^I = -k_v X_D^C + v_0 \qquad \text{(Eq. 10)}$$

The parameters k_u and k_v make the transformation from metric measures, in the camera coordinate system, to pixel, in the screen image coordinate system. Parameters u_0 and v_0 are those that define the projection of the focal point in the plane image in pixels. The translation between the two coordinates systems depend on their value.

5.5 Scene rendering

After finding the screen coordinates of a real world point we need to superimpose a distinct visual effect, on top of the video stream, in relation to that point. Video streaming was accomplished by using the DirectShow API. There is limited support for the combination of filters in DirectShow for mobile devices, in comparison to desktop implementations. This instructed for custom low-level filter development. The series of filters that are interconnected in order to perform a specific task is called a *Filter Graph*. The implemented Filter Graph is constituted out of three filters. The *Video Capture* filter initialises the camera drivers. The development platform (i.e. HTC Touch Diamond) produces frames with RGB565 pixel format and size of 240x320 pixels. Following next, a custom *Sample Grabber* filter is used to retrieve all frames that pass through the Filter Graph, so that any additional image analysis can take place. When done, the frame continues to the final filter, the *Renderer*. This filter is particularly interesting because it renders original frames, as well as any context descriptions that we want to overlay.

Initially, only textual descriptions have been used, which can uniquely identify any available remote instances. GDI functions have been utilised in order to represent text, for the desired amount of time and on a predefined place on screen. But for more complex representations like 2D symbols and 3D elements a new approach is required. As a result, we are developing new versions of the framework, which work on top of OpenGL ES. After initialisation, the Renderer filter creates a new window by using EGL bindings. Its dimensions are similar to the video stream frames. At this point the viewer perspective, in OpenGL, is manipulated, to match the parameters that were received during camera calibration. Following next, a new 2D rectangle is formed and its face is altered by applying a texture. The contents of this texture are retrieved from the video buffer. When a new frame is received from the previous filter, the scene is refreshed to present any changes. By utilising this solution we are able to embed 2D and 3D elements in relation to the natural elements of the video.

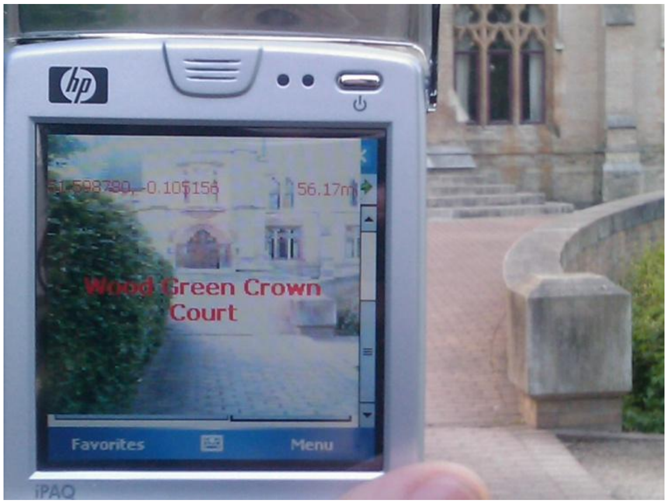

Fig. 5. Text representations on the AR interface, working on Internet Explorer (ARIE)

6 Smartphone application considerations

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. In comparison to independent modes of play, social interactions can intensify user engagement with the system.

Based on this reasoning, a specific use case is proposed and presented, which is currently under development. The scenario presents specific requirements that have been acquired from the potential user and the proposed functionality which can satisfy them. This use case needs a set of novel interfaces for nominal operation and specifically it utilises VR as a base but extends to AR, as appropriate. All proposed communications should conform to explicit personalisation and privacy rules.

The scenario focuses on virtual surveillance and exploration rationale and can take place in an urban environment. Optimally it should be applied in a context-rich area with available objects of interest (i.e. shopping centre or marketplace). The participating entities include one or more actors. The main user can efficiently navigate with the help of the application and the rest can observe their partner or friend move in real time and in real space, in a simulated 3D world available to both parties. The 3D representation of the environment has to be automatically generated by a centralized system and transmitted to all parties utilising the available network infrastructure. Additionally, specific parts of the virtual world (e.g. a shop) could be manually optimised if the relevant stakeholder invested in doing this. These places are considered as geo-bookmarks or hotspots and may offer further interactions, like querying the shop stock. Furthermore, the proposed functionality offers mixed reality representations of previous and current paths, that have been explored, and interactions with the available POIs. Collaboration can be triggered between users, if one of them needs guidance on how to move in an unfamiliar environment. In such case the other user becomes the source of information, who offers in–context advices. Main advantage of this scenario is virtual and ubiquitous presence of anybody in a place that is visited by a familiar person.

Generally, it was found very helpful to combine the real scene with overlaid additional information about the environment for the decision making process. By observing the surroundings from an egocentric perspective the user can make better informed choices if presented with contextual information related to the objects in the scene. For example, a user is located in an unknown environment and seeks a train station. Two relevant candidates appear on the screen. After panning the smart-phone over both options the user can decide which one to choose, based upon their characteristics – e.g. proximity. In addition, if there is a relevant web link stored in the entry about the landmark object, the user could visit the URL directly (through the 3G connection), to gather additional information such as which underground lines are served by this particular station. Further functionalities included in this scenario are: the position of other objects or subjects in the scene (*Position* queries), the distance from the chosen type of objects of interest (*Nearest* POIs), the *Range* queries giving the information on how far to travel to the nearest object of interest. Particularly useful is the ability to display the distance of an object relative to the device and the coordinates of the object of interest in the world frame of reference. This information opens up the whole range of new types of interactions with the virtual content leading to applications in other fields such as – real estate industry, or risk management in the emergency situation context.

7 Conclusion

This paper describes the concept and selected considerations for a technical framework to enable mobile interactions in various contexts. Dedicated system architecture is designed and implemented to support the entire framework and it is presented in this paper. The framework offers smooth displaying of contextual information and the provision of necessary interaction services to the user, depending on the requirements of a particular custom application in hand. Social interactions are encouraged by applying appropriate communication patterns enabling ubiquitous exchange of personal information and allowing the user to immerse in digital environments. In the context of pervasive entertainment services, social interactions, coupled with user-applied personalisation options, can produce satisfactory results, which may upgrade the current commercial business models to a new level that focuses on satisfying wider audiences and more complicated user needs. The scenario described in the previous chapter, amongst others, shows just a fraction of potential business opportunities that emerge, worthy of further investigation.

From the technical perspective, the use of this system architecture offers certain advantages. It can take the form of a COTS product, which may be widely available to commercial end-users and provide advanced services, similar, but not limited to those found in current Location Based Services. The main contribution of the proposed system is that it can operate in unknown environments. Further advantages of this system architecture include the retrieval of contextual information in order to create groups of users that may collaborate in an ambient environment. In other words, actual collaboration between actors in the real world, based on their proximity and personal preferences may occur, making it unique and giving a great deal of potential in terms of the context in which these functionalities can be applied and re-used. Our Augmented Reality algorithm, using position and orientation information to retrieve and display appropriate information, provides the foundation for further advances and versatility in the use of the system. Current progress in context-awareness and sensor technology enables the current techniques to move to higher levels, more readily accessible to the user.

8 References

1. Magerkurth, C., Cheok, A.D., Mandryk, R.L., Nilsen, T.: Pervasive Games: Bringing computer entertainment back to the real world. ACM Computers in Entertainment (CIE) **3** (2005) 19
2. Weiser, M.: Some computer science issues in ubiquitous computing. Communications of ACM **36** (1993) 75-84
3. Milgram, P., Kishino, F.: A Taxonomy of Mixed Reality Visual Displays. IEICE TRANSACTIONS on Information and Systems **E77-D** (1994) 15
4. Cheok, A.D., Teh, K.S., Nguyen, T.H.D., Qui, T.C.T., Lee, S.P., Liu, W., Li, C.C., Diaz, D., Boj, C.: Social and physical interactive paradigms for mixed-reality entertainment. Computers in Entertainment (CIE) **4** (2006) 20
5. Montola, M.: Exploring the Edge of the Magic Circle. Defining Pervasive Games. Proceedings of Digital Experience: Design, Aesthetics, Practice Conference (DAC), IT University of Copenhagen (2005) 4
6. Rashid, O., Mullins, I., Coulton, P., Edwards, R.: Extending Cyberspace: Location Based Games using Cellular Phones. ACM Computers in Entertainment (CIE) **4** (2006) 4
7. Burigat, S., Chittaro, L.: Location-aware visualization of VRML models in GPS-based mobile guides. Proceedings of the 10th international conference on 3D Web technology. ACM Press, Bangor, U.K. (2005) 57-64
8. Reitmayr, G., Schmalstieg, D.: Collaborative Augmented Reality for Outdoor Navigation and Information Browsing. Proceedings of the Symposium of Location Based Services and TeleCartography, Vienna, Austria (2004) 31-41
9. Liarokapis, F., Brujic-Okretic, V., Papakonstantinou, S.: Exploring Urban Environments Using Virtual and Augmented Reality. GRAPP 2006 Special Issue **3** (2007) 1-13
10. Nieuwdorp, E.: The Pervasive Discourse: An Analysis. Computers in Entertainment (CIE) **5** (2007) 13
11. Eyles, M., Eglin, R.: Entering an age of playfulness where persistent, pervasive ambient games create moods and modify behaviour. Third International Conference on Games Research and Development 2007 (CyberGames 2007), Manchester Metropolitan University, U.K. (2007)
12. Lindley, C.A.: Trans-reality gaming. Proceedings of the 2nd Annual International Workshop in Computer Game Design and Technology, John Moores University, Liverpool, UK (2004) 10
13. Walther, B.K.: Reflections on the methodology of pervasive gaming. Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology. ACM, Valencia, Spain (2005) 176 - 179
14. Hinske, S., Lampe, M., Magerkurth, C., Roecker, C.: Classifying Pervasive Games: On Pervasive Computing

and Mixed Reality. In: Magerkurth, C., Roecker, C. (eds.): Concepts and technologies for Pervasive Games - A Reader for Pervasive Gaming Research, Vol. 1. Shaker Verlag, Aachen, Germany (2007) 20

15. Capra, M., Radenkovic, M., Benford, S., Oppermann, L., Drozd, A., Flintham, M.: The multimedia challenges raised by pervasive games. Proceedings of the 13th annual ACM international conference on Multimedia. ACM, Hilton, Singapore (2005) 89-95
16. Broll, W., Ohlenburg, J., Lindt, I., Herbst, I., Braun, A.-K.: Meeting technology challenges of pervasive augmented reality games. Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games. ACM, Singapore (2006) 28
17. Benford, S., Seager, W., Flintham, M., Anastasi, R., Rowland, D., Humble, J., Stanton, D., Bowers, J., Tandavanitj, N., Adams, M., Farr, J.R., Oldroyd, A., Sutton, J.: The Error of Our Ways: The Experience of Self-Reported Position in a Location-Based Game. Proceedings of the 6th International Conference on Ubiquitous Computing (UbiComp), Vol. 3205/2004. Springer, Nottingham, UK (2004) 70-87
18. Benford, S., Capra, M., Flintham, M., Crabtree, A., Drozd, A., Oppermann, L., PankokeBabatz, U., Waern, A., Ermi, L., Jäppinen, A., Montola, M.: A review of design and evaluation methods for IPerG. In: Benford, S., Capra, M. (eds.): WP5 Design and Evaluation (2005) 54
19. Saha, D., Mukherjee, A.: Pervasive Computing: A Paradigm for the 21st Century. IEEE Computer **36** (2003)
20. Mostefaoui, G.K., Pasquier-Rocha, J., Brezillon, P.: Context-Aware Computing: A Guide for the Pervasive Computing Community. The IEEE/ACS International Conference on Pervasive Services (ICPS'04) - Volume 00. IEEE Computer Society, American University of Beirut (AUB), Lebanon (2004) 39-48
21. Klatzky, R.L.: Allocentric and egocentric spatial representations: Definitions, distinctions, and interconnections. In: Christian, F., Christopher, H., Karl Friedrich, W. (eds.): Spatial Cognition, An Interdisciplinary Approach to Representing and Processing Spatial Knowledge. Springer-Verlag, London, UK (1998) 1-18
22. Roecker, C., Magerkurth, C., Hinske, S., Lampe, M.: Designing User-Friendly Interfaces for Pervasive Gaming Applications. In: Magerkurth, C., Roecker, C. (eds.): Concepts and technologies for Pervasive Games - A Reader for Pervasive Gaming Research, Vol. 1. Shaker Verlag, achen, Germany (2007) 12
23. Reitmayr, G., Drummond, T.: Going out: robust model-based tracking for outdoor augmented reality. Fifth IEEE/ACM International Symposium on Mixed and Augmented Reality (ISMAR '06), University of California, Santa Barbara, USA (2006) 109-118
24. Salvi, J., Armangué, X., Batlle, J.: A comparative review of camera calibrating methods with accuracy evaluation. Pattern Recognition **35** (2002) 1617-1635
25. Hall, E.L., Tio, J.B.K., McPherson, C.A., Sadjadi, F.A.: Measuring Curved Surfaces for Robot Vision. Computer, Vol. 15. IEEE Computer Society Press (1982) 42-54
26. Tsai, R.Y.: A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. Journal of Robotics and Automation **3** (1987) 323- 344
27. Vezhnevets, V., Velizhev, A.: GML C++ Camera Calibration Toolbox. http://research.graphicon.ru/calibration/gml-c++-camera-calibration-toolbox.html (2005)

Author Biographies (50 - 100 words)

Stelios Papakonstantinou is a computer scientist and research student at the Dept. of Information Science, of City University London. He obtained a BSc (Hons) in Computing & Networks from University of Hertfordshire and an MSc (Hons) from City University. His research interests include mobile context-aware computing, GIS, HCI and MR techniques for visualisation.

Vesna Brujic-Okretic is the Head of the Information Science Department at City University London. Her most recent experience and expertise are in the area of Augmented Reality and the integration of visual with sensory information for navigation purposes. She is also interested in researching into visual information retrieval and into data visualisation methodologies. She is a member of the Geographic Information Science Group, currently leading the LOCUS project, as co-investigator. The project has been awarded by EPSRC, through Pinpoint Faraday Partnership, as their flagship project, to research into the field of intelligent interfaces on mobile platforms, using Global Navigational Satellite Systems (GNSS).

Author Photos (black-and-white passport size)

Click here to download high resolution image

Author Photos (black-and-white passport size)

Click here to download high resolution image