Camera-based Context-aware Smart Object Controller *

Hyoseok Yoon †, Hyejin Kim †, Choonsung Shin †, Woojin Jung †, Woontack Woo †

Abstract

In this video, we introduce a scenario with a prototype of camera-based context-aware smart object controller for a family in smart home. The smart object controller is a context-based user interface to exploit context of both user and environment. It uses context of user to personalize control interface. Environment context is used to discover services and used to check service conflict among users in smart home. It has a camera-based architecture that captures an image to recognize smart object and generates user interface. Through generated user interface, users can directly manipulate functions of various smart object and services as as universal controller. Furthermore it enhances handover of controllability between users to resolve service conflict in a multi-user scenario.

1. Introduction

In future computing paradigms such as ubiquitous and pervasive computing, thousands of smart objects and services are around users to serve them. These pervasive smart objects and services serve users in personalized and context-aware manners. However, most of earlier works focused on building intelligence in the environment side. The environment was in control of triggering services, monitoring users and storing personal information of users in a centralized server. In this video, we show our approach based on a mobile device for users to regain controls in ubiquitous and pervasive computing environments. In this video demo, we present an idea of camera-based context-aware smart object controller in a smart home scenario.

2. Camera-based Context-aware Smart Object Controller

Our approach utilizes mobile device platforms such as PDA and UMPC platforms. Mobile devices add mobility and functionality to users as a practical form of wearable computing. Since mobile devices are very easy to acquire and use these days, we utilize mobile devices as an interaction tool. A notable feature of today's mobile device is a built-in internal camera. The increased functionality and popular acceptance behind mobile camera is that it gives users a sense of control such as a sense of control of time and moment by taking and keeping pictures of memorable moments. We further enhance this idea to design our smart object controller. The Figure 1 shows the overall architecture of camera-based context-aware smart object controller system.

---

* This research is supported by the Ubiquitous Computing and Network (UCN) Project, the Ministry of Information and Communication (MIC) 21st Century Frontier R&D Program in Korea.

† Authors are with GIST U-VR Lab., Gwangju 500-712, South Korea, corresponding author email: wwoo@gist.ac.kr

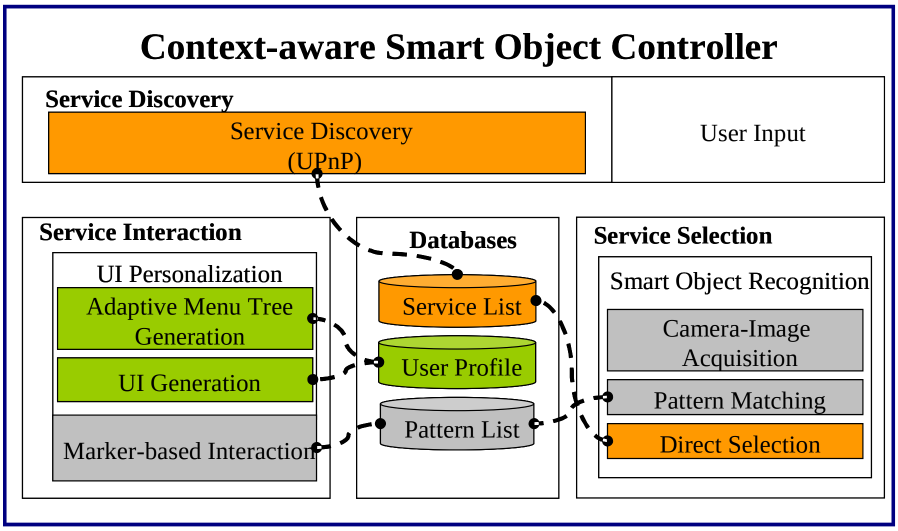

Figure 1. Architecture of Context-aware Smart Object Controller

2.1. Service Discovery and Universal Control

A distinguished characteristic of ubiquitous computing environment is ubiquity of services. Service discovery notifies users for available services in the environment and universal control mechanism enables those services to be controlled through a unified control interface. Since users are able to move in and out of environments, there is a need for certain mechanism to tell users what kind of services are provided and what can they do with it. These requirements can be met by implementing service discovery protocols such as UPnP [1]. However, UPnP itself does not support context-awareness; therefore we concurrently use it with our context-aware application framework, ubi-UCAM [1] to utilize context information in both UPnP service and UPnP control point.

2.2. Service Selection

There are two possible selection methods, list-based and camera-based. In list-based selection method, service information stored in a service database is provided to the user as a list. In this step, icons or tab controls of menu which visually symbolize each service are used. Through this association, users can intuitively know presence of physical devices, multiple devices and logical but invisible services. Second method is camera-based selection method. Since it is time-consuming and difficult to select from a visualized long list of services in one screen, therefore we introduce camera-based method which uses user's explicit command. When the user takes a picture with personal companion, acquired image is compared with patterns in pattern database to identify the object in the picture. A similar concept is popularized and used in many other works [3][4].

2.3. Service Personalization

After a service is selected through discovery and selection steps, a user interface to control and access functions and contents of service is provided. Personalization takes place as a filtering mechanism to tailor services to users. It is impractical to provide all functions and contents to user for a given moment, since the screen on mobile device is limited and users have certain usage patterns. Therefore

[1] UPnP, http://www.upnp.org

we utilize preference of contents and functions by giving higher preference items a higher priority. As a result, higher preference items are listed first and placed in top menu for easy accessibility. This way, personal companion can recommend and guide user's next action. Personalization is classified into two categories. First category is personalization in functions and contents of a service, so different users have different levels of control and accessibility. Second category is personalization in appearance of user interface. In this category, user's preference is used to construct appearance of user interface. Our system supports both categories of personalization. For personalizing functions and contents, it filters based on service genre and for appearance personalization; it customizes colors, fonts, and menu styles. All these are made possible by exploiting user profile.

2.4. Service Interaction

When a personalized user interface is generated, the user can interact with services through camera-based interaction. Our system provides various interaction possibilities using a camera on the mobile device. First example is controlling smart object after recognition, so it acts as a universal remote controller to discover multiple services and control. Another example includes an interaction method with marker-embedded contents where a user can navigate through virtual reality environment by tracking relative distance and angle between marker and camera. Here, mobile device can be used as an input device for giving directions.

3. Demo Scenario

To demonstrate the proposed camera-based context-aware smart object controller, we have sketched four scenes to show how smart objects are controlled and how the mobile device is used to interact with pervasive services.

Scene 1: Controlling ubiLight (Smart Light)

Mother enters into a living room where she is greeted with automatically-lit blue light. Even though blue is her favorite and preferred color, she decides to try another color for a fun. She takes a picture of a light switch and turns on green light and finally red light.

Scene 2: Controlling ubiTV (Smart TV)

When mother sits on a couch, ubiTV detects a user's presence and changes its screen saver into a marker. Mother then takes a picture of this marker to get a personalized user interface where her favorite drama and information channels are sorted on top. She turns on a drama channel and watches for awhile and switches to information channel.

Scene 3: Controlling MRWindow (Smart Window)

When mother moves toward MRWindow, MRWindow detects a user's presence and changes its screen saver into a marker. Mother tries picture view service and navigates through virtual reality world for fun. Here she manipulates her mobile device up, down, right and left to navigate.

Scene 4: Control Hand-Over

A son watches baseball game in the afternoon alone. When mother joins him, a service conflict occurs due to difference in preferred channels. Even though mother may have a higher controllability than *sons, it is impractical to automatically switch to drama channels. Following FCFS (First Come First Served), the son still has control. When a service conflict arises, the son is notified of the conflict, and then he can choose to hand-over control. When he hands over the control, ubiTV displays a contents-embedded marker so mother can use it as an entry point. Mother takes a picture of this marker and turns on a drama channel.*

4. Discussion

The proposed system is implemented on PDA and UMPC platform respectively. Earlier prototype is developed for PDA using Pocket PC 2003, and then it is also tested on PDA with Windows Mobile 5.0. The version used in the video is based on UMPC platform. This version has higher performance compared to PDA version. To test the smart object controller, we have experimented in ubiHome testbed [2] with various smart objects.

5. References

[1] Yoosoo Oh, Choonsung Shin, Seiie Jang and Woontack Woo. ubi-UCAM 2.0: A Unified Context-aware Application Model for Ubiquitous Computing Environments. In *Proc. UbiCNS 2005*, June 2005. Jeju, Korea.

[2] Yoosoo Oh and Choonsung Shin and Woojin Jung and Woontack Woo. The ubiTV application for a Familty in ubiHome. In Proc. The *Proc. The 2nd International Ubiquitous Home Workshop*, pages 23-32, 2005.

[3] Michael Rohs and Philipp Zweifel. A Conceptual Framework for Camera Phone-based Interaction Techniques. In *Proc. Pervasive 2005*, pages 171-189, 2005.

[4] Genta Suzuki, Shun Aoki, Takeshi Iwamoto, Daisuke Maruyama, Takuya Koda, Naohiko Kohtake, Kazunori Takashio and Hideyuki Tokuda. u-Photo: Interacting with Pervasive Services Using Digital Still Images. In *Proc. Pervasive 2005*, pages 190-207, 2005.