객체지향 프로그래밍 기법에 의한 원격학습도구의 개발

이 효 종†

요 약

세계적으로 급속히 발전하고 있는 웹(WWW) 기술은 원격 학습의 새로운 기회를 인터넷을 통하여 제공하고 있다. 가상 교육 또는 원격 교육 운영에 관한 일부 실험에 관한 성공적인 보고도 나와있다. 웹은 다른 네트워크 도구와 병합하여 원격으로 떨어진 학습자들에게 대화형 학습을 할 수 있는 가상 교실을 생성할 수 있음을 연구하였다. 전자우편, 다자참여 학습보드, 뉴스그룹, 또는 화상회의와 같은 원격 학습 도구의 표준에 관한 필요 사항들을 연구하고, 기타 프로그래밍 언어와 객체지향 프로그래밍 기법을 이용하여 이들 학습 도구들을 개발하였다. 객체지향 프로그래밍을 통하여 개발되는 코드의 내구성, 확장성 및 재활용성을 증가시킬 수 있었다. 이들 도구들 활용하여 개발된 시스템은 교사와 학생, 또는 학생들 간에 상호작용을 허용하여 실질적인 원격 교육 시스템에 사용 될 수 있음을 알 수 있었다.

Development of Distance Learning Tools Based on Object-Oriented Programming Technique

Hyo Jong Lee†

ABSTRACT

The rapidly developing World Wide Web technology provides new opportunities for distance education over the internet. Several successful experiments about cyber education or distance learning have been reported. The Web, when combined with other network tools, can be used to create a virtual classroom to bring together a community of learners for interactive education. Requirements for standard tools for distance learning, such as an electronic mail, a multiparticipant board, newsgroup service and video conference tools have been investigated and implemented based on the object modelling technique using Java programming language. The object oriented programming helps the developed codes maintain robustness, extensibility, and reusability. Implemented systems using these tools have proved to be practical for distance learning with allowed interactions either between instructors and students or between students.

1. Introduction

Changes towards a globalized, fast networking and information-oriented society have been accelerating its rate of development everyday. The education field is also an area that experiences rapid changes with aid of the development of internet technology. Traditional distance teaching or computer aided education (CAE) was merely emitting simple course materials using the electronic mail service. It is still used with people who do not know how to use new technologies such as internet. Recently, with the growth of the internet application, much information is now available on the web, unfortunately, considering distant teaching point of

† 종신회원 : 전북대학교 전자정보공학부 · 공업기술연구소
논문접수 : 2000년 8월 4일, 심사완료 : 2000년 10월 17일

view, it is most of the time only on-line courses using html-based internet browsers[2-4]. In some cases, infrastructures for distance teaching system have been also developed[1].

Several research results have been reported about the virtual university (VU), also called *virtual campus*. As technologies have been developed, teaching methods also are being changed. The World Wide Web (WWW) and other internet-based collaborative tools have significantly enhanced the ability to train and educate electronically. Recently Mioduser et. al.[6] constructed web-based learning environments and evaluated on practical implications of the learning system. A neuro-fuzzy approach is also introduced to implement lesson adaptation in a web-based course. The Web provides significant new functionality in transmitting information to the student and providing forums for exchange.

Some virtual universities are already implemented using the most recent and powerful technologies. When integrated with tools such as listserves, usenet newsgroups, multiparticipant board (MP board) and video teleconferencing, the web can greatly increase students' level of involvement in the training experience. It is clear that the internet, with its ability to connect people and information around the world, is already having a significant impact on education at all levels. The lofty goal of an interconnected global school across remote corners of the world is to adequately educate people based on their needs and qualifications. The power of virtual classrooms can be exploited to teach various subjects classified as technical science or social science, such as natural science, engineering, mathematics, psychology, linguistics, and much more.

A wide range of technological options are available to distance education. They fall into major categories, such as printing of texts or pictures, audio, and video. Although technology plays a key role in the delivery of distance education, educators must remain focused on instructional outcomes, not the technology of delivery. The key to effective distance education system is focusing on the needs of the learners, the requirements of the content, and the constraints faced by the teacher, before selecting a delivery system. Typically, this systematic approach will result in a mix of media, each serving a specific purpose, such as interactive audio or video or textual information or pre-recorded video. Using this integrated approach, one of the educator's tasks is to carefully select the best media among the various technological options.

The term object oriented (OO) programming means a software development approach which used one of OO programming language, such as C++, Smalltalk, or Java. As Berard pointed outberard, merely employing OO programming does not guarantee the best results. In order to achieve the ideal results, OO requirement analysis and OO design must be considered with an appropriate OO programming language. Java language is a true object-oriented language in sense that everything in Java is an object and a descendant from a parental object. Java is a whole environment complete with class hierarchies for networking, GUI design, input/output, and various utilities.

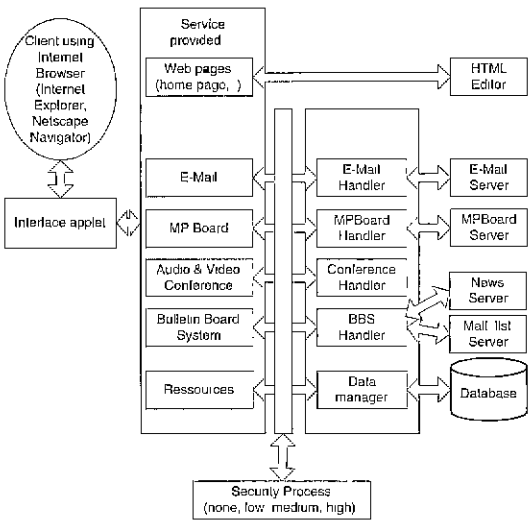
In this paper, requirements for the distance learning tools have been studied, then a system has been implemented using Java language based on the criteria of efficient learning tools. The system has been built as a mix of various instructional media, which meets the needs of the learner in a manner that is instructionally effective and economically prudent. Description is focused only to a mailer and multiparticipant board due to length constraints.

The remainder of this paper is organized as follows. In Section 2, an overall system has been described in detail focusing the mailer and the multiparticipant board. Section 3 presents the implemented system. Finally, Section 4 concludes the distance learning tools with system evaluation.

2. System Architecture

The tools consist of a mail handler, multiparticipant board(MP board), conference handler, news service handler, and a data manager. Each tool can be executed either as stand-alone or as an integrated distance

learning system. The functions of the mail handler are generalized as a common mail system, and can be used for informal one-to-one correspondence. Instructors and students can read messages at their convenience and easily store them for later reference. The multi-participant board is similar to chatting programs, which allow a group of people communicate each other simultaneously. However, MP boards permit instructors and students to communicate multimedia data. A group of people share a board that displays text and graphic images in real time. Video conferencing servers handle motion pictures for instructors and students who are located at different places. (Figure. 1) shows a simplified structure of the distance learning system.



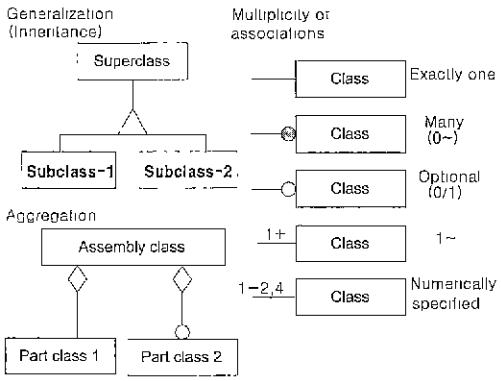
(Figure. 1) Overall system structure

2.1 Mailer

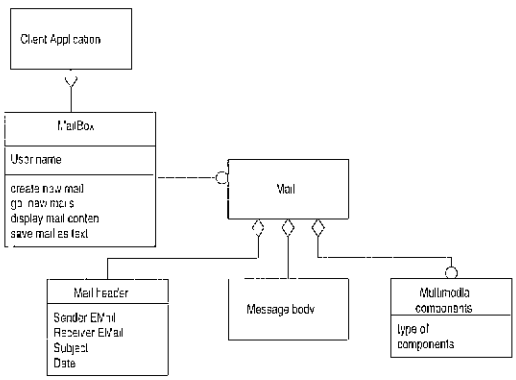
The object classes of mailer are mailbox and mail. The mailbox manages mail with a communication process, a main process and a graphical user interface. It displays only headers for users to check overall mail. The mail consists of mail contents, source addresses and destination addresses.

In order to represent object model clearly, basic concepts of object modelling are graphically defined as depicted in (Figure. 2). The generalization of classes, aggregation and associations are assumed throughout object modelings.

(Figure. 3) demonstrates the mail diagram with attributes and functions.



(Figure 2) Graphical notation of object modelling

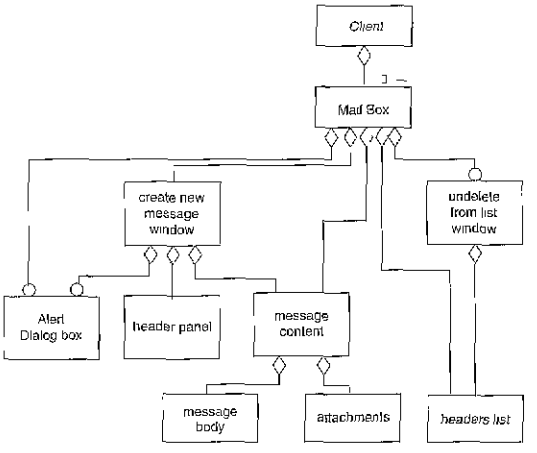


(Figure 3) Mail diagram with attributes and functions

The mail and mailbox are connected with graphical interface so that users can recall them with ease. The detailed object model for the user interface is shown in (Figure 4).

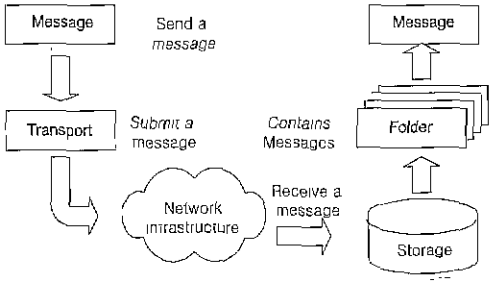
If the mail protocol is not standardized, the tool can transfer mail only inside the system. In order to handle the limited delivery, the IMAP/SMTP7 specification (RFC822, RFC2045) are adapted.

Sent messages are stored in electronic mailboxes until the recipient retrieves them. To see whether any mail has been arrived, a user may need to check his/her



(Figure. 4) Object model of user interface

electronic mailbox periodically. After reading mail, a user can store it in a text file, forward it to other users, or delete it. Copies of memos can be printed out on a printer if a user wants a paper copy. The mail message handling process is depicted in (Figure. 5).



(Figure 5) Message handling process in mail

Basically mail application is divided into four major processes - graphical user interface (GUI) process, main process, listener processor and monitor process. GUI process is commonly found in most object-oriented programs. It is an interface object between the user and the main program. It sends events to the main process when a user interacts with it. Main process receives events from both the interface process and the listener process. It has main algorithm and calls the suitable function depending on arrived events. Listener process has a communication object, i.e., mail package. It does interfacing tasks between the mail server and the mail client application. Monitor process provides the role of a watch-dog, and calls a function from the listener object each time.

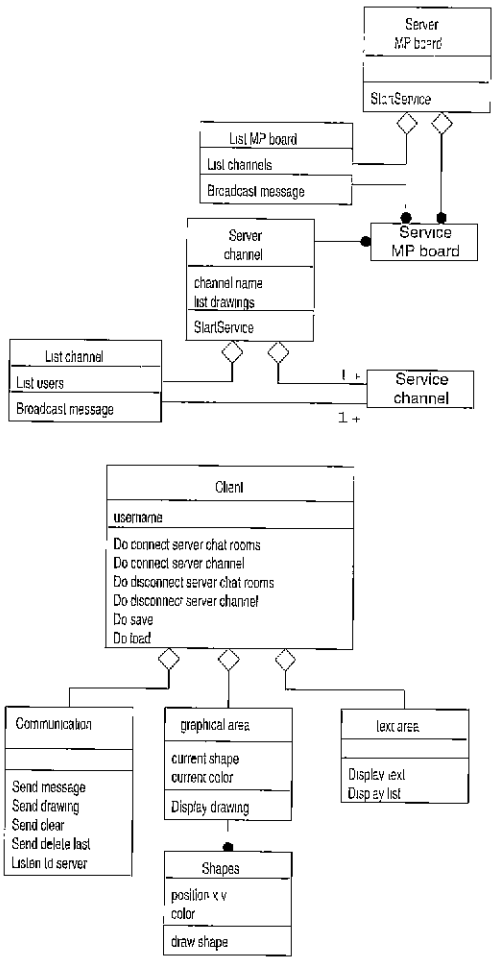
2.2 Multiparticipant board

Talking and *showing* are common activities in the class room. An MP board is a tool providing multiple users, who are separated by different physical locations, live class activities. Each user sends and receives information through a channel. Several channels can be created and users can create, or join to the already-created channels. The list of active members and channels are displayed so that a new user may see who is communicating on the MP board. Users can talk with other members or send graphic images to a specified channel. All members who join the channel can see the updated board simultaneously.

(Figure 6) shows the MP board application with functions and attributes. Client, server channel, shapes, graphical area, text area, and list drawings are defined as objects. The server must save all drawings sent to the other members. If this is done, when a new client comes, the server is able to send all drawings to him.

MP board applications have an interactive interface, which is dominated by the dynamic model. The user interface (GUI or text interface) or application interface is a part of the system that should be represented as a subsystem. Text interface or communication process are typical subsystems of MP boards.

A server MP board launches either a single MP board list and multiple MP board services or a single channels' list and multiple channel services. The client launches a single communication process, one graphical area and one text area so that multiple participant can share text and graphical information. Text areas display text-based messages from the server, while the graphical areas display vector-graphic drawings from the server.

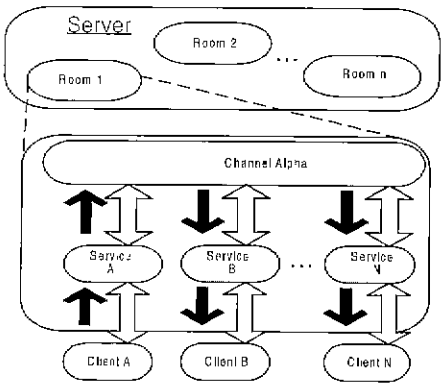


(Figure. 6) Object model of a server and a client in MP board

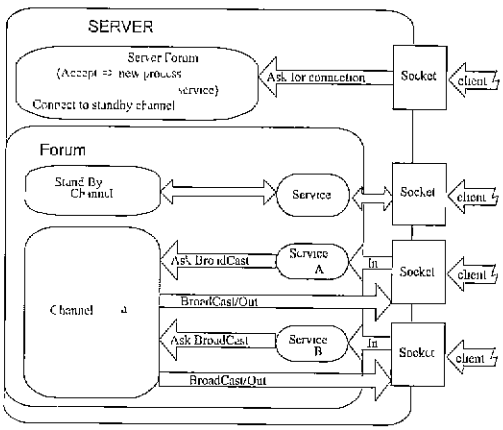
2.2.1 Design of server

Servers of MP boards are a collection of channels working together. MP boards are the place where the client can deliver messages. The message is automatically broadcasted to other clients. Message flow is drawn in (Figure 7).

(Figure 7) shows the two main processes in this part : the *channel process* and the *service process*. Both should be able to run in the same time. The channel process ensures that client is still connected to his channel by checking if connection is still alive and broadcasting messages to other clients who have joined to the same channel. Channel object also works as watchdog process ensuring that all connections are alive. Service process makes link between clients and channel objects. It has communication objects and offers functions to communicate with client objects. All clients must previously request a connection to the server. A maximum number should be set not to overload the server computer.



(Figure 7) Message broadcast



(Figure. 8) Connection process

(Figure. 8) shows connection process between a server and clients. Server forum accepts connection and launches a service process for that client. After launching service process it can go on listening for other incoming connections. Stand by channel stays alive as long as the server is running. Channel

process is created each time a new channel is created. When the last client has left the channel, the channel process has been removed.

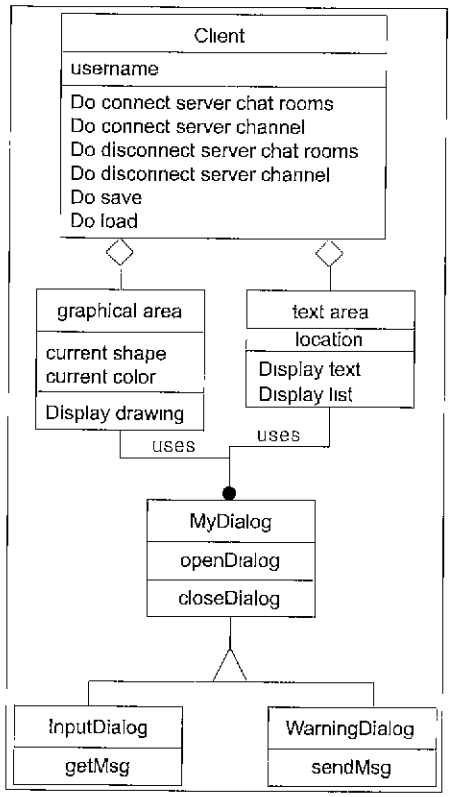
After the connection has been accepted, a *service process* is attached to this client and using this process. Then the client is redirected to a standby channel. At this point the client could see available channels. It can decide to join an available channel or do create its own channel. Communication interface is between the *client* and *service* process. This interface implements how we will physically send or receive messages. Considering *channel* and *service* process, the *ServerForum* process should be able to run with the other processes simultaneously.

2.2.2 Design of client

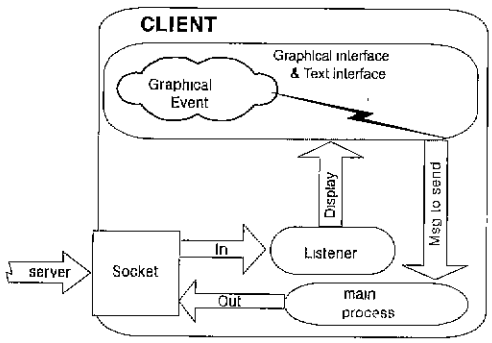
A user interface is designed as event-driven system like most of the graphical interfaces. Event-driven systems have better modular characteristics than procedure-driven systems and are also well suited to graphical user interface. A graphical and text user interface for MP board is developed using the AWT library. The design of of the user interface is heavily dependent upon the user interface objects of the JDK. Dialog boxes are useful means to interact with a system for users. Thus, dialog boxes are defined as objects interacting with client objects. (Figure. 9) shows dialog box objects and client object relationships.

(Figure. 10) displays functions of a client process. Both *listener* and *process client* communicate with a server process through a socket. Server process accepts the incoming connection and launch a new *service* process to provide a service to the client which tries to connect to a server. *Server* can go on listening to the incoming connections. When a client wants to create a new channel, the *serviceChatRooms* process launch a new process to manage the new channel. The socket number of the new channel is sent back to the client.

A user interface process sends events to the main process according to a user's input data. Listener process is called by main process if it wants to send a message to a server. Main process analyzes events from a user interface object and a listener object. Depending on the event, it calls the appropriate functions.



(Figure. 9) Object model of client process



(Figure. 10) Functions of client process

When a message from the server arrives, the *listener* object reads it from the communication interface.

Depending on its message type, the main process update the user interface. When the client enters texts or drawings, an event is thrown from the user interface process and is caught by the main process. Main process uses listener functions to send messages to server. When connection is lost, an error message will be displayed on the text interface or in a dialog-box. Similarly appropriate error messages are printed, if there is trouble in connecting to the server.

2.3 Other tools

It is not possible to describe all tools in detail because of length constraints. In this section a few important tools are described to help readers get idea about object modeling for the distance learning system.

Video conferencing tools are useful to both teaching and learning because of live motion pictures. It can enhance electronic courses not only by transferring the more subtle forms of communication, but also by providing additional visual and audio clues, which help to build pseudo-classroom environment to establish an informal rapport. Desktop video conferencing tools are designed based on point-to-point protocol. It allows two-way motion video and audio signals. A spotlight is also provided so that an instructor can answer to a student who requests the spotlight.

The newsgroup tool is also implemented based on RFC977 NNTP protocol. It works as an advertisement site grouped by theme and subtheme. Anyone can send a message to an USENET newsgroup and anybody is able to read it. Because news clients are similar to a mail tool client, mailbox model has been changed to make only one model for both news box and mail box.

2.4 Security

Distance learning system provides a mechanism to protect illegal users from access to the system by dynamic web programming. The system allows multiple user groups and defines rights for all group users. One solution to the security problem is to provide both authentication of the users and to restrict access based on where the user is located. Thus, access to the system is tested using a classic dialog window where users will enter a login name and correct password. Access will be granted only if the login name and its password are correct. A protection mechanism against URL hacking is also implemented.

The system provides four different levels of security.

No security : Access to trivial system functions, such as simple presentation, mailing list to get news, etc.

Low level : Access to non-sensible data such as BBS. Access will be denied to people who are just visitors to the system.

Medium level : Access to sensible data such as on-line resource and IRC tools.

High level : Access to private documents or specific tools such as video conferencing.

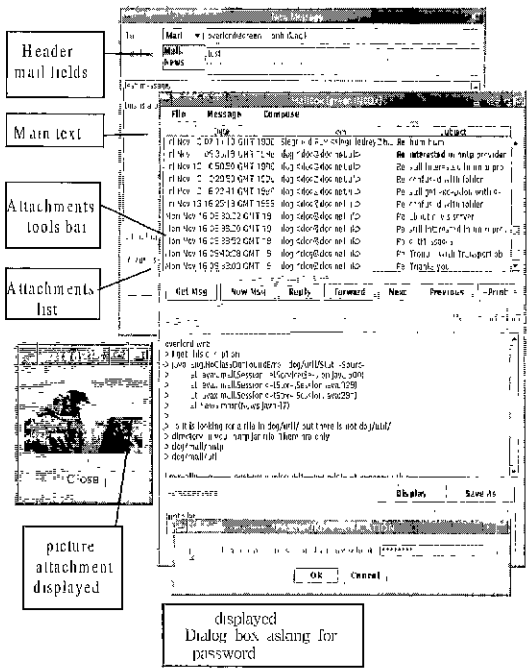
3. Implementation

A distance learning system has been built based on the object modelling described previously. All components are developed in JAVA language using JDK 1.0.2 released by the Sun Microsystem. Sun Sparc 20 workstation is selected as a hardware platform. Any personal computer can be a client system, if it has a network browser installed properly.

The system analysis based on the object modelling has been corresponding with JAVA because of its object-oriented programming style. Implemented systems in JAVA are compiled into byte codes that can be executed in JAVA virtual machine. However, the respond time of the distance learning system is fast enough to cover all interaction between instructors and students. The developed system provides many functions that found in other distance learning system, yet it is powerful to cooperate with internet environment. Because it is cooperant with internet, it is easy to add or modify extra functions or required modules in advanced distance learning system.

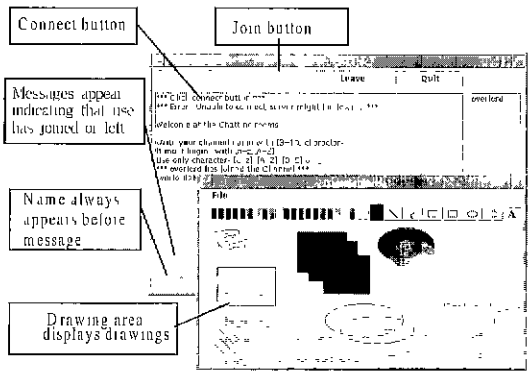
(Figure 11) shows screen capture of mailbox execution. The depicted mail system displays a list of

stored mail. It also shows how to attach images.



(Figure 11) Snapshot of mailbox executing

(Figure 12) shows screens from the multiparticipant board The behind screen displays text messages while the front screen transfers graphics information, such as squares, circles, and fillings This multiparticipant board is updated by any user connected to the same channel. The screen is also updated simultaneously throughout all connected computers



(Figure. 12) Snapshot of MP board executing

4. Conclusions

A distance learning system has been implemented based on the learning tools, such as usenet newsgroups, multiparticipant board, and video teleconferencing. Each module has been developed completely with object-oriented programming. The characteristics of developed systems are summarized as following.

First, the integrated system provides efficient learning tools compared to others. although the functionalities are similar. The integrated environment makes instructors and students feel comfortable and makes remote classrooms interesting.

Second, a multiparticipant board helps users induce creative ideas while they join and work together through a shared multimedia board It works well even for student groups whose levels of understanding about class material differ.

Third, the object-modeling helps the developed codes maintain robustness, extensibility, and reusability The developing period was short because JAVA class provides various well-proven libraries. It is also easy to extend new functions from existing capabilities.

The system exploits the power of inherited characteristics in the case of the hierarchical structure of classes. The quality of current video transmission is not fast enough to assuage jittering. In the future it is necessary to improve system performance for the video conference

References

[1] ASEE/IEEE, "Using Management Concepts in Planning for Distance Education," Nov. 1999.

[2] BMESS/EMBS, "Distance Learning in Biomedical Engineering," Vol 2, Oct. 1999.

[3] Frontiers in Education Conference, "A Distance Learning Program in Action," 1999.

[4] Laura N Gasaway, "Guidelines for Distance Learning and Interlibrary Loan," Journal of the American Society for Information Science, 50(14), pp. 1337-1341, Dec. 1999.

[5] E McNeil, "Debating Distance Learning," Communications of the ACM, 43(2), Feb. 2000.

[6] David Mioduser, Rafi Nachmias, Avigail Oren, and Orly Lahav, "Web-based Learning Environment (WBLE) Current Implementation and Evolving Trends," Journal of Network and Computer Applications, 22, 1999.

Hyo Jong Lee

e-mail hlee@moak.chonbuk.ac.kr

December 1986 University of Utah, Computer Science, B.S.

December 1988 University of Utah, Computer Science, M.S

August 1991 University of Utah, Computer Science, Ph D

1989~1991 : Senior programmer, University of Utah

1991~Present : Associate Professor, School of Electronics and Information Engineering, Chonbuk National University, Korea

Special interests Internet application, parallel algorithms, artificial intelligence, and pattern recognition