This is a preliminary version of an article published in
Proc. of the International Workshop on Human Control of Ubiquitous Systems (HUCUBIS), Dublin, Ireland, July 2008
by Stephan Kopf, Thomas King, Philipp Bostan, Hendrik Lemelson, Sina Deibert, Wolfgang Effelsberg

Fundamental Services for Context-Sensitive Mobile Applications

Stephan Kopf, Thomas King, Philipp Bostan,
Hendrik Lemelson, Sina Deibert, Wolfgang Effelsberg
University of Mannheim
68131 Mannheim,
Mannheim, Germany
{kopf | king | bostan | lemelson | deibert | effelsberg}@uni-mannheim.de

ABSTRACT
A challenge for the development of mobile applications is the heterogeneity of the different device classes. For instance, an application which runs well on a personal digital assistant (PDA) might be unsuitable for mobile phones. To facilitate the development of mobile applications we have developed several services. We will elaborate on three fundamental services which simplify the user interaction: service discovery, indoor positioning, and image adaptation. These services enable a user-centered configuration of mobile applications. The search for a special service based on text queries or web browsing may be long-winded, because many services are not suitable for certain mobile devices. We have developed a *context-sensitive service discovery* technique which analyses user preferences and user context to select suitable services. An important context attribute is the current position of a user. Therefore, besides outdoor positioning services, we have developed an *indoor positioning* service, which reliably detects the position of a mobile device in buildings. The visualization of multimedia data is another important aspect for the human computer interaction due to the different display resolutions of mobile devices. Exemplarily, we present the *image adaptation* service which identifies relevant semantic content in images and adapts an image in an intelligent way to the screen resolution of a mobile device.

Categories and Subject Descriptors
H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *Graphical user interfaces (GUI), User-centered design.*

C.2.1 [**Computer Communication Networks**]: Network Architecture and Design – *Wireless communication.*

General Terms
Algorithms

Keywords
Context-sensitive service discovery, indoor positioning, image adaptation.

1. INTRODUCTION
Although, a great number of people in developed countries use mobile devices, the number of applications which are available on all the different mobile device classes is still relatively low. Usually, applications are only developed and tested for a certain device class (e.g., PDAs) and have to be adapted for other device classes (e.g., smart-phones). The porting of applications to the wide variety of mobile device classes is a time consuming issue. Additionally, testing and maintainability of applications is getting cumbersome for such a huge number of different mobile device classes. Fundamental services that support the development of applications for those heterogeneous mobile devices classes and in general middleware to support context-sensitive applications are still rare.

In contrast to typical desktop applications, mobile applications are different since they need to consider mobility, resource limitations, heterogeneity, personalization and especially different requirements on usability and usage patterns. Particularly personalization which is bound to the context of the user, his device and the current environment, is an important issue for mobile applications. Context information like the current position and the environment of a user may help to select more suitable services, information or simply to adapt a mobile application to the current context or situation. Typical examples for context-sensitive applications are service discovery services, which intend to deliver services or information to the user that best fit the current context. For instance, in case of a restaurant finder application, it would be helpful if the user's current position, his preferences like 'non-smoking', 'prefer Italian food', 'outdoor seats', or 'not too expensive' are considered. If the calendar of a user indicates other appointments in the near future the proposed restaurants should not be too far away (depending on available transportation).

In the case of context-sensitive service discovery, it is important to provide users with enhanced support for finding suitable services in an efficient way. Long-winded interactions between users and mobile devices (e. g., browsing a huge list of services) in the process of service discovery should be avoided since users are restricted in their input capabilities. A middleware, which enables efficient service discovery, is a solution for this problem. Context-sensitive service discovery delivers significant added value since it considerably reduces the required level of interaction and delivers personalized, precise search results that are suitable in the user's current context. We have developed a *context-sensitive service discovery* approach which is presented in Section 2.1.

Since Schilit et. al. initiated research on context-aware computing in 1993 starting with their PARCTab project [28], various researchers have also focused on the subject of context-sensitive (context-aware) service discovery. In the CB-Sec project [25] an architecture has been developed that focuses on service discovery of Web Services using contextual information. Context attributes that are used for service discovery are for example, the location of people, device and service capabilities, context history, and time of day. Another approach that relates to context-sensitive discovery of Web Services is described in [24]. This approach is based on the matching of the user's context and enhanced service descriptions stored in a UDDI repository. Furthermore, their service descriptions include feedback information, which is the context at the time of service recommendation. In the COSS approach [5] ontologies are used for context and service descriptions. Context information is delivered by context provider for the matching process of context and services. For our approach we have chosen a simple markup scheme model that is based on XML and can be extended for several purposes [1, 2].

Usually, the current position of a user is the most important context information. Although, the *Global Positioning System* (GPS) [14] is the predominant system for outdoor positioning, a major drawback is the fact, that radio signals from GPS satellites cannot be reliably received in street canyons created by large buildings, near walls or within buildings. On the other hand, 802.11 infrastructure is usually available in almost all places where people live and work. 802.11 is a wireless local area network technology that is used to provide Internet access for users; however, it can be used for positioning purposes at the same time [18].

The second service we will present in Section 2.2 is an *indoor positioning service* based on 802.11, which determines the current position of a mobile device with high accuracy. Many indoor positioning systems (e.g., [3, 7, 10, 11]) make use of 802.11, because almost all modern mobile phones and laptops are equipped with this wireless technology. Therefore, the devices can be used for positioning as they come out of the box, which means that no additional hardware is required.

Another important aspect for the development of mobile applications is the visualization of multimedia data on different mobile devices. The third service in Section 2.3 handles the presentation of images on mobile devices with different screen resolutions. To display an image it is necessary to adapt the resolution of the image to the screen resolution of the mobile device. A simple scaling of images does not always lead to satisfactory results due to the loss of relevant semantic details. The presentation of the full images in combination with user navigation (scrolling) is usually not acceptable. We have developed an *image adaptation service*, which identifies relevant parts and objects in an image and combines them into a region of interest to be presented on a mobile device.

The goal of image adaptation is the adjustment of image parameters to the physical features of the display of a mobile device [15, 26]. An interesting approach for resolution adaptation is to identify relevant objects (attention objects) [13, 29], which should be recognizable in the adapted image [8, 9]. Important objects combined with an analysis of the color distribution, the contrast and the orientation of edges are suitable to define a region of interest [4, 12, 22, 27].

2. SERVICES FOR CONTEXT-SENSITIVE MOBILE APPLICATIONS

We have selected three services which seem to be highly relevant for a great number of mobile applications. The services enable a user-centered configuration of mobile applications. First, we focus on a context-sensitive service discovery technique. In the following, we will present two services which provide indoor positioning and image adaptation.

2.1 Context-sensitive service discovery

Many sensor technologies have evolved rapidly over the last few years and now support the automatic sensing of context information on mobile devices. Context information sets the foundation for the next generation of mobile services - *context-sensitive services* which deliver services (or information) that best fits the current context of the user. However, middleware support for the development of context-sensitive applications that support service discovery is still missing.

We have built up a generic software platform that supports the development of context-sensitive discovery services for service brokers and providers. Furthermore, this software offers a client framework that supports the development of generic client applications for mobile devices that enable the dynamic integration and execution of discovered services.

To identify relevant services, a user sends a service request to a centralized service broker that delivers a choice of electronic services that he can use for further exploration. The service broker prefilters services that fit to the user's current context (e. g., location, time, weather, user profile). The user selects one of the specialized services by category and if necessary the client application automatically downloads and executes components that are needed to consume the service. On sending a service request to the specialized service using the implicit available context information of the mobile device, the user is returned a personalized, customized list of services suitable to the submitted context.

For our purposes of matching services, we have chosen a simple, flexible model for context that fulfills our requirements and is easy to apply for service brokers and providers. To define the *service description*, we have developed a generic XML-based schema that separates the core data from a domain-specific part of information, since one generic schema suitable for all kinds of services is neither realizable nor flexible. The core is divided into the three main categories: ServiceProfile, ServiceProperties, and ServiceGrounding.

The *ServiceProfile* contains information about the service and its provider. Therefore, a general text description of the service as well as a categorization of the service is specified. Extensions for domain-specific service descriptions and respectively the schema for the extension of the core are added to the ServiceProfile. The *ServiceProperties* describe the properties of a service including the spatial and temporal availability of the service, as well as aspects like payment and security restrictions. While temporal availability for non-electronic services is self explanatory the spatial restriction can be used for example for defining an area of service delivery, e.g. a pizza service that delivers its service in a circle area of 10 miles. The *ServiceGrounding* delivers access information for services dependent on their type. A non-electronic service for example requires the description of the physical location while an electronic service in contrast requires access data like web or server addresses.

The basic goal of our context-sensitive service discovery approach is to match a given set of contextual information to service descriptions. This approach is applicable in any Service Discovery Service, which is the name of a service broker in our architecture. Therefore, we use a transformation approach that relies on facts contained in the service descriptions and uses rules for the transformation into a contextual representation that can be matched against a given context set.

The most important ingredient of the context-sensitive service discovery process is the *context matcher* that tries to obtain the most valuable services offered by service providers for a given set of context attributes. As the transformation relies on the same context representation using context data types and namespaces to uniquely identify a context attribute, the matching process compares the given context set and the transformed context set for equal context data types that share the same namespace. If this prerequisite is fulfilled a predefined matching routine for each equal context attribute and type is applied. Furthermore, with the transformation of service descriptions into the contextual representation it can be defined which context attributes are mandatory and which are optional. The result of the matching process is an ordered set of services that can be ranked by the degree to which they match the requesting client.

2.2 Indoor positioning service

The best positioning results are achieved with 802.11-based positioning systems that utilize the so-called fingerprinting approach. This approach consists of a two-stage mechanism: A training phase and a position determination phase. During the *training phase*, specific radio characteristics from nearby access points are gathered at pre-defined reference spots in the operation area and stored together with their physical coordinates in a database: They are called *fingerprints*. In the *position determination phase*, the user's mobile device samples specific radio characteristics at its (unknown) position and searches for similar patterns in the database. The closest match is selected and its coordinate returned as a position estimate. A wide range of radio characteristics can be used for fingerprinting: For instance, nearly all positioning systems work with signal strength (e.g., [3]), others additionally utilize signal to noise ratio (e.g., [6]), or response rate (e.g., [23]).

In our approach, the user's mobile device is used to sample the fingerprints of adjacent access points and to compute a position estimate. A server provides the fingerprints in the proximity of the user derived from the access points in range. There are several advantages using this approach: In most cases, a sensor to sample radio characteristics, such as an 802.11 network card, is already part of the user's mobile device. To make use of this sensor, only additional software needs to be installed. Furthermore, the infrastructure can be used without any changes and without even knowing that it is used in a completely new way. Privacy concerns remain a major barrier to the adoption of context-aware applications and services. If the data required to calculate a position estimate is sampled and processed on the user's device, the user is in control of disclosing the position to whom, when and at which level of granularity.

We have developed the *Loclib* library for collecting the signal strength measurements [16]. Loclib is a library that provides methods to invoke a scan and returns signal strength measurements collected from the driver of the selected network card. Based on this library, we have derived an indoor positioning service to estimate the position of a mobile device while it is staying inside buildings. In most evaluated indoor scenarios [17, 19], the average position error is about two meters, and the position accuracy is usually sufficient to enable context-aware applications.

Beside the basic algorithm, we have proposed several new techniques to improve the performance of the indoor positioning service. One idea is to use an on-demand fingerprint selection algorithm to avoid the cumbersome and time-consuming task of manually copying fingerprint data from a single repository to mobile devices. Our algorithm dynamically selects only those fingerprints from the repository that are required to compute a position estimate. This also makes sure that the latest fingerprints are always available on mobile devices.

Additionally, we have analyzed how the scanning of the available access points affects the real data transfer in 802.11 networks. On the one hand, the positioning system requires a steady stream of active scans to be able to deliver accurate position estimates to context-aware application and services. Performing an active scan means that the network card switches through all the different channels in search of access points (this requires about 260 milliseconds) [20]. During this time no communication is feasible. On the other hand, there are the demanding real-time and multimedia applications. We have developed a new system which classifies whether the user is moving or standing still [20]. This movement detection system analyses the variance of the recent signal strength values to judge whether or not the user is moving. Based on this classification, we use active scanning in case of the user is in motion. Otherwise, the so-called monitor sniffing approach is used [18]. The throughput, delay and packet loss are significantly improved by our approach.

2.3 Image adaptation service

A very important characteristic of mobile devices is the limited resolution of the display. If images are presented on mobile devices with different screen resolutions in a naïve way, the content cannot be recognized on small displays. A good adaptation service would decide which parts of an image should be presented. We have developed an adaptation service for multimedia content to visualize data on mobile devices with different screen resolutions. The adaptation service must guarantee that even in case of very small displays the relevant content can be recognized.

We have developed a server-based service for the adaptation of images. A client-based adaptation is not feasible because the transfer of large images would cause a lot of traffic in wireless networks, and the analysis and adaptation of an image requires a high amount of computation power. If possible, the analysis of images is done offline, and the results are stored in an XML format. The adaptation based on the XML data is done in real-time. In the following, we focus on image adaptation although the adaptation service supports the visualization of videos or web pages as well.

The *image adaptation service* applies a combination of scaling and cropping of borders. In a first step, *relevant regions* based on the following heuristics are identified:

- Regions with relevant semantic content should be clearly visible in the adapted image. A region should not be part of the adapted image if the semantic content is no longer recognizable due to its limited resolution (e.g., text is no longer readable). We apply algorithms to automatically identify relevant regions with high contrast, text regions and faces in images.
- Regions without expressive content should not be part of the adapted image. Large near-monochrome regions (e.g., sky) adjoining an image border are typical candidates of irrelevant regions.
- A relevant region is scaled to the appropriate screen resolution. The aspect ratio of the selected region should be identical with the aspect ratio of the adapted image.

The relevant regions define the position and size of semantic features. A rating of the semantic features is necessary to identify one *region of interest* (ROI) to be presented on a mobile device. User preferences and the size of a feature in the adapted image are relevant for the rating.

The following approach guarantees a good compromise between scaling and cropping [21]: Each semantic feature is characterized as a rectangle. We assume a proportional coherence between the size of a semantic feature in the adapted image and the visual information in the image. A *minimal perceptible size* is defined for each feature. The content is no longer recognizable if the size of a feature drops below this value. Additionally, we can define an upper size for each feature (*maximal reasonable size*). For instance, if the font size of a text is readable, an additional enlargement of the characters does not provide new information. A region of the image is selected and scaled to the correct size to calculate the visible information in the adapted image.

It is the goal to specify the position and size of the region *R* which maximizes the visible information in the adapted image. The total amount of information $V_{sum}(R)$ is defined as the sum of the semantic features *i* in the selected region *R*:

$$V_{sum}(R) = \sum_i S_i(R) \cdot V_i(R) \quad \text{with}$$

$$V_i(R) = \begin{cases} \frac{H_{max}}{H_i(R)} & H_i(R) > H_{max}, \\ \frac{H_i(R)}{H_{max}} & H_{min} \leq H_i \leq H_{max}, \\ 0 & H_i(R) < H_{min}, \end{cases}$$

$$S_i(R) = \begin{cases} 1 & \text{if } V_i \text{ is part of } R, \\ 0 & \text{else.} \end{cases}$$

$V_i(R)$ estimates the information of the semantic feature *i* and the selected region *R*. A feature is used only if it is completely included in the selected region. The binary variable $S_i(R)$ specifies whether the region of the semantic feature is totally included in the region. The thresholds H_{min} and H_{max} define the minimal perceptible size and maximal reasonable size of features. Both values depend on the characteristics of the display, the distance between user and display, and individual user preferences.

The calculation of all positions and sizes for region *R* is not possible due to the large number of combinations. The number of possible regions can be reduced significantly if the maximal reasonable size is not considered. Each border of the selected region must match with at least one border of a semantic feature. We assume that this condition is valid for the selected region. If we reduce the size of the region marginally, at least one semantic feature *i* will no longer be included in the region and $V_{sum}(R)$ drops approximately by $V_i(R)$. A very small increase of the size of the region leads to a higher scaling factor, and the values of all features in this region are decreased. Therefore, it is mandatory that the borders of the features and the region match. $V_{sum}(R)$ is not necessarily optimal (e.g., in case of text regions) because the maximal reasonable size H_{max} reduces the total information of a region, and smaller regions may increase $V_{sum}(R)$. The maximal reasonable size is ignored in the first step and the region *R* is initialized. In a second step, the size of the region is enlarged until the maximum of $V_{sum}(R)$ is detected.

The position and size of the optimal region depends on the distribution of the semantic features in the image. Each feature is selected as a candidate region and the information value $V_{sum}(R)$ is calculated and stored for single features. In a second step, the bounding rectangles of two features define the regions. The approach continuous until the calculation of $V_{sum}(R)$ is completed for all combinations of features. The region *R* which maximizes $V_{sum}(R)$ defines the region for the adapted image [21].

3. BUILDING CONTEXT-SENSITIVE APPLICATIONS

We have developed two demo applications – a bargain hunter and a gastronomy guide – to prove the concept of our approach. In the following, we will use the gastronomy guide to demonstrate the interrelation between the different services. The idea in our scenario is that a person arrives in a city and wants to find a place to eat. Context information of the user like means of transportation (on foot, tram, or car), current time, current location, preferences to eat, type of restaurant, current weather, etc. should be considered for the selection and ranking of the restaurants.

The service provider has to register the restaurant service at a service discovery service (SDS). We offer an interface to register a new service (e.g. in the style of a web site form, or a web service interface used by an application). Any client application that is build upon our framework is able to send service requests to the SDS. The mobile client is the initial requester of the restaurant services. The SDS returns the restaurant discovery service and interacts directly with it.

Information about the user context is sent to the restaurant discovery service. Very relevant context information is the current position of the user which is provided by the positioning service. The position is relevant to identify all restaurants which can be reached in a certain time (depending on the context information 'means of transportation') and to create the route to the destination. A weather forecast service which looks up the

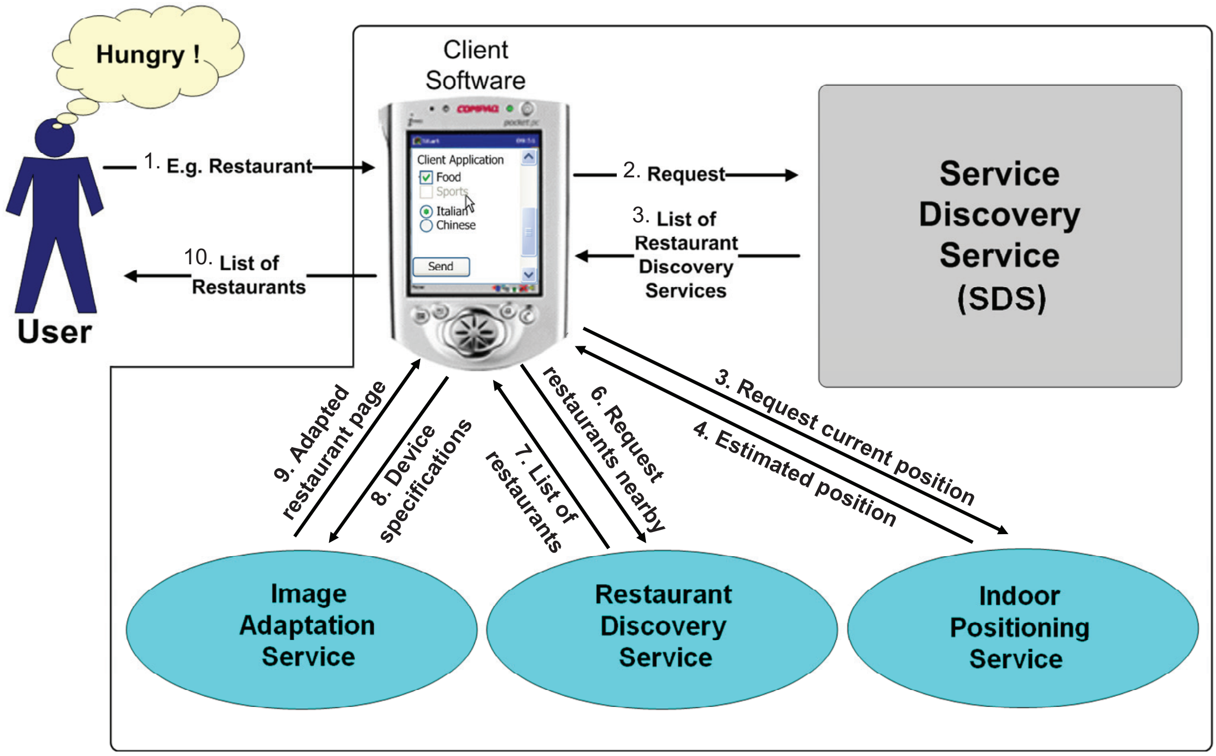

Figure 1: Example of the gastronomy guide application

weather in a certain city is used to filter outdoor restaurants in case of bad weather. Other user preferences (e. g., prefer Italian food, smoking allowed) are used to rank the retrieved list of restaurants.

The user can select a restaurant from the list and look at details like the menu card or images presenting the restaurant. The image adaptation service is used to derive a suitable presentation of the images for different screen resolutions of mobile devices. The images are automatically adapted on the server. This approach reduces the total traffic and the computational effort for the mobile devices. Figure 1 depicts the usage of the services for the gastronomy guide application.

4. CONCLUSIONS AND OUTLOOK

In this paper, we presented three fundamental services which can be used to enhance the functionality of mobile applications. The service discovery enables users to identify and execute services in a fast and efficient way. The indoor positioning service provides the relevant context information about the current position of a user, and the image adaptation service facilitates a suitable representation of multimedia data on mobile devices with different screen resolutions. The human control and user interaction of mobile devices is much easier by using these services. In our ongoing work we will build up new services and integrate them into mobile applications with enhanced functionality. We expect that the acceptance and usage of mobile applications will increase.

5. ACKNOWLEDGMENTS

This material is based on work supported by the Deutsche Forschungsgemeinschaft (DFG).

6. REFERENCES

[1] M. Aleksy et. al, “Interaction Styles for Service Discovery in Mobile Business Applications”. In Proceedings of 9th International Workshop on Network-based Information Systems, DEXA, Krakau, Poland, 2006.

[2] C. Atkinson, P. Bostan and T. Butter, “Context-Sensitive Service Discovery for Mobile Commerce Applications”. To appear in: The Fourth International Conference on Wireless and Mobile Communications, Athen, Greece, 2008.

[3] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In Proc. of the 19th IEEE Conference on Computer Communications, pages 775–784, 2000.

[4] M. Bertini, R. Cucchiara, A. Bimbo, and A. Prati. An integrated framework for semantic annotation and adaptation. In Multimedia Tools and Applications, volume 26(3), pages 345–363. Springer Science & Business Media B.V., August 2005.

[5] T. Broens. Context-aware, Ontology based, Semantic Service Discovery. Master’s thesis, University of Twente, Enschede, The Netherlands, 2004.

[6] P. Castro, P. Chiu, T. Kremenek, and R. R. Muntz. A Probabilistic Room Location Service forWireless Networked Environments. In Proc. of the UbiComp, 2001.

[7] P. Castro and R. Muntz. Managing Context Data for Smart Spaces. IEEE Personal Communications, 2000.

[8] L.-Q. Chen, X. Xie, X. Fan, W.-Y. Ma, H.-J. Zhang, and H.-Q. Zhou. A visual attention model for adapting images on small displays. In ACM Multimedia Systems Journal, volume 9(4), pages 353–364. ACM Press, 2003.

[9] X. Fan, X. Xie, W. Ma, H. Zhang, and H. Zhou. Visual attention based image browsing on mobile devices. In Proceedings of IEEE International Conference on Multimedia and Expo (ICME), volume 1, pages 53–56. IEEE Computer Society Press, July 2003.

[10] A. Haeberlen, E. Flannery, A. M. Ladd, A. Rudys, D. S. Wallach, and L. E. Kavraki. Practical Robust Localization over Large-Scale 802.11 Wireless Networks. In Proc. of the 10th ACM Conference on Mobile Computing and Networking, pages 70–84, 2004.

[11] J. Hightower and G. Borriello. Location Systems for Ubiquitous Computing. Computer, IEEE Computer Society Press, 34(8), August 2001.

[12] J. Hu and A. Bagga. Categorizing images in web documents. In IEEE Multimedia, volume 11(1), pages 22–30. IEEE Computer Society Press, January 2004.

[13] A. Santella, M. Agrawala, D. Decarlo, D. Salesin, and M. Cohen. Gaze-based interaction for semi-automatic photo cropping. ACM Human Factors in Computing Systems (CHI), pages 771-780, 2006.

[14] E. Kaplan and C. Hegarty, editors. Understanding GPS: Principles and Applications. Artech House Incorporated, 2005.

[15] D. Kasik. Strategies for consistent image partitioning. In IEEE Multimedia, volume 11(1), pages 32–41. IEEE Computer Society Press, January 2004.

[16] T. King, T. Butter, H. Lemelson, T. Haenselmann, and W. Effelsberg. Loc{lib,trace,eva,ana}: Research Tools for 802.11-based Positioning Systems. In Proc. of the Second ACM Workshop on Wireless Network Testbeds, Experimental evaluation and CHaracterization, pages 67–74, 2007.

[17] T. King, T. Butter, M. Brantner, S. Kopf, T. Haenselmann, A. Biskop, A. Färber, and W. Effelsberg. Distribution of Fingerprints for 802.11-based Positioning Systems. In Proc. of the MDM, 2007.

[18] T. King, T. Haenselmann, S. Kopf, and W. Effelsberg. Overhearing the Wireless Interface for 802.11-based Positioning Systems. In Proc. of the IEEE PerCom, 2007.

[19] T. King, T. Haenselmann, S. Kopf, W. Effelsberg: A Measurement Study on 802.11 concurrently used for Positioning and Communications. In Proceedings of the Third International Symposium on Wireless Pervasive Computing (ISPWC 2008), pp. 610-615, Santorini, Greece, May 2008.

[20] T. King, M. B. Kjærgaard: ComPoScan: Adaptive Scanning for Efficient Concurrent Communications and Positioning with 802.11. In Proceedings of the 6th ACM International Conference on Mobile Systems, Applications, and Services (MobiSys 2008), Breckenridge, CO, USA, June 2008.

[21] S. Kopf, F. Lampi, T. King, W. Effelsberg: Automatic Scaling and Cropping of Videos for Devices with Limited Screen Resolution. In Proceedings of ACM Multimedia 2006, Santa Barbara, CA, USA, October 2006.

[22] S. Kopf, T. Haenselmann, D. Farin, W. Effelsberg: Automatic Generation of Summaries for the Web. In Proceedings of SPIE, Storage and Retrieval for Media Databases, pp. 417--428, San José, CA, January 2004.

[23] J. Krumm and E. Horvitz. LOCADIO: Inferring Motion and Location from Wi-Fi Signal Strengths. In Proc. of the Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004.

[24] J. Kuck and F. Reichartz, “A collaborative and featurebased approach to Context-Sensitive Service Discovery”, In Proceedings of 5th WWW Workshop on Emerging Applications for Wireless and Mobile Access (MobEA'07), Banff, Alberta, Canada, 2007.

[25] S. K. Mostefaoui and B. Hirsbrunner, “Context Aware Service Provisioning”, In Proceedings of the IEEE/ACS International Conference on Pervasive Services (ICPS 04), 2004.

[26] T. Rist and P. Brandmeir. Customizing graphics for tiny displays of mobile devices. In Proceedings of 3rd International Workshop on Human Computer Interaction with Mobile Devices, pages 1–4, September 2001.

[27] N. Rowe. Content repurposing for small devices. In M. Pagani, editor, Encyclopedia of Multimedia Technology and Networking (Volume I), volume 1, pages 110–115. The Idea Group, Hershey, PA, USA, April 2005.

[28] W.N. Schilit et. al., “Customizing mobile applications”, In Proceedings of the USENIX symposium on Mobile and Location-Independent Computing, USENIX Association, 1993.

[29] V. Setlur, S. Takagi, R. Raskar, M. Gleicher, and B. Gooch. Automatic image retargeting. Proc. of International Conference on Mobile and Ubiquitous Multimedia, pages 247-250, 2005.