A Delivery Method for Compound Video Playback in Wireless Network

Kazuya Uyama[1], Morihiko Tamai[1], Yoshihiro Murata[1], Naoki Shibata[2], Keiichi Yasumoto[1], and Minoru Ito[1]

[1] Graduate School of Information Science, Nara Institute of Science and Technology
8916-5, Takayama, Ikoma, Nara 630-0192, Japan
{kazuya-u, morihi-t, yosihi-m, yasumoto, ito}@is.naist.jp

[2] Department of Information Processing and Management, Shiga University
Hikone, Shiga 522-8522, Japan
shibata@biwako.shiga-u.ac.jp

Abstract. As the progress of wireless communication technology and portable computing devices such as PDAs and cellular phones in recent years, users are requiring more advanced services through those terminals. One promising service is watching multiple video contents simultaneously with the specified layouts. In this paper, we propose a method to realize such an advanced video delivery service. When many users want to receive and play back different sets of videos with various combinations and layouts, it would be difficult to satisfy all users' requirements due restrictions on network resources and each terminal's computational resource. In the proposed method, we introduce proxies which receive multiple video contents from corresponding servers and produce a composite video from the received contents in real-time according to the layouts which users specify. However, if users require sets of video contents with slightly different layouts, multiple similar composite videos will be generated and the network bandwidth will be suppressed to transfer them. So, the proposed video delivery method identifies the common part of user requirements, and transmits to each user a composite video corresponding to the common part and remaining videos so that each user receives and plays back as small number of videos as possible. We have developed a greedy algorithm which calculates the set of videos (both composite and original) to be transmitted within the available wireless network bandwidth, so that the sum of satisfaction degrees of all users is maximized. Through experiments, we confirmed that our proposed method can achieve much higher user satisfaction degrees compared to the case that each user terminal receives multiple videos separately and plays them back in parallel.

1 Introduction

In recent years, digital terrestrial broadcasting services have started in many countries. In some of the countries, video broadcasting services dedicated for mobile terminals are becoming available. In Japan, digital video broadcasting service for mobile terminals called *1 segment broadcasting* based on ISDB-T [1] has already started since April, 2006. In Europe, US and Korea, similar services based on DVB-H [2], ATSC [3] and T-DMB [4] are going to be available. At the same time, mobile users are awaiting more advanced video services which allow them to watch multiple contents (e.g., soccer games and weather forecast) with the specified layout through the same screen like HDTV (we call a set of videos displayed on a screen a *compound video*).

In order to allow users to watch multiple videos at the same time, several studies have been conducted so far. [5] describes a method to locate proxies between content server and user terminal so that the resources required for video delivery meet limitation of network resources and user terminal resources by reducing video quality at the proxies. [6] is a method which uses proxies to produce a composite video from multiple video contents and forward the composite video to each user terminal. In [7], layout-based video delivery method is proposed, where user can specify any layout of multiple videos. In order to realize efficient video delivery for different layouts from multiple users, this method constructs trees of proxies so that multiple video contents are composed to one composite video incrementally at each intermediate proxy, aiming to minimize both computational resource at each proxy and network resource. These existing methods target wired environment where the purpose is to minimize computation resources at proxies and/or network resources of overlay network. Here, minimization of resources of wireless network and/or computation resources of mobile terminals is not considered.

In this paper, we propose a new video delivery method which allows mobile users to watch multiple videos with a specified layout in a typical mobile/wireless environment consisting of a large number of mobile terminals, multiple wireless access points (*APs*) which cover service area, multiple proxies and a wired network which connects APs and proxies.

Our proposed method maximizes the sum of the user satisfaction degrees (i.e., the ratio of the achieved quality of videos to the requested quality) by producing and forwarding composite videos common in layouts of multiple users (see Fig. 1).

The simplest method (referred to as *Method1*) to realize the target multiple video delivery service is to have each mobile terminal receive multiple video streams from their servers and play back those videos simultaneously after reducing the size, etc if necessary.

In Method1, video playback quality at a mobile terminal may drop down to a great extent or power consumption may become large since the terminal may receive surplus packets due to unnecessarily large size video and thereby resize by itself the received video to that specified in the layout before drawing video frames. This extra processing may cause so-called *drop frame*. To cope with the problem in Method1, many of the existing methods utilize proxies to resize or transcode videos before transmitting to mobile terminals. This method is referred to as *Method2*, hereafter. Although Method2 mitigates the computation power of mobile terminal to some extent, each mobile terminal still needs to play back multiple videos at the same time. This will introduce non-negligible processing overhead such as buffering and synchronization of multiple videos.

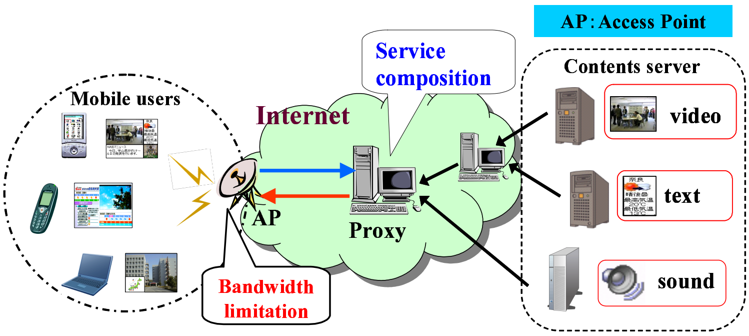

Fig. 1. Method for Delivering Contents

A more advanced method is to have proxies produce a composite video from multiple videos based on the requested layout and transmits the composite video to each mobile terminal. This method is used in [7] and referred to as *Method3*, hereafter. Method3 minimizes the computation power and/or quality degradation degree at each mobile terminal, whereas the number of composite videos to be transmitted through the same wireless network channel is limited (i.e., when many users request videos with different layouts, most of them cannot receive videos as they want).

In order to cope with the above problems, the proposed method identifies common part among layouts requested by multiple users, produces a composite video for the part, and transmits it through AP. The remaining videos in each layout are resized and/or transcoded at a proxy to videos with the quality specified in the layout and transmitted to the user terminal. Each user terminal receives a composite video and a few small videos, and plays them back according to the layout. In general smaller number of videos each user terminal receives, lower computation power is consumed at the mobile terminal. However, the problem to maximize the sum of the user satisfaction degree under limitation of network resources and mobile terminal resource is the combinatory optimization problem to find the best set of videos (composite videos and remaining videos) to be transmitted. To the problem, we propose a heuristic algorithm to calculate semi-optimal set of videos by introducing a mechanism to predict user satisfaction degrees before transmitting a set of videos.

We have conducted performance evaluation through simulation with multiple settings varying ratio of users who require similar layouts. As a result, we have confirmed that our proposed delivery method outperforms other simple methods in terms of the sum of the user satisfaction degree.

In the following Sect. 2, we define the problem of multiple video delivery for mobile users. Sect. 3 first describes simple methods for the multiple video delivery and then presents our proposed delivery method. In Sect. 4, our heuristic algorithm to derive a semi-optimal set of videos for broadcasting is given. Sect. 5 and Sect. 6 describe performance evaluation and conclusion, respectively.

2 Problem definition

In this section, we first describe the assumptions and the target environment, and then we give the formal definition of the problem.

2.1 Assumptions

In this paper, we assume that the proposed system is used in conjunction with PDA or mobile phone as user terminal, and IEEE 802.16 WiMAX or cellular phone infrastructure as a means of wireless communication. Each access point (AP) communicates with multiple user terminals, and each user terminal requests a set of videos with a layout (explained later), where the set is different from each other. In this paper, we assume these contents to be video only. Each user terminal is assumed to communicate with one access point at a time, and bandwidth between an access point and corresponding user terminals is limited. In this paper, we do not handle handover between access points in consequence of user terminal movement.

We assume that each user terminal accesses multiple video servers or a proxy via an AP. Each video is served from one server. Each proxy has the following three basic services: (1) Transcoding received video stream to stream with the specified picture size, framerate, and bitrate in real time; (2) Composing multiple videos into a video with specified layout; and (3) Transmitting processed video stream towards user terminal. We call the processed video *composite video*, and the raw non-processed video *atomic video*. Each user terminal is capable of changing size of received videos and displaying them simultaneously at any position on the screen, within the limits of its processing power. Thus, user terminals can receive multiple videos and play them back with any layout within available bandwidth and processing power. If it does not have sufficient processing power, framerate will decrease (which we call *drop frame*). Video stream is sent from AP to user terminals using broadcast, thus even if there are many terminals which receive a same stream, bandwidth usage is equal to the case when there is one receiver. For the sake of simplicity, we assume that bandwidth between video servers and proxies are unlimited, and proxies have unlimited computational power to process video streams.

Each user specifies the following information in the request.

– A set of videos
– A layout
– Framerates for each video

A layout includes the positions, picture sizes and framerates for displaying multiple atomic videos simultaneously. Fig. 2 is an example of layout. Each user sends a request including a layout to video servers via a proxy. Video servers send the corresponding atomic video to proxies without any processing. Proxies receive videos from servers, process and compose these videos so that the processed videos meet users' requests, and send them to user terminals.

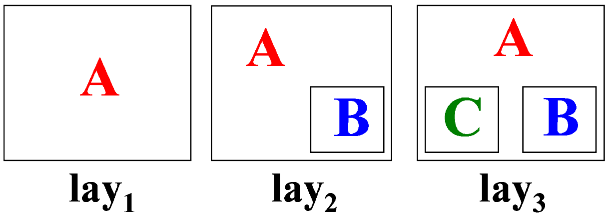

Fig. 2. An example of layout

Since each AP has bandwidth limitation, all users' requests cannot be always satisfied. If there is bandwidth shortage, the proposed method determines the set of delivered videos to maximize users' satisfaction.

2.2 Definition

The set of all available atomic videos is denoted as $C = \{c_1, ..., c_M\}$. A layout includes multiple rectangles to display videos as shown in Fig. 2. We call each rectangle *window*. To each window, picture size and position in the layout are specified. A layout is an ordered set of windows. A composite video generated by using atomic videos $e_1, ..., e_n \in C$ and a layout lay_i are denoted as $lay_i(e_1, ..., e_n)$. For example, a composite video generated by specifying atomic videos c_1 and c_2 to windows A and B of layout lay_2 in Fig. 2 is denoted as $lay_2(c_1, c_2)$.

The set of all users connecting to an AP is denoted as $U = \{u_1, ..., u_N\}$. User request from u_i is denoted as r_i. We can assume the number of APs is one, without loss of generality. The available wireless bandwidth for the AP is denoted by A_{bw}. Bandwidth required to deliver video g is denoted as $bw(g)$. The compound video displayed at user terminal u_i is called *user view* and denoted by v_i.

Satisfaction degree s_i for user u_i who has requested r_i and is viewing v_i is given by a function $s_i(r_i, v_i)$. The range of the function is between 0 and 1, and larger value means larger satisfaction of the corresponding user. There may be many definition of the function s_i, and we will give a definition based on ratio of drop frames in Sect. 4.

2.3 Problem

The definition of problem is as follows.

Given a set of user requests $\{r_1, ..., r_L\}$, determine a set D of videos delivered to the users which maximize the sum of user satisfaction degrees $\sum_{i=1}^{N} s_i$ within the limitation of wireless bandwidth $\sum_{d \in D} bw(d) \leq A_{bw}$.

3 Delivery Methods

In order to realize delivery of sets of videos to user terminals, the optimal set of delivering both composite and atomic videos should be determined to satisfy user requests.

As we already addressed in Sect. 1, the following three methods are considered as delivery mechanisms: (1) *Method1*: each user terminal receives atomic videos directly from content servers without using proxies; (2) *Method2*: each user terminal can receive atomic videos with reduced quality through proxies after resizing or transcoding the videos according to the user's request; (3) *Method3* and the improved version (proposed method): each user terminal can receive composite videos and/or atomic videos with reduced quality according to the user's request by asking proxies to compose multiple videos to one composite video or to resize/transcode atomic videos.

Although Method1 is the simplest approach, many picture frames of a received video can be dropped when a user terminal plays the video back due to increased load for receiving multiple atomic contents, resizing some of them and drawing picture frames on the specified layout. In Method2, in order to overcome the problem of Method1, picture size and/or framerate are reduced at a proxy server and forwarded to user terminal so that the terminal does not consume power for receiving surplus packets and resizing the picture frames and so on. Method3 further reduces computation power required at user terminals by allowing proxies to compose composite videos and forward them to user terminals. With this method, overhead caused by parallel playback of videos could be reduced to a great extent. However, since Method 3 may produce many composite videos which cannot be transmitted within the available bandwidth when users request videos with different layouts. So, we propose the improved version of Method3.

Method1: Delivering videos without proxies

Method1 is the simplest delivery method where each user terminal receives atomic videos specified in the user's request from content servers, changes size of received videos, and displays them on windows of the specified layout (Fig. 3).

For example, in Fig. 3, we suppose that there are three user terminals u_1, u_2 and u_3 which sent the following requests r_1, r_2 and r_3, respectively: $r_1 = lay_2(c_2, c_3)$, $r_2 = lay_2(c_3, c_1)$ and $r_3 = lay_3(c_3, c_2, c_1)$. Here, we suppose to use layouts lay_2 and lay_3 in Fig. 2.

With Method1, u_1 will receive atomic videos c_2 and c_3 from the corresponding content servers directly, and reduce the picture size of c_3, and displays c_2 and c_3 on windows A and B of layout lay_2 as shown in Fig. 3.

Method2: Delivering videos with reduced quality using proxies

In Method2, as shown in Fig. 4, proxies which are capable of reducing picture size and/or framerate, are available. Each proxy receives atomic videos from content servers, reduces picture size/framerate of received videos in real time, and

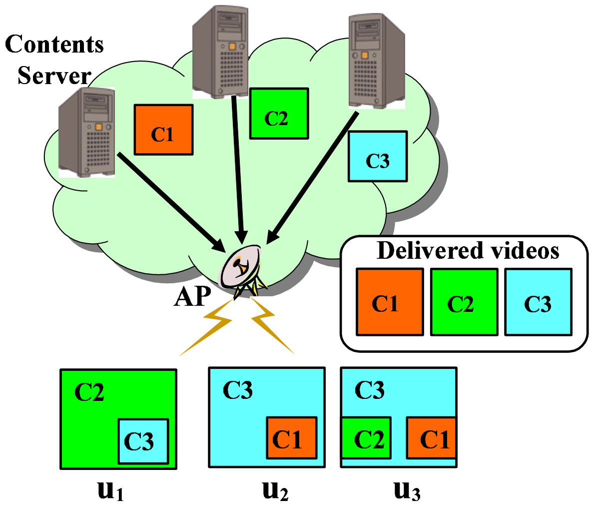

Fig. 3. Video delivery with Method1

forwards them to user terminals. Then, each user terminal displays the received videos on windows of the specified layout.

For example, in Fig. 4, user terminal u_1 receives atomic video c_2 from its content server and video c_3 with reduced picture size via the proxy server, and displays those videos on windows A and B of layout lay_2.

In Method2, processing overhead on user terminal will be lighter than Method1 since the terminal needs neither resize the picture size nor process surplus packets of original atomic video. However, similarly to Method1, user terminal needs to receive all of atomic videos in the request separately and play them back in parallel according to the specified layout.

Proposed Method: Delivering composite videos and videos with reduced quality using proxies

In our proposed method,, as shown in Fig. 5, each user terminal sends its request to a proxy. Similarly to Method3, in our proposed method, the proxy produces composite videos which meet user requests as much as possible within the bandwidth limitation, and transmits them to user terminals. The proposed method is different from Method3 in its flexibility that it allows each user terminal to receive and display one composite content, or to receive and display a set of composite and atomic videos and displays them with specified layout.

For example, in Fig. 5, user terminal u_1 receives a composite video which the proxy server produced from atomic videos c_2 and c_3 using layout lay_2. Similarly, u_2 receives a composite video composed of c_3 and c_1. On the other hand, user terminal u_3 receives an atomic video c_2 and a composite video of c_3 and c_1, and plays the two videos back in parallel, to satisfy u_3's request.

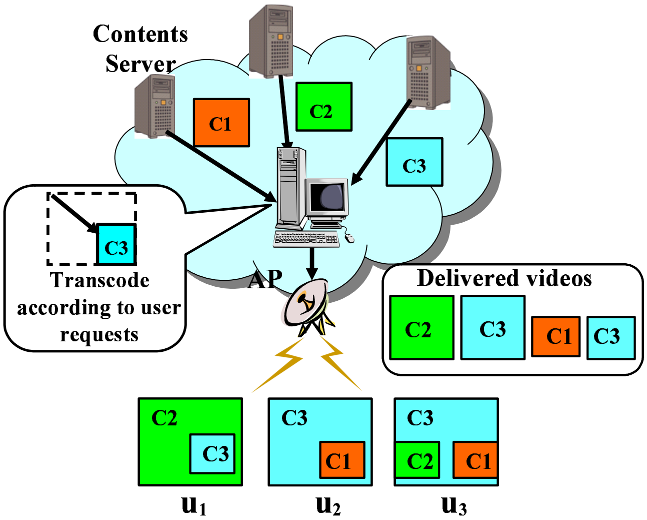

Fig. 4. Video delivery with Method2

4 Algorithm to derive optimal set of videos for broadcasting

In this section, we propose a greedy algorithm to calculate a semi-optimal set of videos for broadcasting which maximizes the sum of user satisfaction degrees. Each user's satisfaction degree depends on the quality of the displayed compound video at the user terminal. So, we also propose a method to predict the quality of the video achieved at the terminal for the given set of videos for broadcasting.

Either Method1, Method2 or our proposed method described in Sect. 3 may have each user terminal play back multiple videos in parallel. To play back multiple videos concurrently, each user terminal must receive and decode multiple video streams separately before drawing video frames. In general, as larger number of videos each terminal plays back, more computation power is required. As a result, if the computation power of user terminal is limited, the rate of drop frames called *drop frame rate* will increase as the number of videos increases.

In Method1, user terminals receiving videos with larger picture size (and/or framerates) than in their requests, have to decrease their sizes (and/or framerates) by themselves before drawing. In this case, drop frame rate will become larger than Method2. Dropped frames cause quality degradation in user view. In the proposed method, to prevent drop frames at user terminal, we take drop frame rate into consideration to define user satisfaction degree (details are given in Sect. 4.1). As explained in Sect. 2, the objective function of our target problem is defined as the sum of user satisfaction degrees. So, we need a technique to predict drop frame rates at all user terminals without playing back videos.

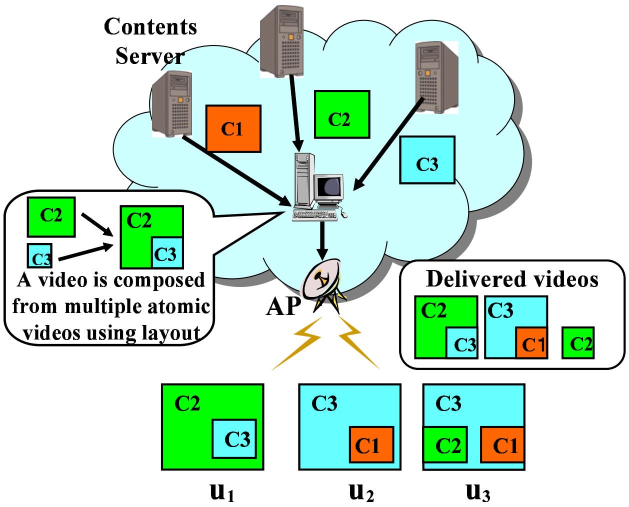

Fig. 5. Video delivery with Method3

In the following subsections, we present a method for predicting drop frame rate and a greedy algorithm to solve the target problem defined in Sect. 2.

4.1 Prediction of drop frame rate

Let p denote the number of pixels processed per unit of time while each terminal decodes videos. We assume that drop framerate denoted by $z(p)$ can be approximated by the following equation [1] :

$$z(p) = \alpha p + \beta \tag{1}$$

where, α and β are constant values specific for each terminal, and p is the number of pixels processed per unit of time. We can obtain these terminal specific constants by measuring drop framerates for different values of p with the Least Square Method. For a compound video q_i, let q_{ij} denote the video assigned in j-th window of layout used for generating q_i. For video q_{ij}, let $w_1(q_{ij})$ denote the product of picture size and framerate of the video received by u_i [2]. If video q_{ij}

[1] In general, several factors such as receiving packets, resizing picture frames, drawing pictures and so on must be considered as a load at terminal. However, these factors can be treated in a similar way to decoding.

[2] In Method1, each user terminal may receive videos with larger picture size (and/or framerate) than in the request, since content servers may retain only a file for each video content. In Method2, each user terminal may receive videos with smaller or larger picture size/framerate than the request due to bandwidth limitation. In our proposed method, q_{ij} may be part of a composite video. In this case, we use the window size in the layout and framerate of the composite video for q_{ij}.

is not received, $w_1(q_{ij}) = 0$. Let $w_2(q_{ij})$ denotes the product of the picture size and the framerate specified for the window corresponding to q_{ij} in the request. We define the user satisfaction degree as follows.

$$s_i = z_i(\sum_{j=1}^{n_i} w_1(q_{ij})) \times \sum_{j=1}^{n_i} \frac{Min(w_1(q_{ij}), w_2(q_{ij}))}{w_2(q_{ij})}/n_i \tag{2}$$

4.2 Greedy algorithm

The problem to find the optimal set of videos for broadcasting is a combinatory optimization problem, thereby it can be proved to be a NP-hard problem (due to page limitation, we omit the proof).

As a heuristic to solve this problem, we use a greedy algorithm since it is easy to implement and likely to run fast. We show pseudo code of our greedy algorithm to obtain a semi-optimal set of videos for broadcasting in Fig. 6. This algorithm can be used for each of Method1, Method2 and our proposed method. In the algorithm, D denotes the set of videos for broadcasting and $s_i(D - \{d\})$ denotes the satisfaction degree of user u_i when the set of videos for broadcasting is $D - \{d\}$.

As the first step of the algorithm, the initial set of videos D_0 is generated according to user requests. Elements of D_0 differ among Method1, Method2 and our proposed method.

In Method1, D_0 is the set of atomic videos included in user requests. For example, in Fig. 3, $D_0 = \{c_1, c_2, c_3\}$ where c_1, c_2 and c_3 are atomic videos.

In Method2, atomic videos with reduced quality are added to D_0 generated by Method1 if some of the users request videos with reduced quality in their layouts. For example, in Fig. 4, $D_0 = \{c_1, c_2, c_3, c_1^s, c_2^s, c_3^s\}$ where c_j^s denotes the video c_j with reduced quality.

Our proposed method adds, to D_0 generated by Method2, all composite videos which can be composed by assigning some of atomic contents to windows of all possible layouts. For example, if we assume that there are three possible layouts lay_1, lay_2 and lay_3 in Fig. 2, when user requests are those as shown in Fig. 5, $D_0 = \{c_1, c_2, c_3, c_1^s, c_2^s, c_3^s, lay_2(c_1, c_2), ..., lay_2(c_3, c_2), lay_3(c_1, c_2, c_3), ..., lay_3(c_3, c_2, c_1)\}$ where $lay_2(c_2, c_3)$ denotes a composite video generated by assigning atomic videos c_2 and c_3 to 1st and 2nd windows in the layout lay_2.

The set D_0 calculated above is assigned to D.

As the second step, the algorithm checks if the current set D satisfies the bandwidth limitation or not. If not, it tries to remove one element in D. It calculates the sum of user satisfaction degrees (denoted by sat) of the set $D - \{d\}$ for every element $d \in D$, and finds the element $\hat{d}$ which has the least impact on sat. Then $\hat{d}$ is removed from D and the second step is repeated until the bandwidth limitation is satisfied for D.

Algorithm $GetOptimalSet(D_0, A_{bw})$
$D := D_0$
while $D \neq \emptyset$ **and** $\sum_{d \in D} bw(d) > A_{bw}$ **do**
  // Find $d \in D$ which maximizes $\sum_i s_i(D - \{d\})$
  $max := -1$
  foreach $d \in D$ **do**
    $sat := \sum_i s_i(D - \{d\})$
    if $max < sat$ **then**
      $\hat{d} := d$
      $max := sat$
    endif
  next

  $D := D - \{\hat{d}\}$
next
return D
end

Fig. 6. Algorithm to derive optimal set of videos

5 Evaluation

In order to evaluate effectiveness of our method, we compared the sum of user satisfaction degrees among Method1, Method2 and our proposed method, varying distribution of user requests.

5.1 Prediction accuracy of drop frame rate

First, we obtained typical values of parameters α and β in function for drop frame rate $z(p) = \alpha p + \beta$ (described in Sect. 3), and then we measured drop frame rates using 9 videos with the different picture sizes and framerates. We obtained the drop frame rates using the Least Square Method. We used a PDA (SHARP Zaurus SL-C700, CPU: XScale 400MHz, Memory: 32MB, OS: Linux 2.4.28) in the measurement. Fig. 7 shows actual values of drop frame rates and the regression line from the measured values.

From Fig. 7, we can see that the average error between the predicted values obtained the regression line and the actual measurement values is about 12 %. There is still a room for improvement, but we believe it is enough for practical use.

5.2 Result on User Satisfaction Degree

In the experiment, we used configuration of 20 user terminals and 5 atomic videos. Each user selects one of the three layouts in Fig. 2 where the larger window's video quality is 320×240 pixels, 24 fps and 500 kbps, and that of the smaller

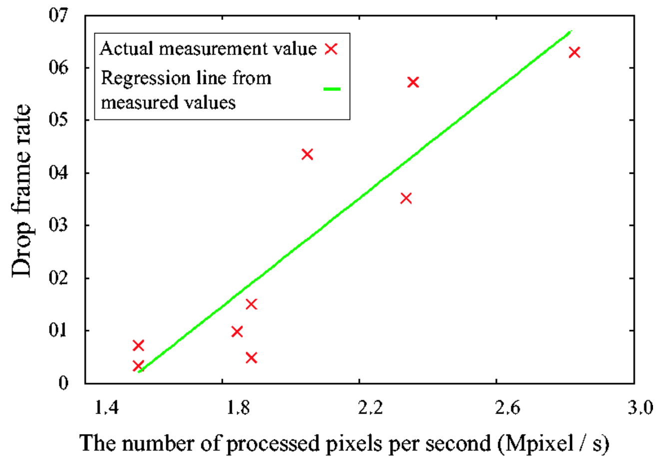

Fig. 7. Actual values of drop frame rate and the regression line from the measured values

window is 160×128 pixels, 24 fps and 350kbps. The available bandwidth for wireless network is set to 3000kbps.

The distributions of user requests are shown in Table 1. Labels A, AB and ABC in the table represent the windows of layouts in Fig. 2, respectively. The numbers in the table show how many different patterns exist in all user requests when we see only the specified window(s) in user requests. The number of each label is counted independently of the other labels. So, the total sum may exceed the number of users (i.e., 20). For example, in distribution $Dist_1$, numbers of different user requests in terms of video(s) to be displayed at the window(s) A, AB, AC and ABC are 5, 19 and 6, respectively. The total number of different user requests on the same layout depends on which video is displayed on each window of the layout. For example, the total number of user requests using layout lay_2 (which contains two windows A and B) is ${}_5C_2 \times 2! = 20$, and 19 patterns of them are involved in the distribution $Dist_1$. We intentionally set that the total number of different patterns decreases in the order of $Dist_1$, $Dist_2$ and $Dist_3$. Therefore, the number of users who require the same videos is smaller in the distribution $Dist_1$, while the number is larger in the distribution $Dist_3$.

Fig. 8 depicts the user satisfaction degrees achieved by Method1, Method2 and our proposed method described in Sect. 3 under the distributions $Dist_1, Dist_2$ and $Dist_3$.

From Table 1 and Fig. 8, we can see that our proposed method achieves highest user satisfaction degrees in all distributions of user requests. In Method2, user satisfaction degrees are almost constant for all distributions of user requests. This is because sets of videos for broadcasting become almost similar due to the bandwidth constraint. This trend can also be seen in Method1. In our proposed method, we can see that the user satisfaction degree is high, in the order of

Table 1. Distributions of user requests

Distribution patterns	A	AB, AC	ABC
Distribution $Dist_1$	5	19	6
Distribution $Dist_2$	5	11	5
Distribution $Dist_3$	5	5	2

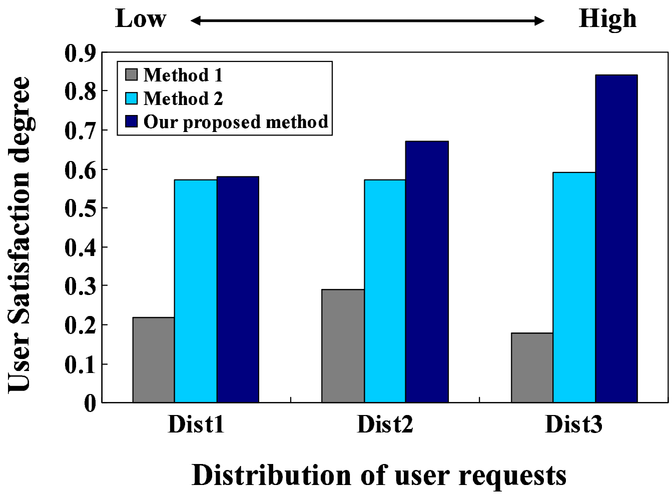

Fig. 8. Comparison of user satisfaction degrees

$Dist_3$, $Dist_2$ and $Dist_1$. This is because the number of users who require similar layouts is large, in the order of $Dist_3$, $Dist_2$ and $Dist_1$. From this result, our proposed method is especially effective when the number of users who require similar layouts is large.

6 Conclusion

In this paper, we defined an advanced multimedia service to deliver compound video to each user, and proposed a method to realize this service for mobile terminals on a wireless network. The proposed method maximizes the sum of user satisfaction degrees, considering both network bandwidth limitation and mobile terminal's computation power limitation. Through experiments, we showed that when a certain proportion of users require similar compound videos, our method which generates a composite video for common part of user requirements increases the sum of user satisfaction degree to a great extent.

Currently, we are implementing the proposed method on PDAs with IEEE802.11b network. The detailed performance evaluation in a real environment is a part of future work. In this paper, we do not consider the resource consumed in the wired network (i.e., computation power of proxies and network bandwidth between servers and proxies). We will treat this problem using the method we proposed in [8]. In addition, we are planning to extend the problem to deal with multi-hop data transfer by mobile terminals, data delivery through multiple access points, and so on.

References

1. Wikipedia: "Integrated Services Digital Broadcasting," `http://en.wikipedia.org/wiki/ISDB-T`.
2. DVB-H Global Mobile TV, `http://www.dvb-h.org/`.
3. ATSC Home Page, `http://www.atsc.org/`.
4. T-DMB Home Page, `http://eng.t-dmb.org/`.
5. J. Jin and K. Nahrstedt, "Source-based qos service routing in distributed service networks," *IEEE International Conference on Communications*, 2004.
6. J. Liang and K. Nahrstedt, "Service Composition for Advanced Multimedia Application s," *Multimedia Computing and Networking (MMCN'05)*, 2005.
7. K. Nahrstedt, B. Yu, J. Liang and Y. Cui, "Hourglass Multimedia Content and Service Composition Framework for Smart Room Environements," *Elsevier Journal on Per vasive and Mobile Computing*, 2005.
8. S. Yamaoka, T. Sun, M. Tamai, M. Yasumoto, N. Shibata and M. Ito, "Resource-Aware Service Composition for Video Multicast to Heterogeneous Mobile Users," *Proceedings of the 1st International Workshop on Multimedia Service Composition*, pp.37-46, 2005