設計及製作介面系統產生器用於整合連結前端認知辨識系統及後端應用軟體系統(2/3)

The design and Implementation of Interface Interfacing generator for integrating and bridging front-end recognizers and back-end application software systems (2/3)

計畫編號：**NSC**97-2221-E-009-062-MY3

執行期限：97 年08 月01 日至100 年07 月31 日

主持人：陳登吉交通大學資訊工程系教授

成果報告目錄

設計及製作介面系統產生器用於整合連結前端認知辨識系統及後端應用軟體系統(2/3)

The design and Implementation of Interface Interfacing generator for integrating and bridging front-end recognizers and back-end application software systems (2/3)

計畫編號：**NSC**97-2221-E-009-062-MY3

執行期限：97 年08 月01 日至100 年07 月31 日

主持人：陳登吉交通大學資訊工程系教授

一、 計畫緣由與目的

In a Windows environment, the commonly used traditional method, which allows developed window application programs with Human Computer Interaction (HCI) control ability, is directly to write the control procedures into the application programs while using low-order designing formula to package procedures into single application system. To apply such devising method, the designer must possess certain knowledge about application system designing and programming in order to devise an application system with HCI control functions. Particularly when the design is completed, it is relatively difficult to revise or add any system functions to it without the primitive code, as shown in figure 1.

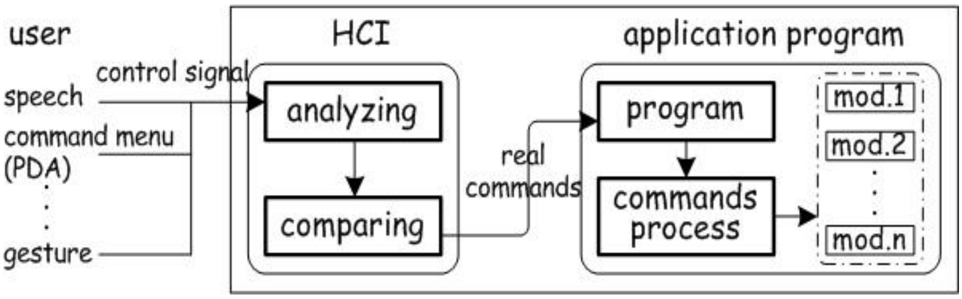

Figure 1 General developing method of HCI control

Three major problems may exist under such development of HCI control procedures. First, system designers must be equipped with abundant knowledge about the design of HCI and programming languages in order to design an application with HCI function. Second, if we want to design an application, which lacked interaction ability before, we need to obtain the primitive code of the particular application due to the difficulty in modifying new programs without the code. Third, even if we have obtained the code, we need to re-analyze the entire structure of application in order to write a suitable control program. These tasks will leave the designer with much trouble and seemingly resulting in less flexibility and efficiency.

To overcome these issues, we emphasize on the research of Software Engineer Methodology to develop a visual generic interface bridge (GIB) system and introducing this system into two parts: First, the "Integration of GIB and Speech HCI," and secondly, "GIB-based Application Interface (GAI) generation," in which a PDA device is taken as an example. The GIB provides visual operating interface, under which designers draw recognizing square object at any corresponding position on the windows and name each square object. Subsequently, we can easily use speech command to control mouse and keyboard actions corresponding to the position of square object. By increasing the operation of application with more flexibility and expandability, we use macro command to define and combine the control commands. One macro command may be combined with several control commands; this will avoid noise effect between long commands and make the application control more flexible with grammar analysis technology. Through this process we can make any application, which did not have HCI control ability previously, with speech or wireless remote HCI control functions in an easier and more efficient manner and do not need to write any program code, as shown in figure 2.

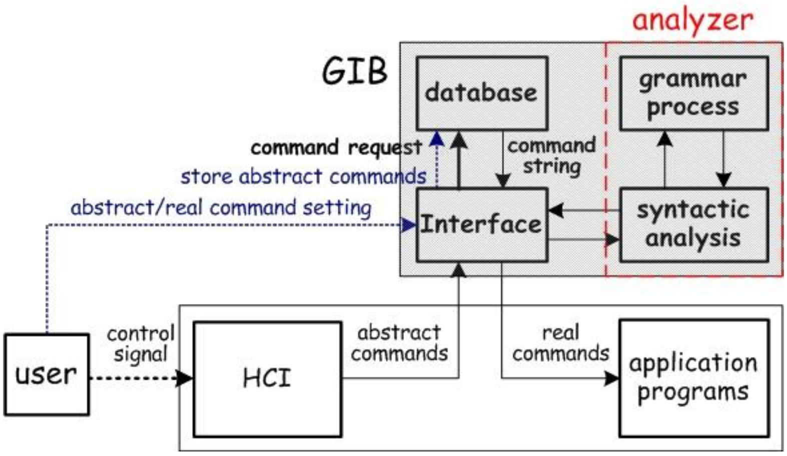

Figure 2 Architecture of GIB control system

二、　報告內容

In the following, we present the 2nd year research results of this three years project.

2.1. Introduction

As it is known, we could conveniently and directly operate ordinary user interface by hand controller. But along with progressing existing techniques and increasing functions of communication equipments for family use, operation and interaction styles between people and device become more and more complex. Most of the multimedia contents can be run and displayed on different kinds of platforms without possessing remote control ability originally, so, people think if they can just use some simple instruments, such as cellular phone or PDA, to remotely control the multimedia application module running on the PC or digital TV, then the control would become vivid and interesting. But because there are various control instruments, display devices, and different kinds of methods, it is not easy to make a specified device to have remote control ability. To do this, we must repeat the design process: (1.) Write control command protocol and HTTP wireless protocol into the applications running on the PC; (2.) Write interface program and wireless control protocol into cellular phone, as in figure 1, for every application system one after another. But this designing process is hard work; we must write many complex procedures even for just adding or modifying a new control function. After that, there still remains a big problem that if we do not have the original source program of application system, it could become impossible to make the specified device have remote control ability. Such repeated design procedures would make developing application systems a cost burden, waste of time, inelastic, and inefficient.

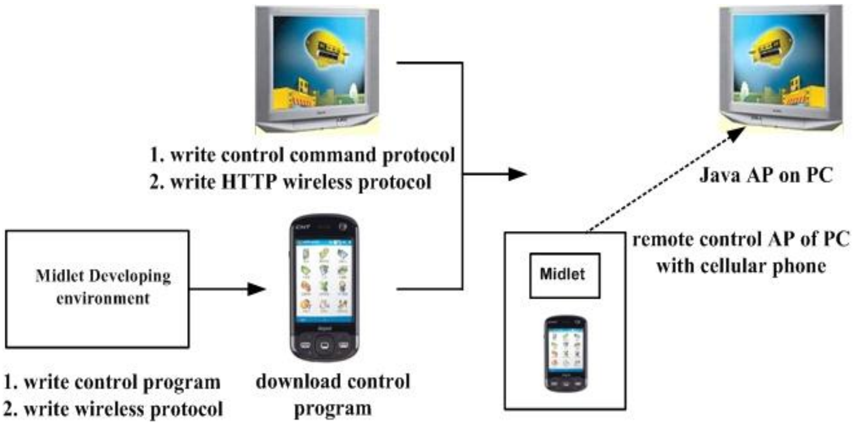
1. write control command protocol
2. write HTTP wireless protocol
Java AP on PC
Midlet Developing environment
Midlet
remote control AP of PC with cellular phone
1. write control program
2. write wireless protocol
download control program

Figure 1 General developing method

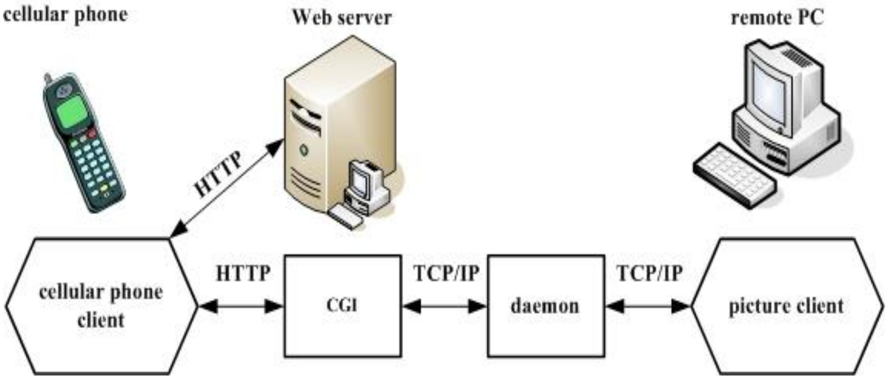

Figure 2 Rajicon System Architecture

2.2. Related Work

Remote control methods have been discussed recently for many years; in 2001, it [1] was proposed that cellular phone be used to control remote GUI running on the PC. Cellular phone is a mobile device possessing a small screen but cannot show the same GUI as on the PC screen. So we must analyze GUI and develop some function programs in the cellular phone for its use in the remote control interaction with GUI on the PC. In view of remote controller, client-server is exactly a simple framework. At first, the client sends request to the server through OTA (on the air). After receiving this, the server parses the command and accesses it, then sends the result response back to the client. This interact agreement is developed for a specific device; for establishing this, we must consider about the device operating ability, whether it supports multimedia content and relative remote control function. When the interaction functions aimed at become more and more continual, the remote control interface and interaction protocol development would become more complex and difficult.

Rajicon System [2] was developed in 2002. Its main function is using cellular phone to control the remote PC, as in figure 2. It works on a specific device, so, if we want to do some complex operation functions, we must define a set of interactive rules. Rajicon System uses macro command to formulate every operating function, so, the more operating functions there are, the more macro commands it would need.

Through pressing keypad of cellular phone in order to input command to control application running on the PC, these operation methods are complex and undirected, and the control commands are not easy to memorize. On the other hand, this system is designed for a specific device, if there are some control functions to be created or modified; such designed procedures should be restarted repeatedly.

In 2002, Jeffery Nichols and Brad A. Myers issued a topic ‘Generating remote control interfaces for complex appliances’ [3]. They proposed a method of personal universal control (PUC) and the document content is about how to create a control interface with graphics or speech by downloading a functional specification explanation of equipment for family use, as in figure 3. After that, PUC system would analyze specification document and use decision tree algorithm [4] to create a specification group tree, as in figure 4, then according to this group tree structure to establish an appropriate control interface. But there are many design factors needed to be considered, including: (1.) Before establishing an interface, it had to download a functional specification explanation for a specific equipment, but this specification description is not easy to establish; (2.) It is tedious to design a control interface, because interface designed algorithm need to consider user’s requirement with the presentation of control device objects.

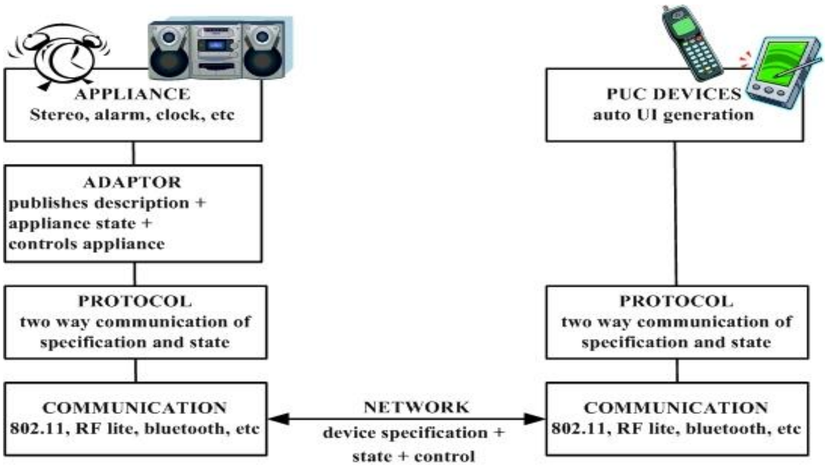

Figure 3 PUC System Architecture

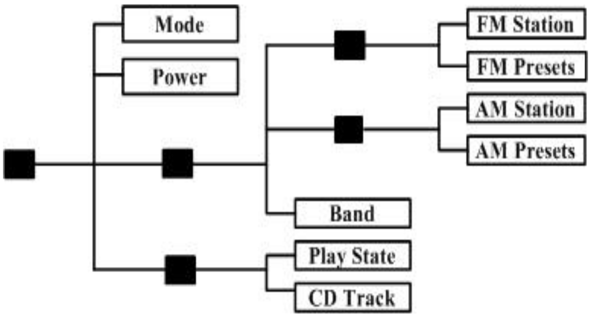

Figure 4 A sample group tree for a shelf stereo

In [5], because traditional remote control typically allows users to activate functionality of a single device, they present qualitative and quantitative results from a study of two promising approaches creating such a remote control: end-user programming and machine learning. In end-user programming, users manually assign the buttons they believe are sufficient to accomplish their tasks to graphical ‘screens’. They then work with a single, handheld remote control that can display those screens. Machine learning also uses a single, handheld remote to display screens; however, it uses the recorded history of a user’s actual remote interactions to infer appropriate groups of buttons for the performed tasks. There may be some questions in the following [5]: (1.) The end-user programming will be very complicated while carrying on the design of remote control to the device with graphical screens. User must define the components needed in operating on a screen that can control many device interfaces remotely at the same time, and all the control procedures need a large number of programs to be written; (2.) The ML can record the operation of user automatically and utilizes the performing algorithm to produce the operating component. Perhaps because of the difference of device, it cannot completely define each function of component of the device; (3.) The method of automatically producing remote control interface needs to accord to the characteristic of the device and the favorite of user, but it also needs a large number of procedures to be written in order to design algorithm automatically. There are some merits with adopting our interface generating system: (1.) the designer can simplify program and design the complicated control procedures that have the characteristics of code reusable by using interface–generating system to define the object and generate operating interface automatically; (2.) Analyze the objects that users defined automatically, and produce procedure control and corresponding remote control operation interface; (3.) Standardizes the way that the user defines the object, to help user to produce the operation procedures and interface on remote control with the device. Such procedure does not have the need to write complicated mathematical calculations or design under a specific interface.

In [6], they described a new widget and interaction technique, known as a “Frisbee,” for interacting with areas of a large display that is difficult or impossible to access directly. It consists of a local “telescope” and a remote “target”. This design satisfies five design principles of: (1) minimizing physical travel, (2) supporting multiple concurrent users, (3) minimizing visual disruption while working, (4) maintaining visual persistence of space, and (5) application independence. However, it needs to write a large number of procedures for the specific and large-scale showing device. Moreover, because of the limitation of hardware specification and the difficulty in obtaining the relevant information, it is comparatively difficult and complicated to design the interface. Our interface generating system could automatically produce remote control system and relevant operating method; it will not be limited to the specification of the hardware, and it can offer many different devices to carry on remote control interaction.

Along with the functions of periphery equipment gradually increased, the design of remote control interface in the cellular phone has become more and more complex and difficult. In this paper, we hope to construct a remote control interface generating system. Through that, designer can easily transform control objects of JAVA application program running on the PC, such as control buttons or labels, into cellular phone automatically. Finally, using the keypad or touching panel of cellular phone controls the JAVA applications running on the PC directly and easily. The key functions of this interface system include: (1.) It can wrap operate mode and static endue each operation item a control command, so there is no need to define complex operating instructions; (2.) It can analyze operating objects of JAVA applications, and produce description file to the interface generator. New operating functions can be developed quickly and effectively to generate interface operating program into the cellular phone; 3. There are many written kinds of specific abstract procedures of template, and these can be reusable by inheriting. System designer can develop new Java multimedia function and transform new service through inheriting template procedure easily, quickly, and effectively, even add or modify remote control function without needing to know the source program of applications. This will simplify the development process and make the control system development and modification more easy, effective, and flexible.

2.3. Proposed Method and Algorithm

2.3.1. Structure

On the traditional developing of remote control functions into the cellular phone, designer need to write the Midlet of cellular phone at first, and then write the JAVA AP controlling statement including interaction between PC and cellular phone. This designed process is hard work, a waste of time, and inefficient.

Our solution for solving this problem is to construct an interface generating system; with it the designer can easily develop remote control interface program into cellular phone without even needing to write the program of cellular phone. Finally, with using this cellular phone as a remote controller, user can interact with the JAVA application running on the PC directly and easily. The framework of remote control interface system is as in figure 5.

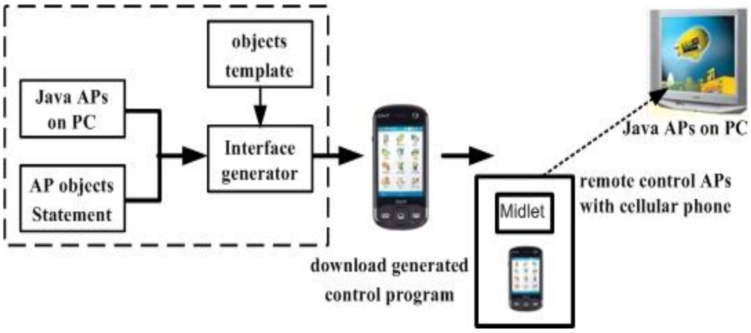

Figure 5 Framework of remote control interface system

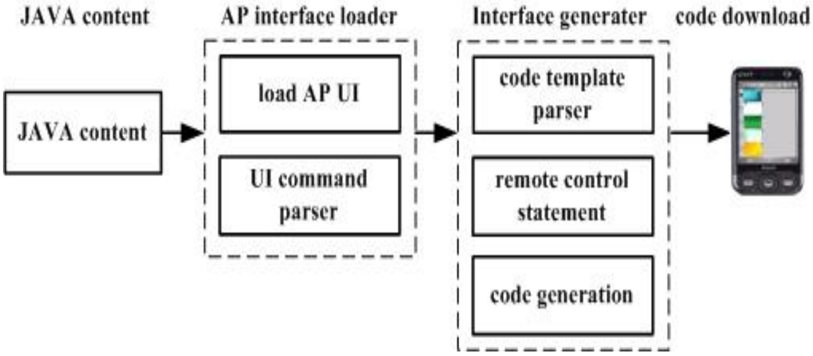

Figure 6 Cellular phone interface generated modules

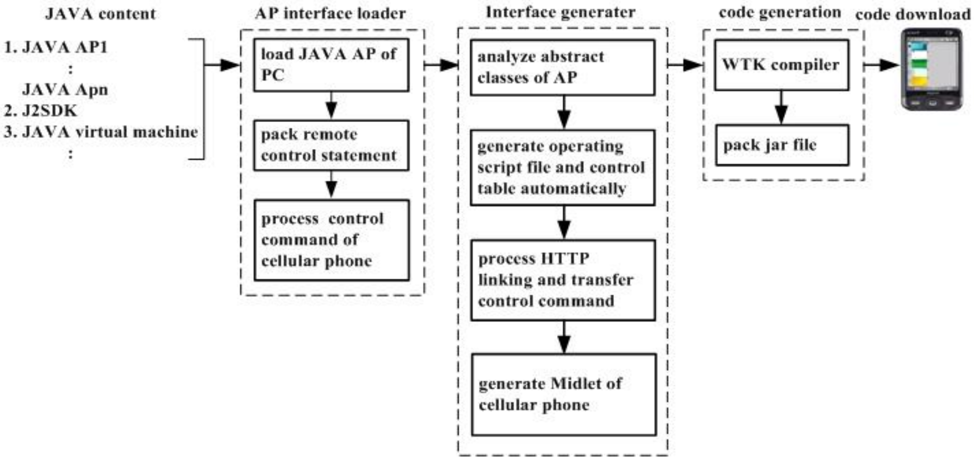
JAVA content
1. JAVA AP1
:
JAVA Apn
2. J2SDK
3. JAVA virtual machine
:
AP interface loader
load JAVA AP of PC
pack remote control statement
process control command of cellular phone
Interface generater
analyze abstract classes of AP
generate operating script file and control table automatically
process HTTP linking and transfer control command
generate Midlet of cellular phone
code generation
WTK compiler
pack jar file
code download

Figure 7 Cellular phone interface generated procedures

The generator modules include JAVA content, AP interface loader, and interface generator, as in figure 6. Module “JAVA content” means the JAVA applications running on the PC to be controlled with cellular phone. Module AP interface loader includes “AP interface loader”, which loads JAVA application into and runs it on the PC, “packs remote control statement”, and “UI command parser” which processes the interface control command of cellular phone. Finally, module “interface generator” includes “code template parser”, which analyzes abstract classes of AP, and according these classes to generate operating script file and control command table, “remote control statement”, which processes HTTP linking and transfer control command, and “code generation”, which generates Midlet of cellular phone automatically according to the operating script file. Finally, it executes the Wireless Toolkit (WTK) compiler and packs program into jar file, then designer downloads this jar file into the cellular phone, as in figure 7.

2.3.2. Algorithm

Control interface systematic framework is composed of application program interface loader, cellular phone interface generator, and a Java application program. Application program interface loader is a graphic user interface (GUI) for designer which helps loading JAVA applications running on the PC into interface generator easily. It is a JAVA server procedure; system user makes JAVA application program template and provides this to the interface loader. After loading application interface

program, it links and transforms operation objects, such as Javax.swing.JButton and Javax.swing.JReadButton, into commands ID, such as ‘001#’ for play_btn_1 and ‘002#’ for play_btn_2.
We have proposed the following algorithms of these modules and the details would be explained in the final report.

2.3.2.1. Algorithm of Application Program Interface Loader

Algorithm 1-1 Loading AP UI

Algorithm 1-2 Parsing UI Command

2.3.2.2. Algorithm of Interface Generator

Algorithm 2-1 Code Generation

After mathematical process of algorithms 1-1 and 1-2, we fully understand the sample version and class definition of Java application program. So users need not develop remote control program on cellular phone, they can easily and directly get remote MIDlet procedure of cellular phone by analyzing program code through interface generator. In algorithm 2-1, MIDlet procedure inherits Form class and CommandListener in order to control the whole mobile interface. We first take charge of loading relative groupware, such as Button, Radio Button, and ComboBox, and analyzes what kinds of operation project are used in application program. It would then generate the main list which belongs to cellular phone interface, such as Button, Radio Button, and ComboBox, which would produce sub-project on cellular phone screen. Finally, we need HTTP connection procedure to send remote control commands from cellular phone to GUI system.

2.4 Conclusion

We overcome some common problems suffered by developers when bridging a remote control interface system of cellular phone with the JAVA applications running on the PC and have created an interface generator system to generate interface automatically into cellular phone, with which system designer can easily develop control interface of JAVA Midlet of cellular phone. For more convenience, we define many kinds of template procedures, such as JButton, JRadioButton, and JComboBox, with which system designer can easily and quickly produce a new serve function. JAVA interface generator of cellular phone is the kernel of proposed system. It can parse abstract class written by the application designer, produce an operation script file and control table according to the template procedures, transfer specified control object of Java application system on the PC, such as control buttons and labels, into cellular phone automatically, and then give every operating object in the control table a command ID, exempt from developing remote agreement on cellular phone repeatedly. Finally it produces and compiles the Midlet program and wraps the result of jar file into cellular phone.

The control interface also combined touch panel and key press operating ability of cellular phone; it can be generated by the interface generator system automatically, even functions of linking and transferring control command protocol. System designer need not define complex macro commands and operating procedures, all the designer has to do is define event procedures when receiving the requested control commands. Through directly pressing interface command object on the touch panel or single keypad of cellular phone, user can remotely control the JAVA application programs running on the PC directly and easily. As we have done, the proposed approach will simplify the development process, abbreviate development time, and make the control system development and modification more flexible and efficient.

We adopt Java Objected-Oriented program analysis based on control patterns as our developing methods, and use that to define the inherited relation and offer the making of the abstract classification to reach the improvement of productivity maintainability.

三、 參考文獻

1. H. Okada, K. Kato, T. Ikegamai, Y. Tatusmi, and T. Asahi, “Proposal of PC Remote Control System by Mobile Devices”, IPSJ Sig Notes (2001-HI-93), 2001(38): 1 6, 2001.

2. Norman Makoto Su, Yutaka Sakane, Masahiko Tsukamoto, Shojiro, Nishio, "Systems Issues: Rajicon remote PC GUI operations via constricted mobile interfaces", Proceedings of the 8th annual international conference on Mobile computing and networking, September 2002.

3. Jeffrey Nichols, Brad A. Myers, Michael Higgins, Joseph Hughes, Thomas K. Harris, Roni Rosenfeld, Mathilde Pignol, "infrastructure for ubicomp: Generating remote control interfaces for complex appliances", Proceedings of the 15th annual ACM symposium on User interface software and technology, October 2002.

4. De Baar, D.J.M.J., Foley, J.D., Mullet, K.E., "Coupling Application Design and User Interface Design", in Conference on Human Factors and Computing Systems, 1992 Monterey, California ACM Press, pp. 259-266.

5. Azam Khan, George Fitzmaurice, Don Almeida, Nicolas Burtnyk, Gordon Kurtenbach, "A remote control interface for large displays", Proceedings of the 17th annual ACM symposium on User interface software and technology, October 2004.

四、 計畫成果自評

The research results from this project have been submitted to conferences and journals for publication. Also, part of the technology developed from this research project has been filed patents application in the territory of Taiwan and the U.S.A. through the IP office of National Chiao Tung University.

4.1 Papers Publication

1) Shih-Jung Peng, Jan Karel Ruzicka and Deng-Jyi Chen, “A Generic and Visual Interfacing Framework for Bridging the Interface between Application Systems and Recognizers,” Journal of Information Science and Engineering, Vol. 22, No.5, September 2006, pp.1077-1091 .(SCI)

2) Deng-Jyi Chen, Shih-Jung Peng and Chin-Eng Ong, “Generate Remote Control Interface Automatically into Cellular Phone for Controlling Applications Running on PC,” Journal of Information Science and Engineering, (2008.09.16. accepted.)

3) Shih-Jung Peng and Deng-Jyi Chen, “A Generic Interface Methodology for Bridging Application Systems and Speech Recognizers,” 2007 International Conference on Information, Communications and Signal Processing (IEEE ICICS2007), 10-13 December, 2007, in Singapore

4) Ming-Jyh Tsai, Deng-Jyi Chen, Chi-Chung Hung and Hao-Chun Lu, 2009, Context Aware and Content Adaptation of Template Based Multimedia Presentation on Handset Devices, Advances in Multimedia Information Processing - PCM 2009, December 15-18, Bangkok, Thailand.

5) Ah-Fur Lai, Deng-Jyi Chen, Web-based Two-tier Diagnostic Test and Remedial Learning Experiment, *International Journal of Distance Education Technologies*, Vol. 8, Issue 1, 2010, pp. 31-55. (published).

4.2 Patents

1) 陳登吉, 彭士榮, 蔣加洛, “介面系統、方法與裝置”, patent No I 299457, 2008.08.01.~2025.12.19. For Taiwan.

2) Deng-Jyi Chen, Shih-Jung Peng, Jan Karel Ruzicka, “Interface System, Method and Apparatus,” patent application No 20070150280A1, 2007.06.28. For USA. (pending)

3) A Generic and Visual Interfacing Framework for Bridging the Interface between Application Systems and Speech Recognizers (USA) inventors: 陳登吉 彭士榮 蔣加洛(pending)