ACTA UNIVERSITATIS AGRICULTURAE ET SILVICULTURAE MENDELIANAE BRUNENSIS

Volume LXI 262 Number 7, 2013

http://dx.doi.org/10.11118/actaun201361072337

AUGMENTED REALITY SERVICES

Tomáš Koubek, David Procházka, Jiří Šťastný

Received: April 11, 2013

Abstract

KOUBEK TOMÁŠ, PROCHÁZKA DAVID, ŠŤASTNÝ JIŘÍ: *Augmented reality services.* Acta Universitatis Agriculturae et Silviculturae Mendelianae Brunensis, 2013, LXI, No. 7, pp. 2337–2342

We assume that one of the key reasons is in the difference between a standalone application and a web service. Both architectures have some advantages and disadvantages. The Standalone application (e.g. Nokia/OVI Maps) provides the required functionality. From the user point of view, main asset of this "offline" approach is network connectivity independence. However, this kind of applications must be upgraded manually. Moreover, it is hard to get any data about the application usage because it requires additional actions from the user – data are usually acquired through conventional ways, such as email or web forms.

The online service such as Google Maps (including its mobile application) can offer the same functionality as the offline application. Nevertheless, a permanent connection to provider servers is necessary. This can be taken as a drawback. On the other hand, usage data collection is easier and can be done without the user intervention. The data collection provides a valuable analysis basis of the user habits and needs. This analysis is necessary for design of a complex "user" based solutions such as Google Now.

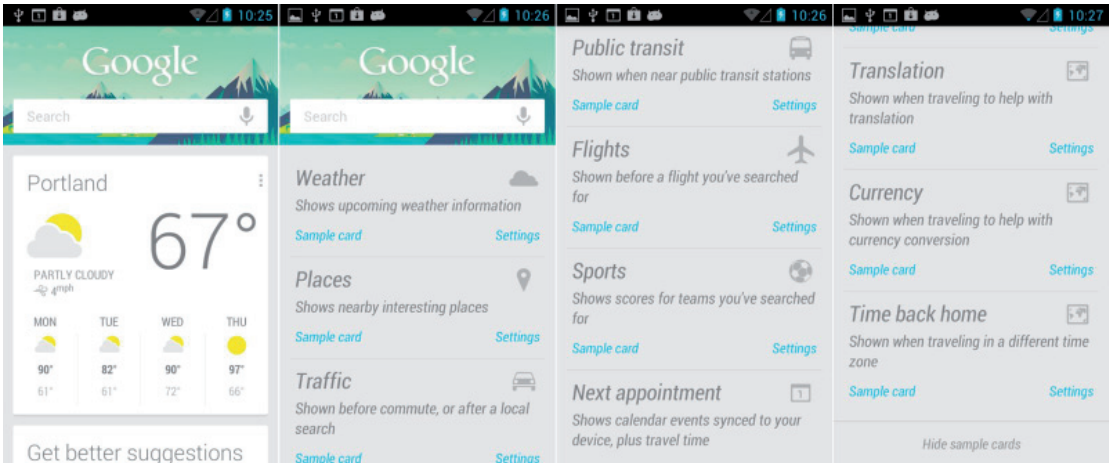
Augmented reality applications are usually based on the first mentioned approach. In this article, we describe our model of augmented reality as a service and compare its features with standalone solutions. Further, other important key aspects for large emergence of augmented reality services in a mainstream market are discussed.

augmented reality, location based services, server client

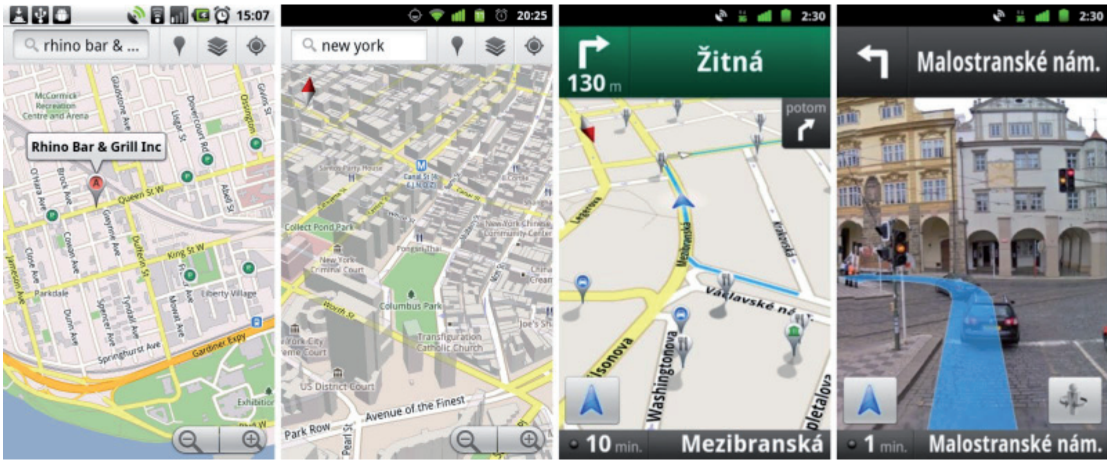
For almost two decades, Nokia has been the leader of location based services research. Nokia employees wrote some of the best papers in the field. Nevertheless, Nokia failed to implement these services into common cell phones or other devices. Six years ago Google came with the Google Maps mobile application and taught the users to locate points of interest within their vicinity. A few years later Google implemented Latitude service – a friend locator – and again succeeded. Brand new service from this year is called Google Now. It provides information based on your schedule, location, weather, traffic etc. It is a pure location based service par excellence. There must be a question asked: Why did Nokia fail?

The areas of location based services and augmented reality are currently discussed separately. We argue that this separation is purely artificial. From our point-of-view, applications from both areas solve the equal problem. The key difference is just in the information presentation. The location based service can be described simply as an application providing required information with regard to the user location. For instance, in case user stands within the area of a train station, this application is able to detect his/her position and provide automatically train schedule for this particular station. The location works as a kind of input device. Advanced application based on this principle is Google Now. Google Now provides different information cars that are based on user location, expected events, contacts etc. (see Fig. 1).

The augmented reality application is usually able to detect objects within the user surrounding using image processing and replace these objects with some connected virtual objects (e.g. augmented reality game Arhhh! described in Kroeker 2010, that inserts into the scene on a cellphone or tablet virtual zombies). Nonetheless, the information that is embedded into the scene is not necessarily based



1: *List of possible Google Now cards. Example of location-based information provided by Google Now application.*
(source: theappslab.com)



2: *Smooth transition from common map with location-based information, through 3D view and common navigation to full AR view using Street View*
(Sources: zive.cz, androidmarket.cz, torontogreenp.appspot.com)

just on image processing (Škorpil, Šťastný, 2009). For instance, the well known augmented reality application Layar[1] uses just GPS. Therefore, it shows the additional information purely on the location basis. If we summarize the gist of the augmented reality approach, the key is not the method used for an object or user detection; it is the way, how information is presented – a combination of a real-world image and additional digital information (text, 3D model etc.). Hence, even though there are exceptions on both sides; the location based services and augmented reality applications are substantially overlapping areas. On the Fig. 2, it is possible to see a smooth transition from a common location based application to clear example of an augmented reality application.

In last years, there was a substantial emergence of the location based services. Examples can include location based searching, map applications, notes with stored location and automatic transportation schedules. With the occurrence of this area grows also the importance of augmented reality that provides a very straightforward way for information presentation. Therefore, we have focused on the issue of merging these two areas from the architectural point-of-view.

1 Common applications

As described in the Introduction, Location based services (LBS) applications integrate its user or device location with other information to provide an added value to the user (see Schiller, Voisard, 2004). Nowadays, LBS are used in many applications: military or emergency solutions and commerce. Last group consists especially of mobile technologies

1 http://www.layar.com

such as cellphones and tablets. In our article, we want to aim at the third category.

Contemporary mobile device is embedded with useful sensors as GPS and GLONASS navigation system receivers, accelerometer, gyroscope, magnetometer and so on. These sensors help detect the user's location (geographical coordinates, altitude) and orientation (cardinal directions from the Earth geomagnetic field). This piece of information allows the developer to create new contexts in an existing application (e.g. determine weather in the current location, regardless of user input or movement) or an entirely new service (e.g. position of another person or device on the map, see Schiller, Voisard, 2004).

These location services mainly works outside, whereas inside building GPS works hardly. Also, the magnetometer is sensitive to electromagnetic interference. There are several methods for the indoor location detection, mainly based on wireless technologies, such as WiFi, the cellular phones of RFIDs (more in Lukiankto, Sternberg, 2011).

As outlined in the Introduction, augmented reality applications display the real world and augment it (combine it) with other pieces of information (e.g. 3D models, text, video, sound etc.). Term registration, sometimes called tracking, represents searching of an appropriate place and position in a real environment for addition of corresponding objects. As described in Li *et al.*, 2008, there are 3 basic groups of methods to determine the correct position of virtual objects – computer vision based techniques, knowledge based techniques and tracker based techniques. These techniques differ in methods and technologies that are used. The last group techniques methods match with methods for LBS position and orientation detection. Shortly, all methods, used for LBS location detection can be used for AR position detection. The purpose is same for all of them – to find an appropriate location.

2 Architecture comparison

One detail distinguishing AR and LBS application is the architecture. AR software is often based on a stand alone paradigm. This thick client representation is made to work out-of-the-box, so all required parts are stored within the device: suitable tracking algorithms, data for augmentation (e.g. 3D objects and sounds) etc. Advantages of this solution are clear; Application works offline, thus it does not depend on network connection parameters. Also, there is low security risk, because of application taciturnity. On the other hand, various problems arise with this approach: At first, resources of the device are frequently limited (very limited amount of data can be stored, computation performance is incomparable with common desktop or server CPUs etc.). The thick client is habitually created for one platform; therefore, it frequently does not work on other platforms. Very noteworthy is also the fact that data updates have to be done individually on each device.

On the contrary, LBS applications frequently use the service based approach (client-server architecture). Data and even whole applications are stored at one place. Thus being accessible from virtually any device. Users connect to this service usually via thin clients. It provides e.g.: cloud data storage, data sharing among users, simple updates of algorithms and data, automatic feedback from users etc. Moreover, it is easier to create a multi platform thin client software then create a multi platform stand alone application (dependence on other APIs etc.). Service also does not depend on the user device. In comparison with thin client device (mobile phone, desktop computer etc.). The necessary Internet connectivity is the biggest bottleneck of this approach. Besides the costs and power consumption, there exist risks of denial of service. Therefore, this approach requires a subtle technical solution and consideration of possible security risks.

Despite mentioned problems, there is a clear trend in LBS: move the application data or move whole application engine to a remote server/cloud. Important developers such as TomTom or Google link their devices and applications with their servers and offer online traffic reports, searching for persons, point of interests and other information pieces. Described advantages outweigh infrastructure and security issues.

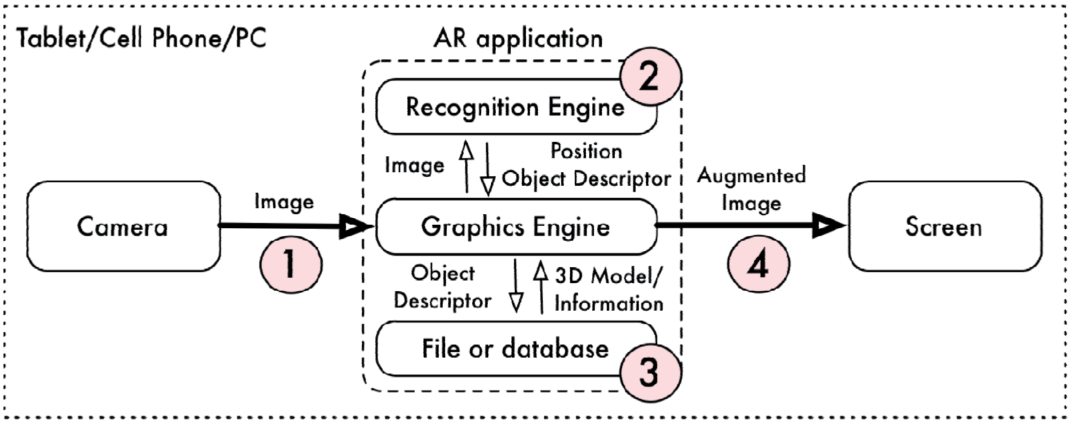
3 Sustainable AR architecture and implementation

At first, some key problems of augmented reality applications should be discussed. In accordance to Gervautz, Schmalstieg (2012), certain issues of AR applications can be detected in scope of concurrent applications.

- There is lack of experience with augmented reality user interfaces.
- Semantics of environment artifacts is required (AR will require not only information on geometry and appearance but also meaning).
- Improvement of tracking and mapping environment is necessary.

Moreover, a just few AR applications use the service based approach that is suitable for effective data management (e.g. Layar, WikiTude or SekaiCamera). Although some articles focus on AR application component architecture (see e.g. Bauer *et al.*, 2001 and Friedrich, 2002), they use closed application databases or did not mention data storing at all. Therefore, data management such as updating or using user data, depends on the vendor of the application.

This goes hand in hand with another already mentioned issues of contemporary AR application architecture: whole application runs on the user device. Data are stored within the application, or the application is connected to single remote database provided by the application vendor. An example of such AR application architecture is shown in Fig. 3.



3: *Common AR application architecture*

3.1 Open communication infrastructure

The first problem we focused on is the open and sharable data structure. As been explained above, both LBS and AR applications store information linked to locations. Therefore, the solution might be based on open standard that is used for LBS data storage – the geodatabases. Geodatabases are widely used, well documented and opened. Communication standards for querying databases, such as widely-used geospatial standards Web Map Service (WMS) or Web Feature Service (WFS) are available and also well documented. Geodatabases bring the advantages of client-server architecture as well as the possibility to use data from more sources. Our solution assumes usage of appropriate geodatabase extensions that are designed for storage of non-text information such as 3D models or multimedia content. Geodatabase accompanied with appropriate geospatial standard enables to connect to all required data sources with own application. This communication standardization is similar to XMPP/Jabber communication protocol, web servers or already mentioned web mapping services. From the long term point-of-view, these open standards are used for longer periods of time than the closed solutions (e.g. proprietary email protocols vs. POP3/SMTP, ICQ vs. XMPP/Jabber etc.).

Implementation

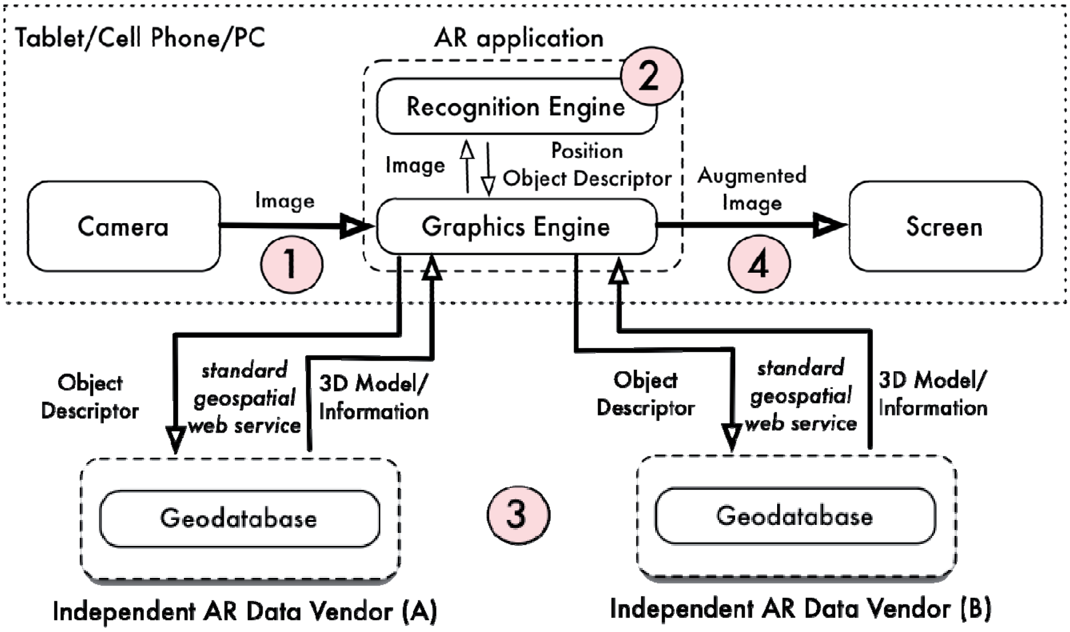
Several databases for geospatial data are offered. One of them, PostGIS[2], is an acceptable solution for AR data storage. PostGIS adds spatial support for database system PostgreSQL. PostGIS extends PSQL functionality and offers a possibility of location queries. It supports import of common formats (e.g. KML). PostGIS also allows 3D object storage. This is significant for AR applications. PostgreSQL also supports multimedia storage; therefore, all required kinds of data can be stored within one location.

Communication protocols maintained by Open Geospatial Consortium (OCG) are commonly used for communication with geospatial databases (see Davis, 2005 or Mitchell, 2005). One of these, Web Feature Services (WFS), is able to offer data from geospatial database or other information on demand. WFS can handle just vector graphics (e.g. in KML or GML format). Support for AR data has to be added, because some crucial information (e.g. tracking method and tracking data or 3D model data information) is needed. Possible option is the usage of *wfs_metadata* attributes for own AR extension of WFS. Metadata specified in by this attribute can be stored within the database, of course. This extension enables all other communication features necessary for AR. Other possible communication protocol is Web 3D Service (W3DS), which was proposed for handling with 3D models, so it provides useful attributes (e.g. level of detail, scene adjustments). Unfortunately, current version of W3DS is still declared just as a candidate for standard, so it is not fully specified. Also, this communication protocol needs AR extension.

3.2 Modularity

Another key feature of our architecture is application modularity. System layers (or components) can be separated for image processing, data storage, user behavior analysis etc. This design is analogous to the model-view-controller design pattern. Each layer provides the required functionality independently, just with inputs or outputs from neighbour layer, and could be implemented as service (e.g. data, image processing) or client side (image processing, graphics engine). The application developer is allowed to distribute computational power between a client device and server. Modularity also leads to cooperation and incremental enhancements without influence on stored (geo)data. For instance, Ha *et al.*, 2010, compares client-side and server-side image

2 http://postgis.net



4: *Proposed AR architecture*

processing and from the results is obvious, that even such complex real-time tasks can be outsourced to the cloud (in detail comparison is in Prochazka *et al.*, 2011). The scheme of our architecture is in Fig. 4. The application uses widely used geospatial web services for connection to independent data vendors. Hence data are shared among the applications.

SUMMARY

The article reviews contemporary LBS and AR application architectures. We argue that augmented reality applications and location based services are overlapping fields. In both cases, their data are related to locations. We focused on key problem of AR applications and proposed new architecture that is based on two principles. The first one is that the application is service oriented, with emphasis on independent function layers for data presentation, user location recognition and data management. The second principle is that data are stored into geodatabase and communication between client and server side is established via appropriate widely-used geospatial standard. The proposed architecture cannot solve all of the key problems; however, it can substantially improve currently used architectures and open them for further enhancements.

Acknowledgement

This work was supported by grant IGA FBE MENDELU 18/2013 (Enhancement of property management effectiveness using point clouds).

REFERENCES

BAUER, M. G., BRUEGGE, B., KLINKER, G. J., MACWILLIAMS, A., REICHER, T., RISS, S., SANDOR, C., WAGNER, M., 2001: Design of a component-based augmented reality framework. In: *Proceedings of IEEE and ACM International Symposium on Augmented Reality*. New York: IEEE Computer Society, pp. 45–54. ISBN 0-7695-1375-1

DAVIS, S., 2005: *GIS for Web Developers*. Raleigh, Texas: The Pragmatic Programmers LLC, pp. 252. ISBN 978-0-9745140-9-3.

FRIEDRICH, W., 2002: ARVIKA-augmented reality for development, production and service. In: *Proceedings of International Symposium on Mixed and Augmented Reality*, ISMAR 2002, Darmstadt: IEEE Computer Society, pp. 3–4. ISBN 0-7695-1781-1.

GERVAUTZ, M., SCHMALSTIEG, D., 2012: Anywhere Interfaces Using Handheld Augmented Reality. *IEEE Computer*, 45, 7: 26–31. ISSN 0018-9162.

HA, J., CHO, K., YANG, H. S., 2010: Scalable recognition and tracking for mobile augmented reality. In: *Proceedings of the 9th ACM SIGGRAPH Conference on Virtual-Reality Continuum and its Applications in Industry (VRCAI '10)*, Seoul: Association for Computing Machinery, pp. 155–160. ISBN 978-1-4503-0459-7.

KROEKER, K. L., 2010: Mainstreaming augmented reality. *Communications of the ACM*, 53, 7: 19–21. ISSN 1557-7317.

LI, Y., PENG, K. S., HUA, Q. Z., QIONG, Z., 2008: Development Actuality and Application of Registration Technology in Augmented Reality. *International Symposium on Computational Intelligence and Design, 2008.* ISCID '08. Vol. 2. Wuhan, China: IEEE Computer Society, pp. 69–74. ISBN 978-0-7695-3311-7.

LUKIANKTO, C., STERNBERG, H., 2011: Overview of Current Indoor Navigation Techniques and Implementation Studies. In: *FIG Working Week 2011 – Bridging the Gap between Cultures*, Marrakech, Morocco: International Federation of Surveyors, pp. 1–14. ISBN 978-87-90907-92-1.

MITCHELL, T., 2005: *Web Mapping Illustrated*, Sebastopol: O'Reilly Media, pp. 349. ISBN 978–0–569–00865–9.

PROCHÁZKA, D., ŠTENCL, M., POPELKA, O., ŠŤASTNÝ, J., 2011: Mobile Augmented Reality Applications. In: *Mendel 2011: 17th International Conference on Soft Computing*. 1. vyd. Brno: Brno University of Technology, pp. 469–476. ISBN 978-80-214-4302-0.

SCHILLER, J., VOISARD, A., 2004: *Location-Based Services*. San Francisco: Morgan Kaufmann, pp. 255. ISBN 1-55860-929-6.

ŠKORPIL, V., ŠŤASTNÝ, J., 2009: Comparison Methods for Object Recognition. In: *Proceedings of the 13th WSEAS International Conference on Systems.* Rhodos, Greece: WSEAS Press, p. 607–610. ISBN 978-960-474-097-0.

Address

Ing. Tomáš Koubek, Ing. David Procházka, Ph.D., prof. RNDr. Ing. Jiří Šťastný, CSc., Department of Informatics, Mendel University in Brno, Zemědělská 1, 613 00 Brno, Czech Republic, e-mail: xkoubek0@mendelu.cz