

Skyguard: A Context-Aware Mobile Information System for Emergency Management

Nan Jing

Abstract—Context awareness is one of the most critical components in building information systems because the systems use that to warrant the relevance of provided information and users' needs. Existing research work in context awareness mainly focused on analyzing either Internet-based web pages or user search keywords. There are works that develop and utilize context awareness in the domains of information system and emergency management, however these works have not specified a comprehensive framework and a clear workflow about how exactly the critical contexts are captured and utilized to produce more effective and relevant instructions. In this paper, we have reviewed the related works in the cross domains of context awareness, mobile technologies and emergency management. After identified the limitations of these works, we have proposed a mobile context-aware information system that utilizes mobile technologies and context awareness to provide relevant emergency instructions to public users. Based on this system, we also described case studies that have been conducted to validate this system. Our motivation of this work is to fill in the gap of using context awareness and mobile technology for more efficient and effective emergency management and also provide reference for similar applications in other domains.

Index Terms—Emergency management, mobile information system, context awareness, mobile software, XML.

I. INTRODUCTION

In order to properly react to and manage any emergency situation, it is vital to provide up-to-date, relevant information to both public community and emergency management centers regardless of geographical and temporal differences. Emergency situation develops fast and dynamically. Only latest information can help decision makers prepare updated instructions and only most updated instructions can efficiently help people reach safe places and save their lives. Meanwhile people in emergency can often give very little attention to the received instructions, meaning only most relevant instructions for each individual can be effective in instructing her to escape from the emergency.

However, at present, most of the applications are static, which means the process of data collection, analysis and reaction is based on traditional process and technology, such as telephone and even just words. It limits the effectiveness and efficiency of emergency management. Clearly in order to improve the timeliness of delivering emergency instructions, we need to embrace the new information technology in building the emergency management system,

which ought to efficiently deliver the effective information to the people in need. In fact, this has been made possible by recent development of handheld devices and mobile software technologies, which have together profoundly transformed the way how people communicate multi-media information including text, image, audio and video.

To date there has been an explosion of handheld digital devices in the past decade including digital media-players, digital cameras, GPS systems and networked communication devices and they have inevitably merged with the power of computers. The combination of wireless and internet technologies allows more complex tasks including seamlessly and real-time synchronizing data between mobile device and information system via wired and wireless networks, and presenting the data in multi-media formats. Mobile devices have their significant advantages in size, mobility, flexibility and usability [1], [2]. And at the same time, with mobile devices becoming more powerful and affordable, the user base has expanded from the early business elites to public population. Consequently, mobile information access is gaining widespread prominence with improving connection speed and access technologies, which should lead to richer content and better user experience in every applicable domain, including the emergency management. Unfortunately the use of mobile software technology in this domain is still in its infancy. Many emergency management centers rely on human-based information exchange and processing procedures. Some may use advanced information technology [3], but only deploy it in the headquarters, instead of implementing it using mobile technology. As such, the authors in this work recognize inherent mobile nature of emergency management and present a better-designed information system the uses mobile software technologies to deliver emergency instructions.

As mentioned, people in emergency can give very little attention to the instructions they are receiving, so only the instructions that are customized for each individual in her own context are most effective and valuable. For example, metropolitan residents may find instructions to take public transportation more feasible as the emergent situation always incur heavy city traffic, whereas the people living in suburb especially with many family members may prefer instructions that has driving directions, as they have more leeway to drive own cars to the rescue place. Another example is, people with high-profile smart phones (high resolution, super processing power) may prefer multimedia-based instructions (audio and video) as these instructions are easier to understand and follow, while people who use traditional phones may find multimedia-based instructions bothering as only texts and images are displayable. All these

Manuscript received May 4, 2014; revised September 9, 2014.

Nan Jing is with the School of Management, Harbin Institute of Technology, Harbin, 150001, China (e-mail: jingnan@gmail.com).

scenarios are real use cases that the authors have faced in practical work, and this requires a well-designed mobile information system to carefully analyze the contexts of users in emergency to provide most effective instructions. This context awareness, if well designed and implemented, can enable the emergency management system to acquire and utilize information about the physical and social situations in which users and their handheld devices are located, such as location, time, transportation, and surroundings information, in order to improve the relevance, quality, and appropriateness of delivered information in the mobile service. In the particular case of mobile software users, the relevant contexts can include location, time, environment, transportation, travel speed, travel direction, and mobile device specifics. These contexts should be thoroughly considered and carefully analyzed in designing the next-generation emergency management systems that can efficiently deliver the effective information to the public users.

Therefore, the goal of this work is to design an information system that recognizes the inherent mobile nature of emergency management and efficiently delivers more relevant information using mobile software technologies to the user's mobile device. The rest of this paper is structured for discussing the related works, application scenarios, and design architecture for this system. Section II reviews the existing related works to context awareness. Section III discusses the application scenarios of a context-aware mobile emergency management system. Section IV describes the architecture and each module of the system in details. Section V presents the preliminary design of a case study for validating the proposed system. Finally Section 6 concludes this paper and outlines the future work.

II. CONTEXT AWARENESS AND RELATED WORK

To put a context-aware mobile information system for emergency management into perspective, this section reviews the study of context awareness hereafter.

In general context means situational information. One of its popular definitions [4] is "any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves". In the studies we have reviewed, there are mainly two ways of using contexts in software applications. First, applications can optimize their outputs according to the contexts, such as major search engines using the keywords and web page content to provide more targeted information. Second, the context information can be used to create new types of applications, such as location-based applications. In these studies, context is often categorized into *physical context* representing the environment of the activity and *logical context* representing more abstract information about the stakeholder and the application. Physical context properties are at a very low level of abstraction. They are continuously updated to take into account the fact that the state of the stakeholder and the application continuously changes, such as spatial and temporal information. Logical context information is needed to enrich the semantics of physical context

information (e.g., stakeholder's preferences) thus making it meaningful for high-level purposes (e.g., stakeholder's visits to certain locations) [5]. Theoretically any information available in the course of an interaction can be used as context information, such as interaction time, event location, and user feedback. In our research, the focus is the context information that is useful and critical to determine the emergency instructions to the users who are in emergent situations and use mobile applications.

We acknowledge that context has no standard definition, since every school of study can give their understanding about context to a valid purpose. However, in the particular area of mobile software infrastructure, the target of using context is to better serve users by providing needed information to these users on mobile devices. Classification of contexts should embody mobile-user-centric essence and, particularly in our research, it should be directly helpful for us to generate and customize more effective emergency instructions for the users. From this point forward, we classify context into the information about handheld device, application user, and the surrounding environment. The *device context* may include the device manufacturer, device model, touch-screen support, screen resolution, keyboard equipment, portrait/landscape display, memory size, stylus equipment, multimedia handling, Bluetooth support, etc. *User context* typically refers to the user's personal preferences for information content and feedback to the provided information. *Environment context* includes event location, environmental lighting, movement direction and speed, user's location and current time of day. These contexts are very critical for mobile users. The awareness of these contexts can help improve user interaction by knowing a-priori the emergency situation, environment conditions, user information, device capabilities, and personal preferences, so that the user doesn't have to manually specify these constraints, and information delivery is automatically adapted to the emergency circumstances. For example, if a context-aware system knows that a user is driving a car in certain direction, it can send alerts in audio messages without requiring any input from the users, and can route the users to a safe location along user's driving direction.

In fact, context awareness is not a new topic. It has been pioneered by Mark Weiser in 1991, who focused on the context-aware computing area under the vision of ubiquitous computing (a.k.a. pervasive computing or ambient intelligence). Ubiquitous computing is a method devised to make distributed computing available by multiple computers throughout the physical environment and make them transparent to the stakeholders [6]. Context awareness as a scientific term was first introduced by Schilit [7] in ubiquitous computing. In his research, context is put into three categories: computing context, user context, and physical context. By these categories, Schmidt further defined context as knowledge of the user's and IT device's state, including surroundings, situation and locations [8]. Recently researchers have paid long due attention to context acquisition and utilization in various mobile platforms. Khedr *et al.* apply agent-based approaches for building mobile context-aware platform using the network-level context [9]. Biegel and Cahill described a framework of

utilizing environmental observance for context aware application development in ubiquitous computing [10]. Gu et al. described context models using ontology in mobile intelligent environments [11]. Another example is context-aware information provision to the mobile phone standby screen, proposed by Nakatsuru *et al.* [12], who demonstrated a system that allowed users to make word-of-mouth recommendations of shops and restaurants in a peer-to-peer fashion. The recommendation content is stored away upon receipt; only when the recipient's context matches the target context is it displayed on the mobile phone standby screen. In [13], Yu *et al.* described a framework for context-aware media personalization which performs context acquisition, reasoning and learning to build an ontology-based context model for in content filtering and recommendation. Particularly in the area of emergency management, Padilha *et al.* [14] help the commander and operation terms build an information system to support interaction between emergency response teams and to evaluate their in a response organization. Barnard *et al.* [15] discuss how various contextual conditions such as motion, lighting and task type affect user's performance and workload. In summary, these works provide valuable references and solid grounds for building the frameworks and approaches to adapt information content to match with the user and application contexts. However, all these works have not specified a comprehensive framework and a clear workflow about how exactly the critical contexts are captured and utilized to return more effective and relevant information. Filling in this gap in the special domain of emergency management, and providing reference for other applications in generic, is the main motivation of this work.

III. APPLICATION SCENARIOS

During an emergent situation, the user can initiate a request for the categories of information she seeks for, such as a hotel, a restaurant, or a rescue center. And the user's mobile device will communicate with the mobile information system that provides emergency instructions to the device. This response could include related located-based information, based on the city locations that the user moves between. As the user changes locations, the instruction updates to the new locality. Besides location, event time can also determine the presented information, such as only show the place that is open for business at the time of request. User travelling method (e.g. walking, driving, public transportation, etc.), speed, direction and environmental conditions (e.g. light, noise, etc.) also impact the displayed information. Finally the user's device capabilities also help the mobile information system determine the proper presentation format for the display information, for example, video and audio for high-profile smartphones, graphics and text for regular cell phones. Hereafter two application scenarios are presented with more details about what the proposed context-aware mobile information system can help users in the emergent situations.

In our first scenario it is in the afternoon and John is driving in the metropolitan area in Los Angeles. Suddenly he drives into the area where an emergent situation has just been detected. Based on the location and time that his

mobile device automatically reports to our mobile information system, the system sends the related information to John's mobile device. This information includes a brief summary of emergent situation in that area, an alternative route to bypass the area, and locations of closest points of interest to this emergency (e.g. rescue center, hospital, etc). Based on the fact that John is driving, which the system can detect by John's travelling speed, the system will provide the driving routes to the above points of interest. Also the system knows John is driving towards the west direction, so it only selects the places that are on the west to John's current location. In addition, the system knows John is using a high-profile smartphone that has dual processors and extra memory, so it sends the points of interests in a 3D-format map that is very intuitive and easy to read, and sends the driving routes in an audio file that John can directly open and listen to.

Second scenario is about Cynthia who is walking in the New York metropolitan area for an important appointment. An emergency in the same area has been reported to our mobile information system which then begins to send alert information to all the registered users in this area, including Cynthia. The initial alert includes the emergency description and the locations of closet points of interests. As the system detects Cynthia is walking based on her traveling speed, it sends her the routes to leave this area by taking public transportations. After then, Cynthia finds out that the routing instructions that she receive will not get her to the appointment that she must attend, so she send a query to the system by herself by asking for instructions to get to the place of the appointment. The system calculates the requested routes based on Cynthia's current location and destination, and send her the routes using public transportations. As the system knows Cynthia is using a regular cell phone based on checking the details of the request (i.e. request header, device identifier), it sends the routes in mainly texts with a few low resolution graphics. Also, when Cynthia begins to follow the routes and walk down to metro station, the mobile client detects the network signals of Cynthia's device goes down quickly and then reminds Cynthia that she should save the messages and graphics on the device locally, if not yet, so that she can still view them when completely out of coverage.

In the next section, we describe the system architecture and functional modules of Skyguard, a context-aware information system to serve emergency instructions to users who are in emergent situations and use mobile devices.

IV. SKYGUARD: A MOBILE CONTEXT-AWARE EMERGENCY MANAGEMENT SYSTEM

The aim of this work is to design an information system that provides mobile users with timely and effective emergency instructions based on their environment, personal and device contexts. This section will discuss the overall system architecture and each functional module in details.

Fig. 1 is a high-level view of our proposed system, consisting of a *client request manager* that processes the requests from the mobile client application and route the key information to other modules; an *instruction customization module*, which analyze the contextual information that the

system gathers from users, device, and environment, and customizes the response that matches with these contexts; and a *client application* that sits on the mobile device, captures the user's contexts, send requests to server, and show the response.

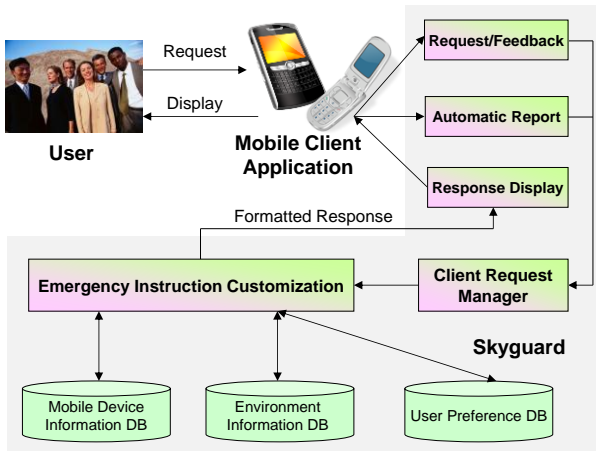


Fig. 1. System architecture.

As shown in the figure, when an emergency is detected, the mobile client application sends the request to the server side in which the first module that receives the request is the client request manager, which will parse the request, filter out the supplementary information (e.g. http protocol header, XML tags), and send the key information (e.g. command, context) to the emergency instructions customization module. The instruction customization module will customize the response for a particular user based on her context information, such as, based on the user's location, heading to the closest safe locations or rescue center. In other words, for a specific user, the instruction customization module will generate a more effective message based on the user's context, such as personal preference about accommodation and food to the previous messages. After then, the instruction customization module will also determine how these messages can be properly displayed on the mobile device based on the device capabilities, such as the screen resolution, orientation and touch-screen support. The rest of this section will discuss each of these modules in greater details.

A. Mobile Client Application

Client application mainly serves four purposes: detect disaster, send request, display response and capture contexts. First, Skyguard system uses two ways to detect a disaster, either by users who can report disasters on the mobile client applications or by various sensors on the mobile devices, which report the disasters when "seeing" the related phenomena, e.g. smoke, speed, location. Second, the device can automatically send a request on the users' behalf, such as requesting for instruction to bypass the emergent area, as long as the user approves this feature in her preference setting, or it will send a request that is initiated by the user, such as looking for routes to a specific location. Third, the device will display the response in the multimedia format that is determined by server and suitable to the device, including video, audio, graphics, text, etc. Last, during the course of an emergency, the mobile client application can

utilize the sensors to capture and send contextual information about the emergency to the server periodically. Below we will discuss what context information can be captured by the sensors that are equipped by most of the latest smartphones:

Audio: the noise level around the user is an important factor to determine the chaotic level of the disaster. High level of environmental noise indicates the necessity of text-based emergency instructions instead of voice-based.

Video: The photos and video clips about the disastrous environment will be very helpful for the emergency management team to understand and analyze the details of the disaster.

Location and direction: Obviously location is one of most important information that emergency management team needs in order to compose the instructions. And the direction that the user headed in can be used to decide the route to a safer place with sufficient network coverage that can guarantee the user to receive further instructions.

Lighting: Level of lighting can be used to determine the environmental conditions of the users, and also determine what devices the rescue team should carry when going to the disaster place.

Movement speed: By tracking the user's average movement speed using the device GPS, the emergency management team can detect the moving abilities of the user and use that to determine how long she can complete traveling the safe route.

Network latency: the level of network latency is measured because the management team can use that to determine the type of emergency instructions that will be sent to the user: more content-rich messages (e.g. audio, video, and graph) can be sent to the user with low latency, and text-only messages will better help the user with high latency. Certainly if possible the management team should instruct the user with high latency to switch locations for better communication.

After all these user contexts have been captured by the mobile client application and received by the client request manager, we can start to build the user context databases that store the collected context information and then provide them to the instruction customization module.

B. Client Request Manager

This module interacts with the mobile client application by extracting the key information from the client request and filtering out other supplementary information. As the request is sent in XML format using http protocol, the supplementary information may include XML tags and http protocol headers. The key information includes user's command (e.g. route to a specific destination) and all applicable contexts. All the key information is sent to the instruction customization module which uses the contextual information to produce customized instructions. The contextual information includes the contexts from the request and the contexts from the server databases.

C. Context Databases

Skyguard system considers three kinds of contexts: disaster environment, user profile, and mobile device. Each

kind of context is stored in a separate database respectively as shown in Fig. 1.

Disaster environment database includes most of the information that sensors can capture, such as location, direction, lighting, speed, etc. This information can help the emergency management team compose and select most helpful instructions, such as with closest rescue location, correct direction, and places with better lighting.

Device information database stores the mobile device features and capabilities, including device manufacturer, model, touchscreen support, screen resolution, equipped keyboard, portrait/landscape, memory size, multimedia rendering, and Bluetooth connection. Device features and capabilities differ vastly amongst various manufactures and models. This information can be identified by the web server when receiving a request and analyzing the request header. And this information will be used to present the users with the emergency instructions in best format for the device.

User profile database stores the profile information that can be usually captured in a user registration, such as user home location, email address, phone number, age group, newsletter preference, automatic request sending, etc. When a new user registers himself in the system, he logs on to the application server and enters his preferences (e.g., restaurant preferences). These preferences are stored in the user profile database.

After these three databases are built, they can be constantly updated when the emergency management system processes the new user requests. Upon receiving a new request, the server can detect the user identity by checking the login authority and device cookie. Once the user identity is determined, the web server can retrieve the user profile and previous feedback from respective databases, and also read the disaster environment and mobile device information from the request, altogether to determine the most relevant instructions for the users.

D. Instruction Customization Module

With the variety of contextual information available from the mobile client application and context databases, Skyguard make proper customizations on the emergency instructions before sending them to the users, in order to provide the users with most effective instructions. The customization we have made in this work includes:

1) Environment customization: Skyguard will track the user's current location, time, speed and traveling direction, and provide the user with information relevant to her environment (e.g. a close hotel in user's traveling direction).

2) Device customization: based on the device features and capabilities, Skyguard will customize the media format and information content of the emergency instructions that is suitable to the device's hardware and software platform (e.g. high-definition video for an advanced smartphone with a high-resolution screen).

3) User customization: Skyguard will also consider the user's preference (e.g. user prefers public transportation than driving, or user likes a specific brand of hotel).

E. Environmental Customization

Before sending an emergency instruction to the user,

Skyguard will first check whether the instructions have location dependent information (e.g. a rescue place in a specific location), or only has location independent information (e.g. description of the emergency event). The location-independent information will be directly sent to the user. The location-dependent information will be first matched with the location of the user's mobile device, which should be on the same direction as the user travels and should be close to user's current location considering her speed. Also the time of day may determine the emergency instructions if the place of interest is only open for a specific time frame.

In addition, Skyguard can detect user's situation based on her environment information and the make further customization on the emergency instructions, such as when user's speed is slow than a certain threshold, the system can assume she is walking and only provide routes using public transportations.

F. Personal Customization

As a context-aware mobile information system, Skyguard considers user contexts when responding to user's requests, and these user contexts in this case mainly refer to personal preference to the information related to the emergency. Generally user contexts are considered as critical determinants in context-aware computing. They should be carefully analyzed and used for individual customization.

Skyguard uses XML to record user's preference (see Listing 1) and when the emergency instructions are ready, the preference will be used to match with the applicable information in order to parse out the effective information. Listing 2 shows the places of interest in an emergency instruction and Listing 3 shows the parsed result.

Listing 1:

```
<Preference><Hotel importance = "low" > Hilton </Hotel>
<Hotel>Intercontinental</Hotel><Hotel>Marriott</Hotel>
<DistanceMax importance = "medium" >5</DistanceMax>
<PriceMax importance = "high" >200</ PriceMax>
</Preference>
```

Listing 2:

```
<Hotel><Brand>Country Inn</Brand>
<DistanceinMiles>2.3</ DistanceinMiles >
<Price>80<Price></Hotel>
<Hotel><Brand>Intercontinental</Brand>
<DistanceinMiles>4</ DistanceinMiles >
<Price>230<Price></Hotel>
<Hotel><Brand>Marriott</Brand>
<DistanceinMiles>4</ DistanceinMiles >
<Price>20<Price></Hotel>
```

Listing 3 (based on match between user preferences in Listing 1 and results in Listing 2):

```
<Hotel><Brand>Marriott</Brand>
<DistanceinMiles>4</ DistanceinMiles >
<Price>20<Price></Hotel>
<Hotel><Brand>Country Inn</Brand>
<DistanceinMiles>2.3</ DistanceinMiles >
<Price>80<Price></Hotel>
<Hotel><Brand>Intercontinental</Brand>
<DistanceinMiles>4</ DistanceinMiles >
<Price>230<Price></Hotel>
```

G. Device Customization

Emergency instructions can be displayed in various formats on a mobile device as simple text links, colorful images, or interactive audio/video. The size of a mobile web page should be much smaller compared to an Internet web page in order to reduce the download time and to fit the page to the small screen of mobile devices. A multimedia-based message is more intuitive and instructional than text-based, but also takes longer to download and requires heavy processing power.

Based on all such considerations, instruction customization module uses mobile device information, disaster environment information and user profile to define heuristics to determine which kind of information presentation are more appropriate for the current user on the mobile device. A typical emergency instruction in Skyguard can include text taglines, animated image, voice message, instructional videos, etc. Below we describe the heuristics that Skyguard uses:

First, device information is a key factor in this module because the capabilities of mobile devices vary significantly. The latest generation of mobile devices is usually equipped with a big touch screen, and thus can show more content on the same page without interrupting the user. This is compared to old devices with a smaller screen, and the user can only scroll the page by repeatedly pressing navigational keys. Audio and video based messages could be more intuitive and helpful for users to follow the instructions, and these messages also depend on latest device capabilities to be properly displayed.

Second, user profile is also important to select the presentation type of response message. For example, if the user set video messages as the preferred format in registration, video should be used in sending the messaging. On another hand, if the user profile indicates she only has a fundamental cell phone (e.g. without any audio/video player), the text message should be the only format.

Last but not least, emergency environment information should also be considered when selecting the proper message format. For example, if the lighting condition is very limited in the user's environment, the instructions should be mostly in audio format, instead of text and/or video.

V. PRELIMINARY DESIGN OF CASE STUDIES

Based on the system described above, the authors conduct several preliminary case studies to evaluate the performance of Skyguard system. In these studies, we mainly consider two parameters: efficiency and effectiveness. Efficiency can be measured by the time of the roundtrip from the client application sends the request and receives the response. Effectiveness can be measured by checking the relevance between user's needs and emergency instructions, obviously high relevance indicating better effectiveness.

Assuming users' contexts are clearly and accurately captured, the emergency instructions that Skyguard generates and customizes using contextual information can be more effectively than randomly selected messages. Therefore, in the experiments, the authors focus on checking the timeliness of the system to make sure the users can still

receive instructions in a timely manner even the system has taken an extra step to achieve better effectiveness.

The experiments were conducted from a strict user perspective as mentioned, which determined the delay between the server receiving the request from mobile client and the reception of instructions on the mobile client, and then compared the delay with and without applying the step of selecting emergency instructions based on user contexts. In these experiments, the authors have tested different types of messages, including text (up to 140 bytes), image (variable, less than 100KB in the tests), audio (less than 4MB in the tests) and video (less than 7MB in the tests), each with a considerable number of times.

The results from these experiments revealed that for small text messages, the total delay between when a client request is sent out and when the instruction is received by the client, is around 150 seconds in average without analyzing and using the contexts for selecting better matched instructions, around 158 seconds in average if analyzing and using the contexts, which means a 5.3% increase in the delay. For image-based instructions, there is a 3.2% increase. For audio and video based instructions, the increase percentage is less than 1%. Safely we can state that the larger the data size of the instruction is, the less delay that is caused by the application of our approach. The reason behind that is evident: the larger data needs a longer time to be transmitted, while the delay caused by contexts processing, which is already minimal even compared to the transmission time for the smallest type of instructions (i.e. text), is often the same for either large or small data since context analysis and instruction customization is independent of the data size of the response itself, and so the percentage of the delay in the entire transmission time decreases while the data size increases.

As such, we can conclude that this approach of analyzing and using user contexts to determine most relevant emergency instructions is more suitable for the users who carry advanced mobile devices and prefer multimedia-based messages (audio/video), because the extra delay by using the approach is very minimal compare to the transmission time, and, on another hand, using the approach to select a most suitable instructions will be more helpful for these users because otherwise they may wait for a long transmission time yet get a less useful message.

VI. CONCLUSION

In this paper, we have proposed a mobile information system, namely Skyguard, which analyzes various contexts and customizes relevant emergency instructions. We classified mobile application contexts into three categories according to their characteristics, i.e. environment context, device context, and user context, and then discussed the benefits and challenges of adapting these contexts to the instruction customization and presentation process. In this paper, we have discussed the main modules of Skyguard, including mobile client application, client request manager context databases, and instruction customization module. Our future work is to further evaluate the mobile information system and research approach, for which we will need much more usage data from real-life cases to

validate whether the customized emergency instructions match users' real cases. In addition, we need to outline how to protect privacy when using and analyzing users' preferences and activities of the mobile users. On the theoretical side, we will focus on capturing more variety of contextual information using mobile client applications and developing more intelligent context-aware instruction customization algorithms on the server side. Also it would be interesting to look specifically into different types of contexts, including social (e.g. community) and technical (e.g. device), and consider the impact these contexts may bring to interaction amongst the user, application and emergency situations.

REFERENCES

- [1] E. Koblenz, *The Evolution of the PDA 1975-1995*, Technology Rewind, 2005.
- [2] M. J. Tooley and A. Mayo, "Handheld technologies in a clinical setting: State of the technology and resources," *Critical Care Nurse (Suppl.)*, 2004.
- [3] A. Meissner, T. Luckenbach, T. Risse, T. Kirste, and H. Kirchner, "Design Challenges for an integrated disaster management communication and information system," in *Proc. the First IEEE Workshop on Disaster Recovery Networks*, New York, USA, June 2002.
- [4] A. Dey, "Understanding and using context," *Personal and Ubiquitous Computing*, vol. 5, no. 1, pp. 4-7, February 2001.
- [5] G. Kappel, W. Retschitzegger, E. Kimmerstorfer, B. Pröll, W. Schwinger and T. Hofer, "Towards a generic customisation model for ubiquitous web applications," in *Proc. 2nd International Workshop on Web Oriented Software Technology*, Málaga, Spain; 10 June, 2002.
- [6] M. Weiser, *The Computer for the 21st Century*, Scientific American pp. 94-104, September 1991.
- [7] B. Schilit, N. Adams, and R. Want, "Context aware computing applications," in *Proc. IEEE Workshop on Mobile Computing Systems and Applications*, Santa Cruz, California, pp. 85-90, December 1994.
- [8] A. Schmidt, K. A. Aidoo, A. Takaluoma, U. Tuomela, K. V. Laerhoven, and W. V. de Velde, "Advanced interaction in context," in *Proc. the first International Symposium on Handheld and Ubiquitous Computing*, Karlsruhe, Germany, September 1999, pp. 89-101.
- [9] M. Khedr and A. Karmouch, "ACAI: Agent-based context-aware infrastructure for spontaneous applications," *Journal of Network and Computer Applications*, pp. 19-44, 2005.
- [10] G. Biegel and V. Cahill, "A framework for developing mobile, context-aware applications," in *Proc. the Second IEEE Annual Conference on Pervasive Computing and Communications*, PERCOM, 2004.
- [11] T. Gu *et al.*, "An ontology-based context model in intelligent environments," in *Proc. Communication Networks and Distributed Systems Modeling and Simulation Conference*, 2004.
- [12] T. Nakatsu, K. Murakami, and H. Sakai, "Context-aware information provision to the mobile phone standby screen," in *Proc. 7th International Conference on Mobile Data Management*, 2006, p. 43.
- [13] Z. Yu, X. Zhou, Z. Yu, D. Zhang, and C.-Y. Chin, "An OSGI-based infrastructure for context-aware multimedia services," *IEEE Communications Magazine*, vol. 44, no. 10, pp. 136-142, 2006.
- [14] R. Padilha, J. Gomes, M. Borges, and J. Canos, "The design of collaboration support between command and operation teams during emergency response," in *Proc. the 2010 14th International Conference on Computer Supported Collaborative Work in Design*, pp. 759-763, 14-16, April 2010.
- [15] Barnard *et al.*, "Capturing the effects of context on human performance in mobile computing systems," *Personal and Ubiquitous Computing*, vol. 11, issue 2, pp. 81-96, January 2007.



Nan Jing received the bachelor degree from Peking University, Beijing, China, and the master and Ph.D. degrees from the University of Southern California, Los Angeles, all in computer science. He is currently a postdoctoral researcher at Harbin Institute of Technology (HIT) where his research focuses on information management and intelligent systems. Prior to HIT, he was a senior staff with the R&D Department, Bloomberg L.P., New York, NY, where he conducted research on mobile software platforms and applications for financial services.

He is the author or co-author of a number of papers and one book in software design, web service, decision science, and business process management. He is a frequent reviewer for conferences and journals that deal with information system and mobile technologies. He is also on the editorial review boards of the International Journal of Handheld Computing Research and the International Journal of Software Engineering and Knowledge Engineering.