

# An Empirical Study of Audience-Driven Interactive Live Television on the Internet

Yoshia Saito  
Iwate Prefectural University  
152-52 Sugo, Takizawa, Iwate 020-0193, Japan  
y-saito@iwate-pu.ac.jp

Yuko Murayama  
Iwate Prefectural University, Japan  
152-52 Sugo, Takizawa, Iwate 020-0193, Japan  
murayama@iwate-pu.ac.jp

## ABSTRACT

Traditional video services on the Internet are more in the broadcasting service nature such as streaming and video-on-demand (VoD). Recent services incorporate more of the interactive nature of network applications such as easy video sharing and those with a chat function. We have proposed an audience-driven broadcast service model in which audiences can send their wish to a broadcaster such that they like to see some specific objects while broadcasting as well as the broadcaster can reply back to the request. We implemented a prototype system for audience-driven live broadcasting and evaluated based on the results from the experimental broadcast at our university campus festival with two hundred audiences. This paper reports our experiment and findings.

## Categories and Subject Descriptors

H.5.3 [Group and Organization Interfaces] Synchronous interaction

## General Terms

Design, Experimentation

## Keywords

Interactive television, Internet broadcast, Audience-driven live TV

## 1. INTRODUCTION

Broadband Internet services enhance video streaming on the Internet to a great extent. People without technical knowledge on broadcasting can contribute and obtain video contents easily through the web-based video-sharing services such as YouTube [1]. Recently, some broadcast services provide users with the functions to communicate with broadcasters and audiences using the interactive nature of the network; Ustream.tv [2] and Stickam [3] are typical ones. Those services provide live broadcast function and a chat function such that broadcaster and its audiences can communicate with each other while watching live streaming. Moreover, personal broadcasting is so easy. One could produce an original live program using widely available software with an economical web camera and a personal computer. We presume that the more live broadcast services we have, the more popular small and middle-scale personal broadcasting will become.

In our research, we have been trying and conducting experimental Internet broadcasting in practice [4]. Through the experiments we have found that it is quite difficult to provide audiences with the contents with quality. We are not professional

broadcaster with a huge amount of budget and technical knowledge after all. What we need is something we can afford yet we can entertain the audiences. To solve this problem, we proposed an audience-driven broadcast service model which enables audiences to request a broadcaster to execute some actions and an interactive broadcasting system, AdlivTV (Audience-Driven LIVE TV system) for the application of the model [5]. We implemented an early version of our prototype system. The prototype system, however, had only a few simple functions and was not evaluated in a practical environment.

In this paper, we report the design and implementation of our latest prototype system and the evaluation based on a practical experiment. The prototype system has several functions for audiences to make a request to a broadcaster. We evaluate the system based on our experimental broadcast at our university campus festival in October 2009 for two days. The paper is organized as follows. In the next section, we describe related work. Section 3 introduces the system model of the AdlivTV. Section 4 presents the design of the prototype system of the AdlivTV, its system architecture and user interface. Section 5 reports the experiment using the prototype system and its results. Section 6 gives some conclusions and our future work.

## 2. RELATED WORK

Interactive television (iTV) [6] is a research area to give interactive features to television in order to break away from a traditional one-way broadcast service model. The iTV provides audiences who watch the TV with various interactive functions with shared experiences. Typical iTV systems have chat functions such as text chat and voice chat to let audiences communicate with each other so that they have shared experiences [7, 8, 9]. They can talk with the other audiences who watch the same TV programme and feel empathy. Harboe et al. [10] proposed a Social TV system which lets audiences to know what TV programmes the other audiences are watching through ambient displays. The system makes the audiences aware of context of their friends and family. Shamma et al. [11] developed a synchronized video player for online videos to provide a shared experience. They added the video player into an instant messenger so that the IM users can watch the online videos chatting with each other. Drugge et al. [12] studied telepresence which provides an experience as if audiences were in the same place and offers various interactive functions to the audiences for the shared experience.

The AdlivTV is different from these iTV work in that it aims to provide audiences a shared experience as if they were the members of the broadcaster. The audience would be involved in broadcasting in more active manner than with telepresence,

because they could send what they would like to watch, to the broadcaster.

### 3. AUDIENCE-DRIVEN LIVE TV SYSTEM

The AdlivTV provides several interactive communication functions between audiences and a broadcaster to allow the audiences to send a request to the broadcaster and gives them a right to control indirectly broadcasting operations. The motivation of the AdlivTV is to let the audiences participate in the broadcast more actively and keep them interested.

Figure 1 shows the system model of the AdlivTV. In this model, we define broadcaster as a group of members who work together to broadcast a live content such as server operators and camera operators. Firstly a broadcaster starts broadcasting a live video streaming to the Internet. Secondly audiences watch the live video streaming on their computer or TV-like screen. Thirdly the audiences can request some actions to the broadcaster. For example, the requests would be “Look here”, “Move here”, “Zoom up/out” and so on. The broadcaster waits for the audiences’ requests. The audiences send their requests to the broadcaster using input devices such as a mouse, a keyboard, a microphone and a TV-like controller. The requests are represented by various ways not only chat. Fourthly when the audiences’ requests are arrived, the broadcaster should respond the requests as much as possible. The requests and responses can be received by all audiences in order to realize shared experience. While broadcasting, the request and response cycle are repeated.

The scope of the AdlivTV is small and middle-scale personal broadcasting and the target users are inexpert broadcasters neither with any budget nor professional knowledge to produce broadcasting contents. The AdlivTV provides the inexpert broadcasters with effective broadcast skills since they can have direct feedbacks from the audiences while broadcasting.

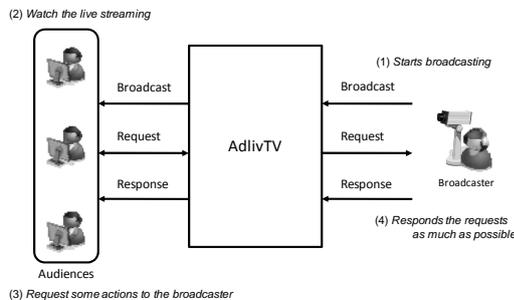


Figure 1. System Model

## 4. IMPLEMENTATION

We designed a prototype system of the AdlivTV to conduct a practical experiment. The implementation of the prototype system is based on Adobe Flash so that audiences can watch live video streaming on their web browser without specific software. In this section, we describe the system architecture and a user interface with interactive request functions.

### 4.1 System Architecture

The prototype system has a streaming server, several clients and a camera operator. Figure 2 shows the system architecture of the prototype system.

We use Red5 v0.7.0 final to build the streaming server. The Red5 is an open source Flash server written in Java. The streaming server receives video data from the camera operator and broadcasts it to clients. A server program also works on the server to handle requests from the clients and manage the system.

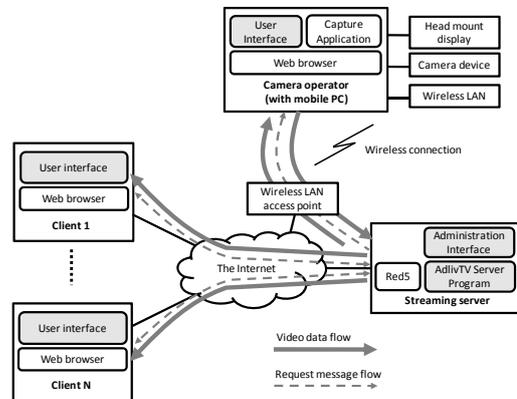


Figure 2. System Architecture

The camera operator has a Head-mounted display (HMD), a camera device and a small mobile PC equipped with wireless LAN (WLAN). The camera operator can check the current broadcasting and audiences’ request through the HMD. The mobile PC is connected with a WLAN access point and the camera operator can walk around to comply with audiences’ various requests.

When the client accesses to a web page for the broadcasting, a Flash application for the audiences is launched. At first, the client connects to the AdlivTV server program with a unique number. The server program generates an audience’s ID using the received number as a seed and sends it to the client. The client keeps the unique number in the Cookie on the web browser and can receive same ID from the server each time. The ID is used for system logs identifying audiences. The logs maintain connected time, disconnected time and audiences’ request with the IDs. After obtaining the ID, the client establishes several connections to the server and they can send requests through an original user interface for the AdlivTV watching live video. In the next subsection, we describe detail of the user interface

### 4.2 User Interface

Both audiences and broadcasters use the same user interface to watch a live video with the prototype system. Figure 3 shows the user interface of the AdlivTV. (Note: It is the Japanese version and the underlined English parts are added for the purpose of the explanation.) The user interface provides a live video player and two functions for the audiences to make a request.

The first function is a chat. This popular communication function lets the audiences and the broadcasters exchange text messages each other.

The second function is a request icon function. Those icons are presented in the menu and the users can select one of them by a click and the request is sent to the broadcaster. We have the following icons; (1) **Look**: Focus the camera onto the specific object, (2) **Move**: Move to the specific location, (3) **What**: Ask about the specific object, (4) **Attention**: Pay attention to the specific object, (5) **Like**: Understand that the specific object is my

favorite, (6) **Talk**: Talk to the specific person, (7, 8) **Zoom in/out**: Zoom in/out at the current angle, (9, 10, 11, 12) **Look left/up/down/right**: Turn to the specific direction, (13) **Good**: Continue current good broadcasting, (14) **Bad**: Change current bad broadcasting. (Note: The *Good* and *Bad* icons increase counters shown on the bottom of the video field.) Those icons are unique with our system. The icon function provides the audiences and the broadcaster with easy and intuitive communication tool. It shows the icons selected by the audiences on the video in a few seconds. It is useful for stylized requests.

All audiences can use the icon function anytime and the icons are shown on the video field of the all audiences. Sometimes several audiences may use the request icon function at the same time and several icons are shown on the video field. In this case, the broadcaster, i.e. the camera operator judges what request should be applied and executes one of them.

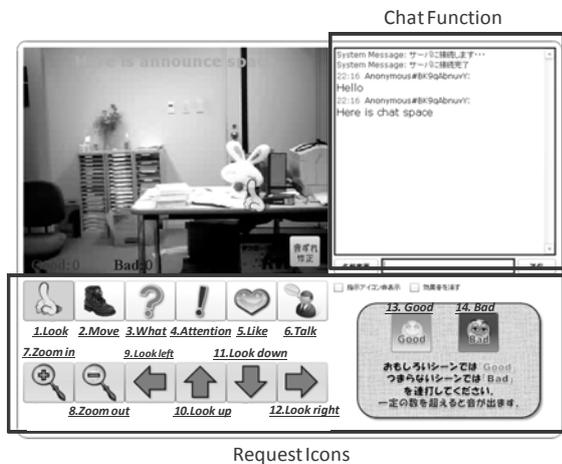


Figure 3. User Interface

## 5. EXPERIMENT

We carried out a practical experiment to study how effective the audience-driven broadcasting works and in what way audiences make requests to the broadcaster while finding issues of the system in the practical situation. The experiment was conducted in the campus festival of Iwate Prefectural University on October 25 and 26, 2008.

### 5.1 Environment

The festival was held in the campus of Iwate Prefectural University. Figure 4 shows the network configuration in the experiment. The AdlivTV server was placed in a broadcasting server room of our laboratory and connected to the Internet at 100 Mbps. Many live performances such as singing and dancing were held at the main stage. We set up a WLAN access point equipped with a router function near the stage. The WLAN access point connected to the AdlivTV server via LAN in the University. In our experiment, we used two cameras, a mobile one and a fixed one so that the broadcaster could switch the video footage when the camera operator changed the battery of the mobile PC or some problems were happened. We used mainly the mobile camera in the experiment.

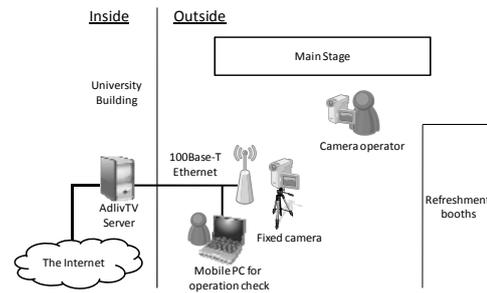


Figure 4. Network configuration

### 5.2 Results

At first, we counted the total audience on October 25th from 10:00 to 15:00 and 26th from 10:00 to 17:00, 2008. Note that we excluded broadcasters' access from all the results. The total number of the audience was 93 on the first day and 200 on the second day. It was small-scale broadcasting in the scope of the AdlivTV and the number of audiences was quite adequate for our experiment.

We analyzed the changes in the number of audiences for the two days to study increase and decrease of the audiences. Figure 5 shows the changes per 60 seconds. In our experiment, we had two types of cameras, viz. the mobile one and the fixed one. We needed quite often to switch the mobile one to the fixed one due to the various troubles of the mobile one such as a need to change the battery of the mobile PC attached to the mobile camera. Vertical solid and dotted lines in the graph show the time when we switched to the mobile camera and to the fixed one respectively. We can see that audience rate decreased when we switched to fixed camera and increased when mobile camera was used. From the result, we find that audiences are sensitive for the camera work and broadcasters could get more audiences if they provided their live programme with audience-driven and dynamic camera work.

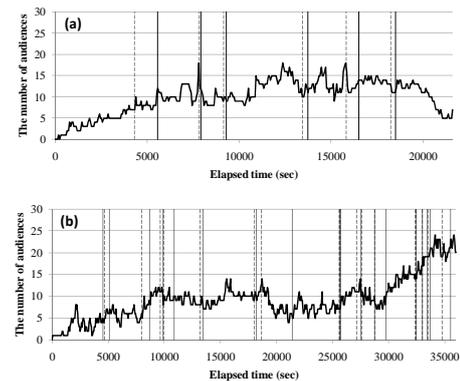
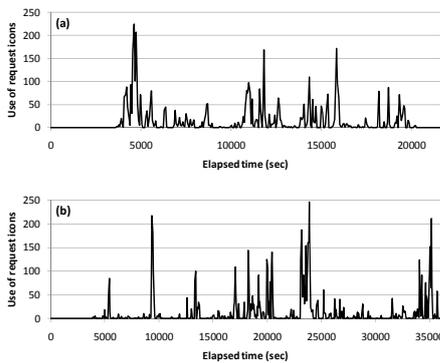


Figure 5. Change in the number of audiences per 60 seconds (a): 25 Oct. 2008, (b): 26 Oct.2008

We analyzed the use of request functions as well. As for the chat function, the audiences used the chat function 146 times on the first day and 63 times on the second day. The request messages included in the chat were 17 on the first day and 4 on the second day. Examples of the request message includes "Please go into the specific building", "Move to the best shooting place for fireworks" and "Too much noise, fix it please" and so on.

However, the request messages were only ten percents of the total chat messages and the rest were comments on the broadcasting. From the result, we can see that the chat function was not used frequently to send requests to the broadcaster but it was useful for complaints and some requests in the specific situation.

As for the icon function, we analyzed change in the use of the request icons per 60 seconds and total use of each request icons. Figure 6 shows the use of the request icons. From the graphs, we can see that the use of the request icons is concentrated in a short time and it happens suddenly. Since the camera operator judged which requests should be responded, the concentration of requests sometimes made the camera operator confused. We should support the camera operator to be able to select an appropriate one out of hundreds of requests.



**Figure 6. Change in the use of the request icons per 60 seconds (a): 25 Oct. 2008, (b): 26 Oct. 2008.**

**Table 1. Total use of each request icon**

Type of request : Total use			
Look:1230	Like:617	Look left:1460	Good:1155
Move:602	Talk:592	Look up:190	Bad:1218
What:943	Zoom in:617	Look down:90	
Attention:466	Zoom out:555	Look right:888	

Table 1 shows the total use of each request icon. The frequently-used icons include “Look”, “What”, “Like”, “Look left”, “Look right”, “Good” and “Bad”. The Look icons would be useful for the audiences to control camera works. The “What” icon was important to acquire information from the broadcaster and the “Like”, “Good” and “Bad” icons helped to present audiences’ impression to the broadcaster. On the other hand, “Look up” and “Look down” were almost never used in the experience because the main stage was horizontally wide and it did not need vertical camera works. From the results, we found that the audiences could use the request icon function easily and they were involved actively in the broadcast using the function.

We also found an issue of the request icon function for the camera operator to respond requests. The audiences used “Talk” icon to a person on the main stage and they continued to use the icon although the camera operator could not execute the action. Since the camera operator could not reply that the request was not executable, the audiences kept sending the useless requests. It would not satisfy the audiences if their request were ignored many times. The system should provide a response function so

that the camera operator can decline expressly inexecutable requests with an easy operation.

## 6. CONCLUSION

We introduce our work on the AdlivTV and reported our implementation of a prototype system and its evaluation by a practical experiment at our university campus festival. From the experiment, we had four findings; 1) broadcasters could have more audiences if they provided audience-driven and dynamic camera work, 2) the icons for requests were used in a short period of time and in a burst way, 3) the audiences used the icon function easily and were actively-involved in the broadcast by the use of the function, 4) a response function should be provided for the camera operator to decline inexecutable requests.

As future work, we will study support functions for the camera operator since workload of the camera operator is intensive with the current prototype system. We need to provide more support functions for the camera operators so that they can select an appropriate one out of hundreds of requests and decline requests expressly with easy operation.

## 7. REFERENCES

- [1] YouTube, <http://www.youtube.com/>.
- [2] Ustream.tv, <http://www.ustream.tv/>
- [3] Stickam, <http://www.stickam.com/>
- [4] N. Takamizawa and Y. Murayama, “Use of Telepresence in Informal Broadcasting over the Internet”, In Proc. of the ACM SIGMM 2004 Workshop on Effective Telepresence, pp. 12-15, 2004.
- [5] Y. Saito, Y. Isogai and Y. Murayama, “AdlivTV: An Interactive Broadcasting System for Audience-Driven Live TV on the Internet”, In Adjunct Proc. of EuroITV2008, pp. 217-221, 2008.
- [6] J. Jensen, “Interactive Television – A Brief Media History”, In Proc. of EuroITV2008, pp. 1-10, 2008.
- [7] D. Geerts, “Comparing Voice Chat and Text Chat in a Communication Tool for Interactive Television”, In Proc. of NordiCHI '06, pp. 461-464, 2006.
- [8] T. Coppens, L. Trappeniers, M. Godon, “AmigoTV: towards a social TV experience”, In Proc. of EuroITV2004.
- [9] H. Weisz, S. Kiesler, H. Zhang, Y. Ren, R. Kraut and J. Konstan, “Watching Together: Integrating Text Chat with Video”, In Proc. of CHI'07, pp. 878-886, 2007.
- [10] C. Harboe, C. Metcalf, F. Bentley, J. Tullio, N. Massey and G. Romano, “Ambient Social TV: Drawing People into a Shared Experience”, In Proc. of CHI'08, pp. 1-10, 2008.
- [11] D. Shamma, M. Bastea-Forte, N. Joubert and Y. Liu, “Enhancing online personal connections through the synchronized sharing of online video”, CHI'08 extended abstracts, pp. 2931-2936, 2008.
- [12] M. Drugge, M. Nilsson, R. Parviainen and Peter Parnes, “Experiences of Using Wearable Computers for Ambient Telepresence and Remote Interaction”, In Proc. of SIGMM 2004 Workshop on Effective Telepresence, pp. 2-11, 2004.