Symmetry **2015**, *7*, 294-304; doi:10.3390/sym7020294

OPEN ACCESS

symmetry

ISSN 2073-8994

www.mdpi.com/journal/symmetry

Article

Live Mobile Distance Learning System for Smart Devices

Jang Ho Lee [1], Doo-Soon Park [2], Young-Sik Jeong [3] and Jong Hyuk Park [4,*]

[1] Department of Computer Engineering, Hongik University, 72-1 Sangsu, Mapo, Seoul 121-791, Korea; E-Mail: janghol@cs.hongik.ac.kr

[2] Department of Computer Software Engineering, Soonchunhyang University, Asan, Chungnam 336-745, Korea; E-Mail: parkds@sch.ac.kr

[3] Department of Multimedia Engineering, Dongguk University, 30 Pildong-ro 1-gil, Jung-gu, Seoul 100-715, Korea; E-Mail: ysjeong@dongguk.edu

[4] Department of Computer Science and Engineering, SeoulTech, 232 Gongneung-ro, Nowon-gu, Seoul 139-743, Korea

***** Author to whom correspondence should be addressed; E-Mail: jhpark1@seoultech.ac.kr; Tel.: +82-2-970-6702

Academic Editor: Neil Y. Yen

Received: 9 December 2014 / Accepted: 12 March 2015 / Published: 25 March 2015

Abstract: In recent years, mobile and ubiquitous computing has emerged in our daily lives, and extensive studies have been conducted in various areas using smart devices, such as tablets, smartphones, smart TVs, smart refrigerators, and smart media devices, in order to realize this computing technology. Especially, the integration of mobile networking technology and intelligent mobile devices has made it possible to develop the advanced mobile distance learning system that supports portable smart devices such as smartphones and tablets for the future IT environment. We present a synchronous mobile learning system that enables both instructor and student to participate in distance learning with their tablets. When an instructor gives a lecture using a tablet with front-face camera by bringing up slides and making annotations on them, students in the distance can watch the instructor and those slides with annotation on their own tablets in real time. A student can also ask a question or have a discussion together using the text chat feature of the system during a learning session. We also show the user evaluation of the system. A user survey shows that about 67% are in favor of the prototype of the system.

Keywords: mobile learning; synchronous collaboration; smart device; tablet

1. Introduction

As mobile devices became popular in the last decade, they came to draw the attention of researchers and developers in the field of distance learning because they not only provide good performance but also allow users to participate in the distance learning regardless of their location even without desktop PC or notebook [1–5].

There are two types of mobile distance learning system depending on whether students participate in learning in different time (asynchronous) or at the same time (synchronous). ActiveCampus [6] is an asynchronous mobile distance learning system to support study community where students can share class material, have a discussion, do voting, *etc.*

One example of synchronous mobile learning systems is MLVLS that runs on a Symbian OS smartphone [7]. It supports live video and slide from the instructor but doesn't support interaction between an instructor and students. Furthermore, students have difficulty recognizing letters and figures of a slide on the smartphone since the display of a smartphone is not large enough. One solution to this small-display problem of the smartphone is to use the large-display smart device. There is a tablet-based synchronous learning system called Classroom Presenter [8] that allows a teacher and students in the same classroom to share slides and annotation. LiveNotes [9] also provides live mobile whiteboard sharing on a tablet. However, Classroom Presenter and LiveNotes don't provide live video of the lecture, which sometimes makes students have difficulty understanding the lecture. Therefore, we previously developed a tablet-based synchronous learning system [10] that provides live video, slide with annotation, and text feedback from students. However, the system needed improvement due to the following reasons. Since the system requires the instructor to use a desktop PC while students use tablets, the letters and figures in a slide with annotation that are large enough for an instructor to look at on his large PC display, give students hard time recognizing on their tablet display. It is also inconvenient and uncomfortable for an instructor to make annotation using a mouse compared to free drawing by hand on the touch display of a tablet. Although a stylus pen can solve these problems to a certain extent, it increases the equipment cost for setting up the client PC for instructor and restricts the instructor's location compared to the tablet that allows an instructor to give a lecture almost anywhere as long as the wireless network connection is available.

Thus, we developed a synchronous mobile learning system that provides tablet clients for both instructor and student. It allows an instructor to give a lecture using a tablet with front-facing camera by bringing up slides and making annotations on them. Students can watch the instructor and slides with annotation on their smart devices as well as asking questions using text.

2. Synchronous Mobile Learning System with Tablet Clients for Instructor and Student

Figure 1 shows the run-time architecture of the proposed synchronous mobile learning system that provides tablet client for both instructor and student. The system consists of clients and a server.

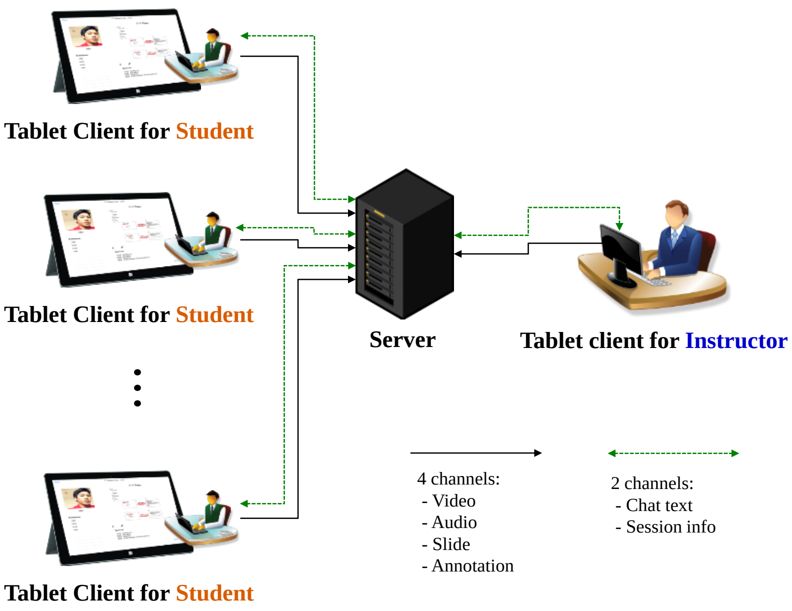

Tablet Client for Student
Tablet Client for Student
Tablet Client for Student
Server
Tablet client for Instructor
4 channels:
- Video
- Audio
- Slide
- Annotation
2 channels:
- Chat text
- Session info

Figure 1. Architecture of the presented synchronous mobile learning system.

There are two types of clients: client for instructor and client for student. The instructor client encodes video in H.263 [11] and audio in G.723.1 [12] respectively and sends them to the server in separate channels. It also sends to the server slides in pdf, packets of grouped annotation events, chat text and session information in separate channels. The server broadcasts those data to student clients. When a student client receives those data, it decodes video, audio and slide and presents them to the students with other data such as annotation, chat text, and session information. The student client can also send chat text and session information to the server.

Figure 2 shows the architecture of the tablet client for instructor. The tablet client for instructor consists of 6 modules: video handler, audio handler, slide handler, annotation handler, chat hander, and session handler.

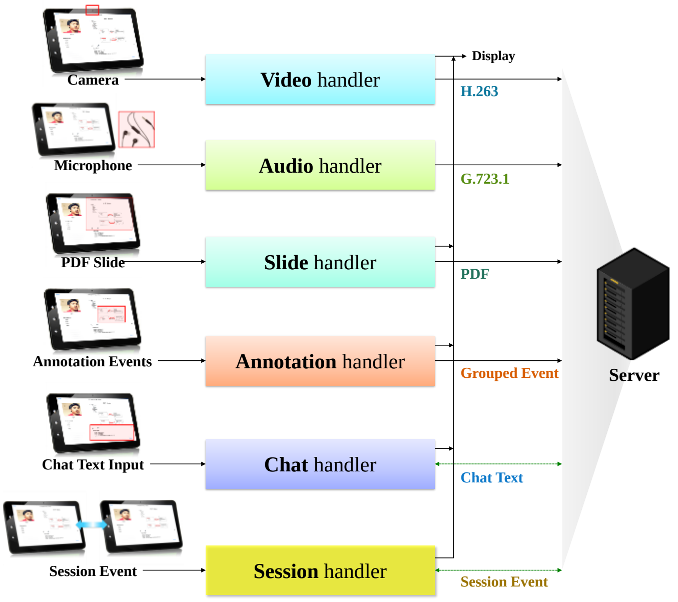


Figure 2. Architecture of the tablet client for instructor.

The video handler consists of three parts: local video handling module, video encoding, and video transmission module. The local video handling module captures the video with camera attached on the front side of the tablet and renders video on the display of the instructor’s tablet by processing raw video data and generating RGB data. The video encoding module is in charge of encoding the captured raw video data into the compressed video data for network transmission using the H.263 standard video compression algorithm. The video transmission module gets the compressed video data encoded in H.263 by the video encoding module and sends it to the server. The audio handler consists of two parts: audio encoding module and audio transmission module. The audio encoding module captures the audio signal from the microphone of the tablet and then encodes the audio data in the PCM format into the compressed audio data for network transmission with the G.723.1 standard audio compression algorithm.

The audio transmission module gets the compressed audio data encoded in G.723.1 by the audio encoding module and transmits it to the server. It should be noted that the audio handler does not play the captured audio through the local speaker of the instructor’s tablet since the instructor can hear his/her own voice. The slide handler is composed of two modules: local slide handling module and the slide transmission module. The local slide handling module reads a slide file in the pdf format decodes it and renders a page on the tablet display. The slide file could be read directly from the local storage of a tablet or from a remote storage of a cloud through network. The slide transmission module is in charge of delivering a slide page in pdf format to the server through network whenever an instructor moves to a certain page of a slide file. The annotation handler module consists of three parts: local annotation handling module, annotation event grouping module, and annotation transmission module. The local annotation handling module captures the input events related to the annotation and renders the annotation on a slide page on the display of the instructor’s tablet.

The annotation event grouping module is responsible for grouping a series of annotation events into a packet that occurs during a short period of time. The annotation transmission module gets the packets from the annotation event group module and sends them to the server. It should be noted that sending a group of events instead of sending every single event can reduce the network congestion. The chat handler is also composed of two parts: local chat handling module and chat networking module. The local chat handling module renders on the display of the instructor’s tablet chat text either from the local input of the instructor’s tablet or from the chat networking module. The chat networking module is in charge of sending to the server the chat text from the local input of the instructor’s tablet as well as receiving the chat text from the server and delivering it to the local chat handling module. The session handler consists of two parts: local session handling module and session networking module. The local session handling module updates its local copy of the session state by reflecting the session event either from the instructor’s tablet or from the session networking module, and renders the session state on the display of the instructor’s tablet. The session event occurs when a user joins a session or leaves a session. The session networking module not only sends to the server session events from the local input of the instructor’s tablet but also receives session events from the server and delivers it to the local session handling module. It should be noted that chat handler and the session handler of the instructor’s tablet client can send data to the server as well as receive data from the server while the video handler, the audio handler, the slide handler and the annotation handler of the instructor’s tablet client can only send data to the server.

Figure 3 shows the architecture of the tablet client for student. The tablet client for student consists of 6 handlers: video handler, audio handler, slide handler, annotation handler, chat handler, and session handler.

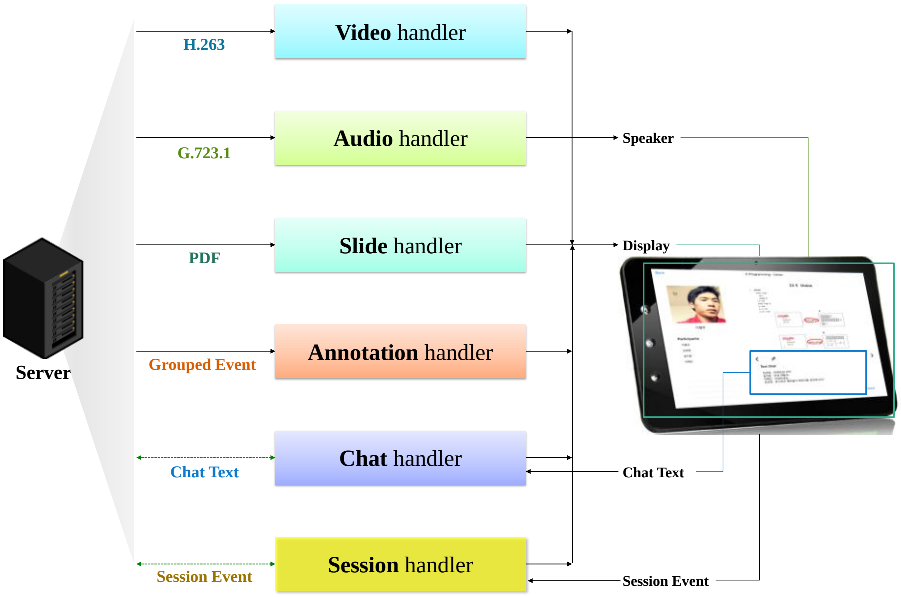


Figure 3. Architecture of the tablet client for student.

The video handler consists of three parts: video receiving module, video decoding module, and video rendering module. The video receiving module receives the video data in the H.263 format from the server and gives them to the video decoding module. The video decoding module gets the video data in H.263 from the video receiving module and decodes them into RGB and hands them over to the video rendering module. The video rendering module gets the decoded video data in RGB and renders them on the display of the student's tablet. The audio handler consists of three parts: audio receiving module, audio decoding module, and audio play module. The audio receiving module receives the audio in the G.723.1 format from the server and hands them over to the audio decoding module. The audio decoding module gets the audio data from the audio receiving module, decodes them into PCM with the G.723.1 standard audio decompression algorithm and hands them to the audio play module. The audio play module gets the audio in PCM and plays it through the speaker.

The slide handler is composed of three parts: slide receiving module, slide decoding module, and slide rendering module. The slide receiving module receives slide pages in pdf from the server and gives them to the slide decoding module. The slide decoding module gets the slide pages in pdf from the slide receiving module, decodes them and hands them over to the slide rendering module. The slide rendering module gets the decoded slide page data and renders them on the display of the student's tablet. The annotation handler consists of three parts: annotation receiving module, annotation event ungrouping module, and annotation rendering module. The annotation receiving module receives the packets of grouped annotation events from the server and hands them over to the annotation event ungrouping module.

The annotation event ungrouping module gets the annotation event packet, ungroups it into a series of annotation events and hands them over to the annotation rendering module. The annotation rendering module receives those series of annotation events and renders them on a slide page on the display of the student's tablet. As the chat handler of the instructor's tablet client, the chat handler of the student's tablet client also consists of two parts: local chat handling module and chat networking module. The local chat handling module renders on the display of the student's tablet chat text either from the local input of the student's tablet or from the chat networking module. The chat networking module not only sends to the server the chat text from the local input of the student's tablet but also receives the chat text from the server and hands it over to the local chat handling module.

As the session handler of the instructor's tablet client, the session handler of the student's tablet client also consists of two parts: local session handling module and session networking module. The local session handling module updates its local copy of the session state by reflecting the session event from the student's tablet or from the session networking module, and renders the session state on the display of the student's tablet. The session networking module sends to the server session events from the local input of the student's tablet as well as receives session events from the server and gives it to the local session handling module. It should be noted that the chat handler and the session handler of the student's tablet client can send data to the server as well as receive data from the server while the video handler, the audio handler, the slide handler and the annotation handler of the student's tablet client can only receive data from the server.

Figure 4 shows the architecture of the server that is in charge of multicasting data to clients. The server consists of 6 modules: video multicasting module, audio multicasting module, slide multicasting module, annotation multicasting module, chat multicasting module and session multicasting module.

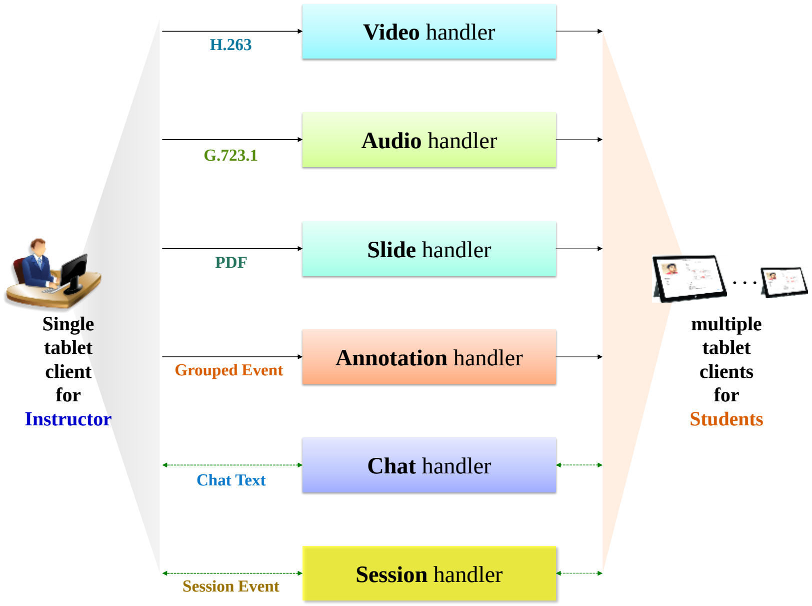
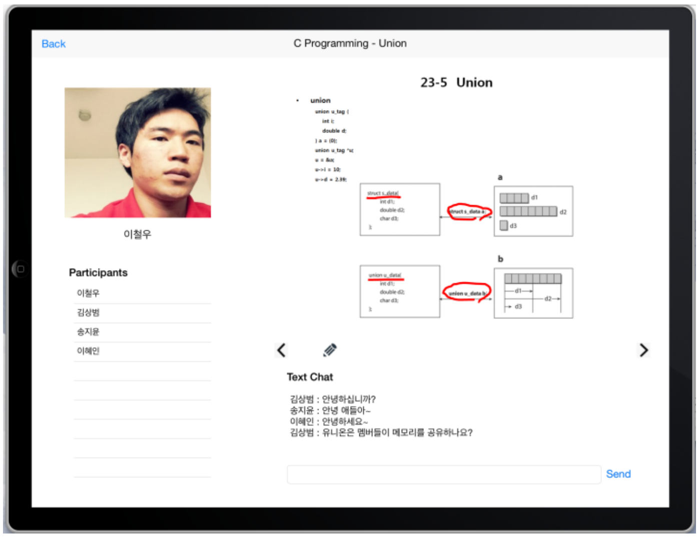


Figure 4. Architecture of the server.

The video multicasting module receives the video data in H.263 from the instructor's tablet client and multicasts them to the multiple student's tablet clients. The audio multicasting module receives the audio data in G.723.1 from the instructor's tablet client and multicasts them to the multiple student's tablet clients. The slide multicasting module receives the slide pages in pdf from the instructor's tablet client and multicasts them to the multiple student's tablet clients. The annotation multicasting module receives annotation packets of grouped annotation events and multicasts them to the multiple student's tablet clients. The chat multicasting module receives chat texts from the instructor's tablet client or one of the student's tablet clients and multicasts them to the rest of the clients. The session multicasting module receives session events from the instructor's tablet client or one of the student's tablet clients, updates its own copy of the session state and multicasts them to the rest of the clients. It should be noted that a newly joined client receives the current session state from the server that maintains its own copy of the session state. The clients for instructor and student are currently being developed on iPad in Objective-C using the Xcode 5 [13] integrated development environment and the iOS 7 SDK [14] on Mac OS X 10.9. The server is being implemented in Microsoft Visual C++ with MFC on Windows 7.

3. User Interface of the System

Figure 5 shows the user interface of the tablet client for instructor. It consists of five parts: video panel, slide panel, slide control, chat panel, and participant list panel.

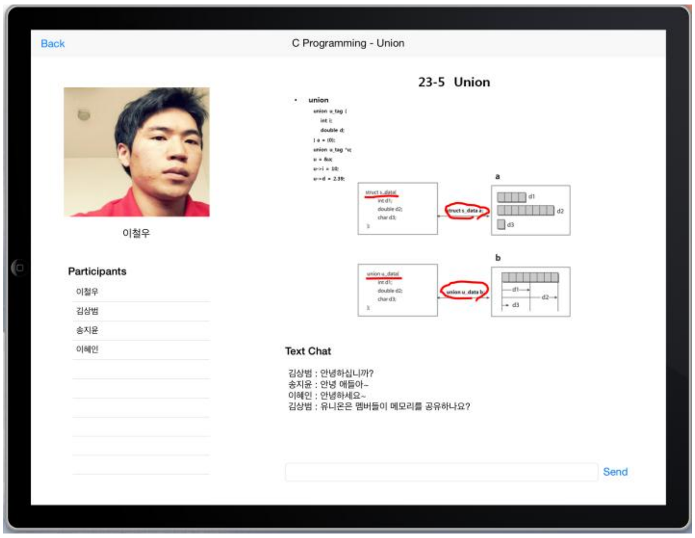


```
union
  union u_tag {
    int i;
    double d;
  } a = {0};
  union u_tag *u;
  u = &a;
  u->i = 10;
  u->d = 2.39;
```

김상범 : 안녕하십니까?
송지윤 : 안녕 애들아~
이혜인 : 안녕하세요~
김상범 : 유니온은 멤버들이 메모리를 공유하나요?

Figure 5. User interface of the tablet client for instructor.

The video panel shows the instructor who is giving a lecture in real time. The slide panel shows a slide page as well as annotations that the instructor is currently making on the slide page. Below the slide panel, there is a slide control including arrows and a pen that enables an instructor to move to the previous or next page and making annotation, respectively. The chat panel shows the chat text as well as allows the instructor to type and send chat text to students. The participant list panel shows the list of students and an instructor who are currently participating in the ongoing lecture. Figure 6 shows the user interface of the tablet client for student. It consists of four parts: video panel, slide panel, chat panel, and participant list panel. It should be noted that, unlike the tablet client for instructor, it doesn't have the slide control because only an instructor is allowed to move to the next or previous page as well as make annotation on a slide. The video panel shows the instructor who is giving a lecture in real time. The slide panel shows a slide page and annotations being drawn on the slide page.



- union

```
union u_tag {
   int i;
   double d;
} a = {0};
union u_tag *u;
u = &a;
u->i = 10;
u->d = 2.39;
```

Text Chat

김상범 : 안녕하십니까?
송지윤 : 안녕 애들아~
이혜인 : 안녕하세요~
김상범 : 유니온은 멤버들이 메모리를 공유하나요?

Figure 6. User interface of the tablet client for student.

The chat panel allows a student to see the chat text as well as to send chat text to others. The participant list panel shows the list of students and an instructor who are currently participating in the lecture.

4. Learning Scenario

A learning scenario based on the user interface shown in Figures 5 and 6 is as follows.

[Step 1] *Creation and join of a learning session (or lecture) by an instructor*: Before the scheduled start time of the lecture, an instructor creates a learning session in advance using his tablet client shown in Figure 5. The instructor types the title of the lecture, which is shown on the title bar of the instructor's tablet client, joins the learning session, brings up a slide file, and waits for the students to join the learning session by checking out the participant list panel.

[Step 2] *Joining of the learning session by students*: Before the start time of the lecture, students bring up their clients on their tablets and join the learning session created by the instructor. Late students are also allowed to join the ongoing learning session.

[Step 3] *Initiation of the learning session by instructor:* When students are in the learning session, the instructor starts to give a lecture. The instructor starts a lecture on the content of the slide page that is currently shown. There are four types of actions that an instructor can use in giving a lecture: explaining with voice, making gestures with face, making annotations on a slide, and typing text on the chat. The instructor can use any combination of those four types of actions to maximize the learning.

[Step 4] *Feedback or question and answer between students and an instructor during the ongoing learning session*: Figure 6 shows that students are talking together using chat text after they joined a lecture. During an ongoing learning session, a student can ask the instructor a question by using text chat, which is currently shown as the last sentence of the text chat panel. When the instructor sees a question on the chat panel in the tablet client for instructor, he or she can answer it by making gestures, using voice, making annotations on a slide and/or typing chat text.

[Step 5] *End of the learning session by an instructor*: The instructor finishes the lecture and ends the current synchronous learning session, which makes all the participants leave the session.

5. User Evaluation

We asked a group of students (9 students) in our department of computer science to test our prototype for 3 days and then conducted the preliminary user survey asking them how they felt about our system.

The result of the survey shows that 67% (6 students) of the testers are in favor of the system because the combination of video, audio, slide with annotation, and text chat presented on a student's tablet client makes them learn effectively from a distance and the instructor's tablet client is easy to use when making annotations because it is based on touch display. However, 22% (2 students) of the testers were against the system because the quality of the video and audio is not good enough, which, we think, will be improved when the implementation including the optimization is completed. 11 percent (1 student) of the testers were neutral, meaning that they didn't find the system good nor bad.

After this preliminary user survey, we conducted more detailed user evaluation with the same group of 9 students. For the purpose of this evaluation, we first identified the following three aspects of learning effectiveness: how effective students feel about the reception of the lecture content, how effective students feel about the interaction among participants, and how good students feel about their concentration on learning. Then, we asked students to give points in each aspect (1 is the lowest point and 3 is the highest point). Table 1 shows the result of the survey.

Table 1. Students' perception on learning effectiveness (3 is the highest point).

Effectiveness	**#Students**			**Mean**
	1	**2**	**3**	
Lecture reception	1	2	6	2.56
Class Interaction	1	1	7	2.67
Lecture concentration	4	3	2	1.78

The mean value of the students' perception on lecture reception (mean: 2.56) and class interaction (mean: 2.67) are slightly above the middle. But the students' concentration on lecture is low (mean: 1.78), which means that the students have some difficulty concentrating on the lecture with our system. Students felt that the slide with annotation on a 9.7-inch tablet is still small for effective lecture reception. They also felt that typing texts on a virtual keyboard is still inconvenient for effective class interaction. They also mentioned that they often have hard time concentrating on a lecture on a tablet because of the constant notifications from various apps including SNS apps on their tablets. More in-depth user evaluation of our system can be found in [15].

6. Conclusions

The unique contribution of this research is that we presented a synchronous mobile learning system supporting tablet clients for both instructor and student that provides the following features, as well as conducted user evaluation of the system. It allows an instructor to give a lecture in front of the tablet quipped with front cam by bringing up a slide, using gestures and voice, making annotation on the slide, and using text chat in real time. A student can watch the instructor's gesture, listen to the instructor's voice, look at the lecture slide with annotation in real time. A student can also ask questions and have discussion using text chat feature of the system.

The user evaluation shows that the presented tablet-based synchronous mobile learning system provides students with effective lecture reception and class interaction to a certain extent, but gives students hard time to concentrate on the mobile learning due to the constant notifications from other apps such as SNS apps on their tablets.

We are currently implementing the presented system. When the implantation is completed, we plan to conduct empirical studies about giving lectures using a tablet as an instructor and taking a class using a tablet as a student.

Author Contributions

This paper is largely written by Jang Ho Lee based on his earlier work. Doo-Soon Park helped to conceive the idea of the mobile distance learning system for smart devices as well as to develop the organization of the paper. Young-Sik Jeong improved the description of the system in Section 2 and developed the idea of the learning scenario in Section 4. Jong Hyuk Park helped to improve the organization of the paper in terms of the academic value.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgements

This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (NIPA-2013-H0301-13-4007) supervised by the NIPA (National IT Industry Promotion Agency).

References

1. Wains, S.I.; Mahmood, W. Integrating M-learning with E-learning. In Proceedings of the 9th ACM SIGITE Conference on Information Technology Education, Cincinnati, OH, USA, 16–18 October 2008; pp. 31–38.
2. Lo, E.; Tan, Q. Design Principles for Facilitating Collaboration in Mobile Environments. In Proceedings of the 6th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), Chengdu, China, 23–25 September 2010; pp. 1–4.
3. Chung, K.; Lee, J. Design and Development of M-Learning Service Based on 3G Cellular Phones. *J. Inf. Process. Syst.* **2012**, *8*, 521–538.
4. Gupta, V.; Chauhan, D.; Dutta, K. Incremental Development and Revolutions of E-learning Software Systems in Education Sector: A Case Study Approach. *Hum. Cent. Comput. Inf. Sci.* **2013**, *3*, doi:10.1186/2192-1962-3-8.
5. Weng, M.; Shih, T.; Jung, J. A Personal Tutoring Mechanism Based on the Cloud Environment. *J. Converg.* **2013**, *4*, 37–44.
6. Gridwold, W.G.; Shanahan, P.; Brown, S.W.; Boyer, R.; Ratto, M.; Shapiro, R.B.; Truong, T.M. ActiveCampus: Experiments in Community-Oriented Ubiquitous Computing. *IEEE Comput.* **2004**, *37*, 73–81.

7. Ullrich, C.; Shen, R.; Tong, R.; Tan, X. A Mobile Live Video Learning System for Large-Scale Learning-System Design and Evaluation. *IEEE Trans. Learn. Technol*. **2010**, *3*, 6–17.
8. Anderson, R.; Anderson, R.; Davis, P.; Linnell, N.; Prince, C.; Razmov, V.; Videon, F. Classroom Presenter: Enhancing Interactive Education with Digital Ink. *IEEE Comput.* **2007**, *40*, 56–61.
9. Kam, M.; Wang, J.; Iles, A.; Tse, E.; Chiu, J.; Glaser, D.; Tarshish, O.; Canny, J. LiveNotes: A System for Cooperative and Augmented Note-Taking in Lectures. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Portland, OR, USA, 2–7 April 2005; pp. 531–540.
10. Lee, J. Development and Usability Assessment of Tablet-Based Synchronous Mobile Learning System. In *Ubiquitous Information Technologies and Applications*; Lecture Notes in Electrical Engineering; Springer: Berlin/Heidelberg, Germany 2014; Volume 280, pp. 301–306.
11. ITU-T H.263: Video Coding for Low Bit Rate Communication. Available online: http://www.itu.int/rec/T-REC-H.263 (accessed on 12 March 2015).
12. ITU-T G.723.1: Dual-Rate Speech Coder for Multimedia Communications Transmitting at 5.3 and 6.3 kbit/s. Available online: http://www.itu.int/rec/T-REC-G.723.1 (accessed on 12 March 2015).
13. Xcode 5. Available online: https://developer.apple.com/xcode (accessed on 12 March 2015).
14. iOS7. Available online: https://developer.apple.com/ios7 (accessed on 12 March 2015).
15. Lee, J. Evaluation of the Synchronous Mobile Learning System Supporting Tablets for Instructor and Student. In *Computer Science and its Applications*; Lecture Notes in Electrical Engineering; Springer: Berlin/Heidelberg, Germany, 2015; Volume 330, pp. 1093–1099.