

Filtering Location-Based Information Using Visibility

Ashweeni Beeharee and Anthony Steed

Department of Computer Science, University College London, Gower Street, London
WC1E 6BT, United Kingdom
{A.Beeharee, A.Steed}@cs.ucl.ac.uk

Abstract. In this paper we present an approach for exploiting knowledge about features in the real world in order to compute visibility of buildings. This is performed with the awareness of the inconsistencies and lack of accuracy in both mapping technology and GPS positioning in urban spaces. Electronic tourist guide systems typically recommend locations and sometimes provide navigation information. We have augmented this system to exploit visibility knowledge about neighbouring physical features

1 Introduction

One reason that mobile, ubiquitous and mixed-reality systems are attracting a lot of attention recently is that they can present information about the real world that surrounds the system and its users. Along with increased power of portable systems, and accessible tracking technology such as GPS, this has enabled a new class of highly interactive ‘guiding’ that can give information and instructions to the user as they move through their environment [10]. This is a very broad class of application, but typical applications include presenting route knowledge from road databases [4] or location-specific information such as web pages about buildings and spaces. There are many configurations of such systems depending on the technology available from WAP-based phone, through to wearable personal computers. They rely mostly on retrieving some form of location data about the user, either by explicit input or passive tracking, and matching this against a database of knowledge about the world.

In this paper we start by claiming that such systems can suffer from two problems. The first is that the forms of information that are presented are often quite abstract, and thus hard to relate to the real world. For example, route knowledge might be presented as a series of instructions such as ‘Turn right after 100m’, that can be difficult to interpret accurately. This is because those instructions require judgements that users might find it difficult to make such as relative distance. The second problem is that the information often ignores critical parts of the user’s perception of their own context: what they can or can’t see from their current location.

We present a guiding application for urban environments that exploits knowledge about the physical structure of the real world. The guiding application has

fairly standard functionality for a guiding application: as the user moves around the real world, their location is plotted on a map and icons which link to geo-tagged and context sensitive information are presented to the user. The location is either sensed by a GPS receiver, or explicitly input if a GPS unit is not available or a usable GPS signal is not available. As they move, a location-based recommendation system presents a series of media resources and links that might be relevant. Primary amongst these is photographs of the real world.

Our contributions are in exploiting knowledge about the real world. Specifically we exploit building geometry from a geographic information system (GIS). We use this knowledge in two ways. Firstly, information can be attached to the buildings themselves rather than simply to coordinates in space. This could be done explicitly, but our first technical contribution is in presenting a method for taking photographs and associating them to the buildings that are actually in the photograph. Secondly, when retrieving information we filter it based on whether it is likely to be visible from the current location. Our contribution here is in showing how this can be implemented even if the user's location is known only imprecisely.

By using building geometry as the primary means to index information we get around some of the problems of the imprecision and inflexibility of just associating information to positions in space. By retrieving information based on an estimate of visibility, we can present data that the user can more easily relate to their actual experience.

In this paper we will first discuss some related work in more detail. We then present an overview of the system encompassing the user interface devices (a laptop or PDA) and the back-end database services. In the following three sections we then present the implementation: user interface for editing recommendations, geo-indexed photograph database and process for generating and filtering recommendations. Finally we discuss the deployment of the system and some preliminary experiences with the system.

2 Related Work

There are many examples of electronic consumer guiding systems available. For example, see [3] for a review of the capabilities and issues of vehicle GPS navigation systems, and [12,14] for example, location-based PDA guide systems. In-car systems often utilise a GPS system to give location information. Indeed moving vehicles are a situation where GPS performs well: some consumer GPS units exploit velocity information and models of vehicle dynamics for improving tracking accuracy. Also most roads are uncovered and it is often possible to get a good view of the sky because roads are built away from buildings. On a vehicle, a good GPS aerial position can be chosen and the equipment can be powered reliably.

GPS-based systems perform less well in urban environments and in situations where they are carried by pedestrians. There are several reasons including, the occlusion of the sky by buildings and trees, less stable aerial placement, pres-

ence of an occluding body near the device (the user) and less certain movement behaviour of the carrier. Thus mobile city guides use a much wider variety of location technologies, including 802.11 wavelan [15] and explicit input [6,13]. For the purpose of this paper it is worth noting that all these technologies provide different qualities of error. GPS varies in accuracy over time [16], and can often fail to provide a position. However, when it does provide a position, it can be accurate to within a few meters, and this is expected to improve as technical improvements are made to the supporting technology. For phone cell, and other region-based location technologies it can often be difficult to describe the region that is sensed. For example, see [8] for a description of tracking accuracy using a 802.11-based system. For our work, we will be able to use any approximation to the location of the user since we use a sampling approach to visibility analysis

3 System Overview

The George Square system (subsequently referred to simply as GSSystem) provides a guiding application for urban spaces [7]. A user has a PDA or tablet PC running User Viewer software. The user viewer software shows a map of the local area. The map is generated automatically from the MasterMap data [2] using the mapserver software[1]. As the user's movements around the real world are tracked by GPS or as they move an icon representing them on a map, the GSSystem pushes recommendations on to the user interface. A recommendation is a link to a resource such as a web page, or a photograph. The recommendations have a focus that is a polygonal region of space on the map to which they are attached. A single focus can have multiple recommendations attached to it. The system includes the ability to take a photograph and attach it to an existing focus. The focus closest to the user is used.

The GSSystem is novel in that recommendations are not generated purely by proximity to the user; rather they are based on likely contextual relevance based on analysis of previous use of the recommendations [9]. That is, a recommendation will more likely to be made in the past, another user who shared some previous recommendations, then went on to find subsequent recommendations useful. The service that performs this, Recer Service, is a form of collaborative filtering, but note that the symbols over which matches are made are not explicitly defined, rather activity in the system is recorded verbatim as text strings. The GSSystem is also novel in that it is peer to peer in nature. This means that although recer is an independent service, any number of such services can be run. Recer services communicate through a shared data space system, EQUIP [5]. The recer service records behaviour of all user viewer clients that connect to this data space and can make recommendations to all peers on that space. It provides no other services and, in particular, peers automatically set up collaboration between themselves as required.

The original GSSystem thus comprises the elements User Viewer, Shared Data Space, Recer Service and Focus Database from Figure 1. In this paper we

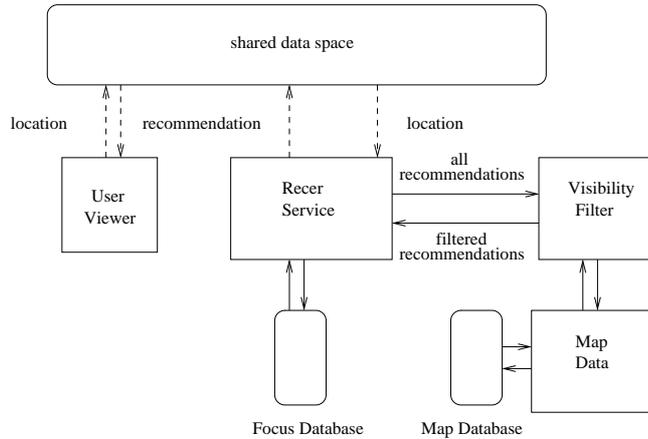


Fig. 1. Overview of the Enhanced George Square System in Run-Time Mode

discuss the Enhanced George Square System (EGSSystem), which also includes the Visibility Filer, Map Data and Map Database elements from Figure 1.

4 Photo Visibility Computation

The main contribution of this paper is in the use of visibility information to filter recommendations. A recommendation is a list of resources. In the original GSSystem, these were purely generated based on prior activity [9] and our enhancement is to also provide a filter based on probability of visibility from current location.

In the original EGS, new photographs were simply attached to the closest focus. We provided a simple editor such that more specific information about the photographs of the buildings can be stored and used to compute visibility of real buildings within those photographs. The images need not be attached to any particular existing building but can exist at any location on the map. By using the editor, the user can also specify the view volume for each photograph by drawing on the map.

In [11] Cohen-Or et al. present an overview of visibility algorithms from a computer graphics research point of view. Although analytic solutions for visibility exist, because our system is probability based, we need to estimate how much of a photograph a building covers. This, and ease of implementation suggests that a simpler, sampling-based approach is more suitable. We have used a raycasting algorithm to compute the visibility of foci boundaries from a photograph's view volume. This algorithm has the advantage of being easy to implement, and the from region visibility we will use later, is a simple variant.

The visibilities are calculated using the following steps. Firstly, a set F is populated with all features (such as buildings) which intersect with the view

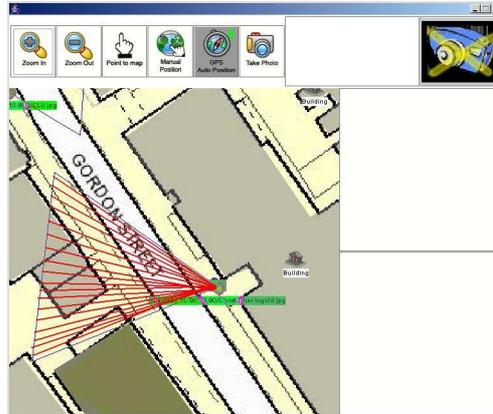


Fig. 2. Example of raycasting approach for associating foci boundaries to photographs. The lines originating from the photograph icon indicate some example rays that are cast from the photograph's location. Note that because of the direction the photograph is taken in, the photograph is not associated to the closest building which is 'behind' the camera

volume of the image. About hundred rays, with the view volume's origin as their initial point, are created according to the expanse of the view volume. These rays are cast for each of the buildings in the set F .

If a feature is hit, a check is performed to ensure that there is nothing between the feature and the viewpoint where the picture was taken, that has been hit by the same ray. This is done by comparing the distances of the intersection points of the features to the view volume's origin. If the feature under inspection is the nearest one to be intersected by the ray, it's counted as a hit. Otherwise it moves on to the next ray. We repeat the same operations until the set F is exhausted.

5 Filtering Recommendations

The second contribution is in taking recommendations and filtering them based on visibility to the user's current location. A recommendation consists of a list of resources, each of which has an associated foci. We use these foci and test if they are visible from the user's location.

5.1 Visibility Filter

Because the user's location is known only inaccurately, we use a from-region visibility algorithm [11]. We assume that some form of probability distribution is given for the user's location. Once again, a sampling-based method is appropriate because it can estimate likelihood of visibility no matter the shape of the location probability distribution.

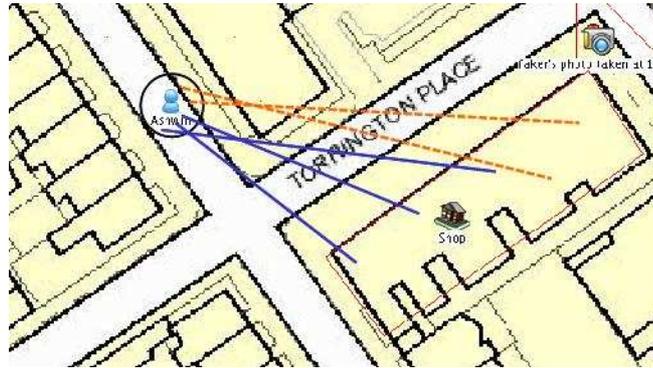


Fig. 3. Visibility computation for the filter - A random point is taken in the region around the location (the circle). Another random point is taken on the target building, and ray is defined between them. The thicker rays hit the target while the thinner rays are obstructed by buildings in their path.

The algorithm used to compute the visibility of the recommendations is as follows. A set P is populated with all polygons to be considered in the visibility computation. A view polygon around the view point is considered. This is representative of the error in the GPS positioning.

Another set of polygons, R , is created corresponding to the recommended features. R need not be, but usually is a subset of P . The following steps are repeated for all polygons in R , referred to as target polygon.

1. From the set P , a new subset is defined as the polygons which lie within a rectangular region subtended by the viewpoint and the centroid of the target polygon. This subset of polygons is the set of potentially relevant polygons.
2. Select n number of random points within the view polygon and the target polygon.
3. For each ray from a point in the view polygon to the target polygon, find intersections with the set of relevant polygons. Search for intersection (for a line) is halted as soon as it intersects ANY polygon. The visibility confidence of a given target polygon is given by $1 - (\text{number of intersection found} / \text{number of rays considered})$.

This process is illustrated in 3, where five rays are shot, but only three hit the building, giving a confidence of 60 using the set of focus polygons found in the recer database or by using the set of closed polygons from the Ordnance Survey map database.

5.2 Presenting Recommendations to the User

Figures 4 show simple example of filtering. a) Un-filtered recommendations b) Filtered recommendations. Note the building at the top of the figure has been re-

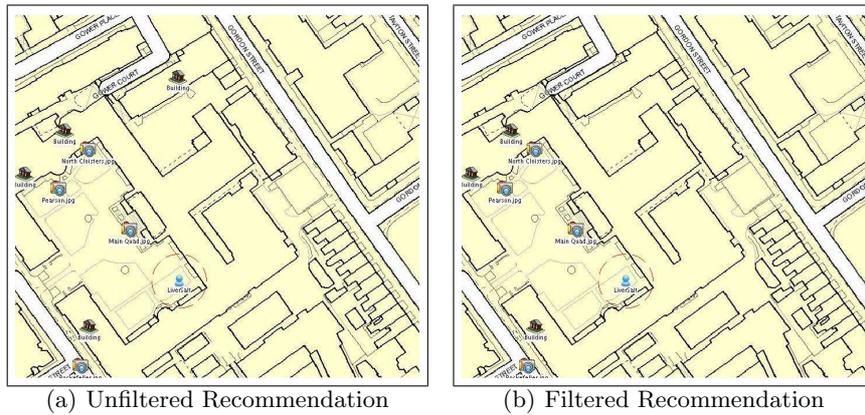


Fig. 4. Maps showing recommendations from Recer Service

moved. Also note that the photograph on the bottom left has not been removed, because the building it is a photograph of is still visible.

5.3 Highlighting Recommendations at Run-Time



Fig. 5. Highlight recommendation to the user

The building which is highlighted upon selection has a tool-tip which says 1) from which photograph it can be seen 2) how much space in the photograph it occupies.

5.4 Small Form Factor Versions

The previous sections have used screenshots from a version of the User Viewer software designed to run on a laptop or tablet PC. For mobile use we also created a version of the user viewer that runs on a PDA.

This small form factor is a challenge to any mapping application, and filtering of recommendations on visibility can serve as an aid to removing clutter on the display. For this version, the recommendation service has been implemented as a web service. Figure 6 in the next section shows examples using the PDA version.

6 Experience on the Streets

The original GSSystem takes its name from the area of its original deployment in Glasgow, Scotland. Some minor technical changes within the EGSSystem have made it deployable anywhere in the UK given availability of the OS MasterMap information. A test system has been deployed in the Bloomsbury area of London, England. Thirty photographs were taken in the local area and published on a website. These were then added to the map and associated to buildings using the EGSSystem. The User Viewer software running on a laptop was used to test the system with GPS tracking on and explicit location input. Finally we tested with the PDA version. For this purpose we used publicly available 802.11b networks, though other forms of networking would be possible. It is possible to run the plain GSSystem standalone, but the EGSSystem requires access to a database.



Fig. 6. Recommending a picture to the user

Figure 6 shows a picture of the PDA version on the street. A photograph has been recommended, and it is obvious from the picture that the building in this photograph is visible from the user's location, though the actual photograph was taken from a very different angle. This is the core functionality that we wish to support in the application: presentation of information filtered based on likelihood of visibility. Note that the EGSSystem can be used in two modes: tracking

by GPS, or explicit input. In the former mode, the photographs are useful for the user to orient themselves to the map and the other recommendations. In the latter mode, it is useful to be able to ‘preview’ an area to see what it will look like when you get there.

Informal observations of use in the street suggest that recommending photographs based on visibility is quite intuitive to understand. One concern was that allowing recommendation of photograph that were taken from locations that are invisible but of buildings that are visible would be confusing for the user. However, we realised that buildings are quite recognisable from different directions, so it would usually be obvious which building was meant. If this proved a problem, it would be possible to extend the system to check which parts of buildings were visible. A second concern was that visibility filtering would mean results would change state frequently. However the sampling approach means that visibility changes fairly slowly. Because a threshold is used, buildings that are only marginally visible are not allowed through the filter.

A fuller user evaluation is currently being designed.

7 Conclusion

We presented a system, the Enhanced George Square System, a mobile guide application that exploits the geometry of the real world. It uses GIS information about buildings as a basis for filtering information based on whether it is likely to be visible to the user in the real world. In this paper we have presented technical contributions on how to implement this filtering based on raycast sampling methods. This method has the advantage of being simple to implement and fast. We have also presented a system and application that exploit this visibility information. These should be seen as early examples of the potential of exploiting visibility information. Furthermore, we would claim that this application is one of the first to start investigating the exploitation of knowledge about the geometry of the world as a primary determinant of user context. There are several avenues for further work. The algorithms we have presented are simple, and we have already indicated that there is a lot of other work from the 3D graphics field that could be incorporated to improve and speed up these algorithms. However, the most fruitful areas for further work are on other methods or paradigms for exploiting knowledge about local context.

Acknowledgements

This work was supported by the UK projects, EQUATOR Interdisciplinary Research Collaboration (EPSRC Grant GR/N15986/01) and Advanced Grid Interfaces for Environmental e-science in the Lab and in the Field (EPSRC Grant GR/R81985/01). The vector data used was supplied by the UK Ordnance Survey.

References

1. Mapserver. Web Resource - verified 15-02, 2005. <http://mapserver.gis.umn.edu>.
2. Uk ordinance survey. Web Resource - verified 15-02, 2005. <http://www.ordnancesurvey.co.uk/oswebsite>.
3. ABBOT, E., AND POWELL, D. Land-vehicle navigation using gps. *Proceedings of the IEEE* 87, 1 (1999), 145–162.
4. AGRAWALA, M., AND STOLTE, C. Rendering effective route maps: Improving usability through generalization. In *SIGGRAPH 2001, Computer Graphics Proceedings* (2001), E. Fiume, Ed., ACM Press / ACM SIGGRAPH, pp. 241–250.
5. BENFORD, S., BOWERS, J., CHANDLER, P., CIOLFI, L., FLINTHAM, M., FRASER, M., GREENHALGH, C., HALL, T., HELLSTROM, S.-O., IZADI, S., RODDEN, T., SCHNADELBACH, H., AND TAYLOR, I. Unearthing virtual history: Using diverse interfaces to reveal hidden virtual worlds. *Ubicomp 2001* (2001), 225–231.
6. BENFORD, S., SEAGAR, W., FLINTHAM, M., ANASTASI, R., ROWLAND, D., HUMBLE, J., STANTON, D., BOWERS, J., TANDAVANITJ, N., ADAMS, M., FARR, J. R., OLDROYD, A., AND SUTTON, J. The error of our ways: The experience of self-reported position in a location-based game. *Ubicomp 2004* (2004).
7. BROWN, B., CHALMERS, M., BELL, M., MACCOLL, I., HALL, M., AND RUDMAN, P. Sharing the square: collaborative visiting in the city streets. *CHI 2005* (2005), under submission.
8. CASTRO, P., CHIU, P., KREMENEK, T., AND MUNTZ, R. A probabilistic location service for wireless network environments. In *UbiComp 2001* (2001), S. Verlag, Ed., pp. 18–24.
9. CHALMERS, M. Cookies aren't enough: Tracking and enriching web activity with recer. *Preferred Placement: Knowledge Politics on the Web* (2000), 99–102.
10. CHEVERST, K., DAVIES, N., MITCHELL, K., FRIDAY, A., AND EFSTRATIOU, C. Developing a context-aware electronic tourist guide: Some issues and experiences. In *CHI 2000* (2000), ACM Press, pp. 17–24.
11. COHEN-OR, D., CHRYSANTHOU, Y., SILVA, C., AND DURANT, F. A survey of visibility for walk-through applications. *IEEE Transactions on Visualization and Computer Graphics* 9, 3 (2003), 412–431.
12. ESPINOZA, F., PERSSON, P., SANDIN, A., NYSTRM, H., CACCIATORE, E., AND BYLUND, M. Social and navigational aspects of location-based information systems. In *UbiComp 2001* (2001), B. S. Abowd, Ed., Springer, pp. 2–17.
13. KRAY, C., AND KORTUEM, G. Interactive positioning based on object visibility. *Mobile HCI 04* (2004), 276–287.
14. LANE, G. Urban tapestries: Wireless networking, public authoring and social knowledge. *Personal and Ubiquitous Computing* (2003), 169–175.
15. SCHILIT, B., LAMARCA, A., BORRIELLO, G., GRISWOLD, W., McDONALD, D., LAZOWSKA, E., BALACHANDRAN, A., HONG, J., AND IVERSON, V. Challenge: Ubiquitous location-aware computing and the place lab initiative. In *The First ACM International Workshop on Wireless Mobile Applications and Services on WLAN (WMASH 2003)* (2003), pp. 29–35.
16. STEED, A. Supporting mobile applications with real-time visualisation of gps availability. *MobileHCI 2004* (2004), 373–377.