Video Proxy Server Management Policy using Virtual Caching Technique

Jun Pyo Lee and Sung Han Park
Department of Computer Science & Engineering, Hanyang University
1271, Sangnokgu, Sa-3 Dong, Ansan, Kyungki-Do, Korea
E-mail: jplee@cse.hanyang.ac.kr, shpark@hanyang.ac.kr

Abstract: Due to the limited storage space in video proxy server, it is often required to replace the old video data which is not serviced for a long time with the newly requested video. This replacement causes the service delay and increase of network traffic. To circumvent this problem, we present a virtual caching technique for an efficient video streaming service in video proxy server. For this purpose, we employ a virtual memory. The requested video segment is not residing in the proxy server in first loaded in the virtual memory. The video in the virtual memory is stored in the proxy server depending on the consuming pattern by users. In addition, we propose an efficient replacement scheme in a proxy server. The simulation results show that the proposed algorithm performs better than other algorithms in terms of packet hit rate and number of packet replacement.

1. Introduction

In recent years, there has been an increasing interest in Video-on-Demand (VoD) service. A number of multimedia applications such as distance learning, digital library, video conferencing, and Entertainment-on-Demand rely on the streaming technique. In order to realize multimedia applications such as VoD over Internet, suitable quality of service (QoS) of the continuous media such as audio and video have to be guaranteed in accordance with users' requirements, available computing and network resources during the service session.

There are three main obstacles in video streaming over the Internet: inadequate bandwidth in the Internet to deliver a large number of concurrent video streams, limited capacity of a VoD server, and limited capacity of a client. In general, these obstacles can be solved by proxy server [1]-[7]. Proxy server can partially satisfy the need for rapid multimedia data delivery by providing multiple clients with a shared storing location. The requested videos are always delivered from the central video server through the proxy server to clients, thus the proxy server is able to intercept and store these videos to decrease the amount of video data that has to be delivered by the central server. In this context, if a requested video exists in a storage area in proxy server, clients get a stored video, which delivery time is typically reduced. The storing technique of proxy server is one of the key solutions to improve the performance of multimedia service systems on the Internet. However, since the storage capacity of proxy server does not have an infinite-capacity for keeping all the continuous video data, the challenge for the proxy server is to determine which videos should be stored or removed from the storage area of proxy server.

The well-known storing techniques of proxy server are interval caching and distance caching. These are memory based storing techniques and not appropriate to be applied to the disk storage which has limited bandwidth. The LRU (Least Recently Used) method is a recentness algorithm while the LFU (Least Frequently Used) and the TP (Time Popularity) methods [8] are the frequency one. Due to the large sizes of streaming media objects, these algorithms are not providing optimal performance in the internet environment.

In proxy server, most media objects should be stored partially to save the storage space efficiently. This leads us to a segment-based approach to proxy storing of large media objects [9]. The motivation of media segmentation is that we can quickly discard a big chunk of a stored media object that was once hot but has turned cold. In this way the storage manager of proxy server can quickly adjust to the changing reference patterns of partially stored object. However, to the best of our knowledge, none has dramatically utilized the effectiveness of media segmentation approach for large media objects and considered the time-variable request pattern by clients. We propose the video proxy server management policy using virtual caching technique as well as time-based approach to minimize the initial latency, central server load, and network traffic significantly.

2. Virtual Caching Technique

In this section, we present the virtual caching technique for an efficient multimedia streaming service in proxy server. We exploit the short-term temporal locality of two consecutive requests on an identical video. For this purpose, the proxy server loads the requested video segment which is delivered from the central server in virtual memory and forwards it to users directly, as shown in Figure 1.

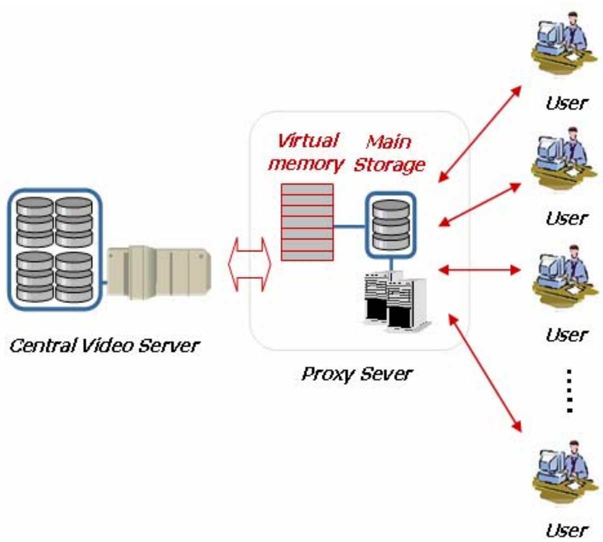

Figure 1. Virtual Caching Technique.

The requested video segment is not residing in the proxy server in first loaded in the virtual memory. But, if the same video segment is requested once again by a different user in case when the previous request is consuming the same video, we utilize the time distance between the earlier request and the latest request. Within this time distance, proxy server can preserve the segment which is delivered by the latest request in virtual memory. In addition, if the new request starts to utilize the video in virtual memory, proxy server can store the segments which are utilized by the new request in order to exploit the finite-capacity storage area efficiently.

If virtual memory space is insufficient for loading the segments in virtual memory, we calculate the request distance of each video in virtual memory. A request distance represents the time distance from current to expected request time. The request distance of a video i is given as shown in equation (1).

$$Request\ Distance_i = |\ MRRT_i + MRI_i - CT\ | \quad (1)$$

where MRRTi and MRIi denote the most recently request time of video i and the mean request interval of video i and CT represents the current time respectively.

We delete the last video segment which has maximum request distance for loading the new segments in virtual memory. If the virtual memory space is sufficiently for loading the new segment, the calculation of request distance will be ended.

3. Replacement Policy in Proxy Server

Since the storage space of proxy server does not have infinite-capacity storage for keeping all the continuous video data, the proxy server should have an appropriate replacement algorithm to make storage space for newly stored data in case when there is not free space enough to hold the new one. The challenge for the replacement algorithm is to determine which video segment should be stored or removed from the storage area of proxy server.

Replacement algorithm is one of the key components in proxy server and performed when the storage capacity is running out of space. The crucial aspect of replacement algorithm lies in selecting victim. In order to perform our replacement algorithm for selecting the victim, the proxy server periodically measures the time popularity of each video within a predefined interval. Based on this measured data, we can estimate the victim priority of each video in storage area, as shown in Figure 2.

In this paper, the lowest time popularity represents the highest victim priority. Thus, the video which has the highest victim priority is selected as a victim. After deleting the victim, we calculate a free space for storing the new video in storage area. If the storage space is sufficient for storing the new video, our replacement algorithm will be ended.

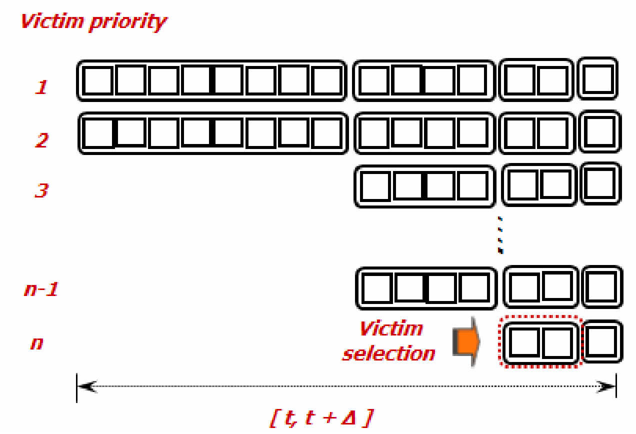

Figure 2. Victim selection based on priority.

Figure 3 depicts the selective storing method using victim priority. If the priority of newly requested video is higher than the last victim video, the proxy server bypass the newly requested video in order to avoid the repetition between the storing and deletion of a video.

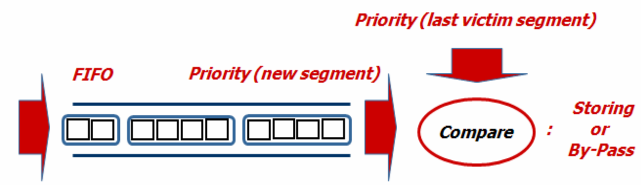

Figure 3. Selective storing method using victim priority.

Figure 4 depicts the deletion of segment using time interval to utilize the finite-capacity storage efficiently. If the segment of video is not requested by users within a predefined interval, this segment will be deleted automatically to provide enough storage space for newly stored video data.

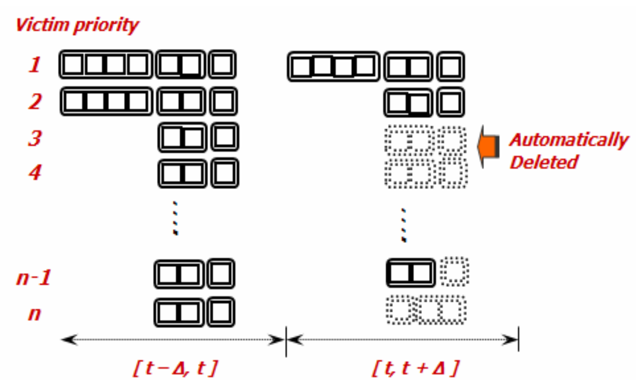

Figure 4. Deletion of segments using time interval.

4. Performance Evaluation

In general, hit rate has been used as the performance measurement of the storage schemes for traditional data such as text and image. But it is not proper for continuous media data. Thus, we use the packet hit rate, which can reflect the proxy server management policy. In addition, we also use the number of packet replacement in order to verify the efficiency of storage management policy more correctly.

4. 1 Simulation Parameters

We compare our proxy server management policy with the well-known algorithms such as Time Popularity and Distance-based through simulations.

In order to verify the effectiveness of the proposed proxy server management policy, we conduct simulation under two different kinds of segmentation approach: pyramid segmentation and fixed-length segmentation. Thus, a video is divided into segments of exponentially increasing length or predefined length, as shown in Figure 5.

(a)

(b)

Figure 5. Examples of media segmentation approach: (a) pyramid segmentation, (b) fixed-length segmentation (segment length = 2).

We consider a set of approximately 65Mbyte long videos and assume that videos are randomly requested by 500 users. In addition, we perform simulations to determine the optimal value of time interval size. The detail simulation parameters are shown in Table I.

Table 1. Simulation Parameters

Parameters	Value
Simulation time (T)	72 hours
Number of videos	1200 files
Video sizes	Approximately 65MB
Segmentation methods	Pyramid segmentation
	Fixed length segmentation(2-packet)
	Fixed length segmentation(3-packet)
	Fixed length segmentation(4-packet)
Storage size (GB)	5, 10, 15, 20, 25, 30, 35
Number of users	500 users
Number of packet	6500 packets/video
Time interval size (Δ)	25 minutes

4. 2 Simulation Results

In our simulation, to verify the effectiveness of our storage management policy, we conduct simulations under random access pattern when the storage sizes of 5, 10, 15, 20, 25, 30, and 35GB are used respectively.

Figure 6 shows that the packet hit rate of our video proxy server management policy compared with that of other well-known algorithms. The simulation results show that the proposed algorithm performs better than other algorithm such as Time Popularity and Distance-based methods in terms of packet hit rate.

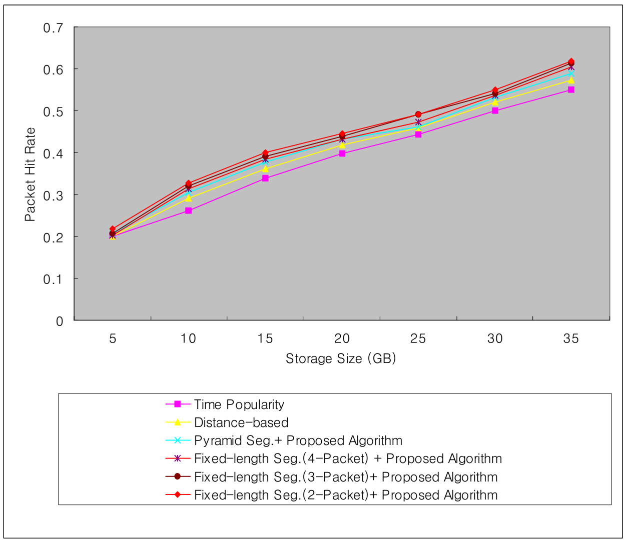

Figure 6. Comparison of packet hit rate under various storage size

Figure 7 shows the efficiency of our storage management policy, where the number of replacement of our algorithm is significantly fewer than that of other algorithms. This improvement may be due to the fact that the segmented video which represents a low request ratio can be deleted quickly in storage area. Therefore, we prove that the packet replacement overhead of our proxy server management policy is not significant compared to other algorithms.

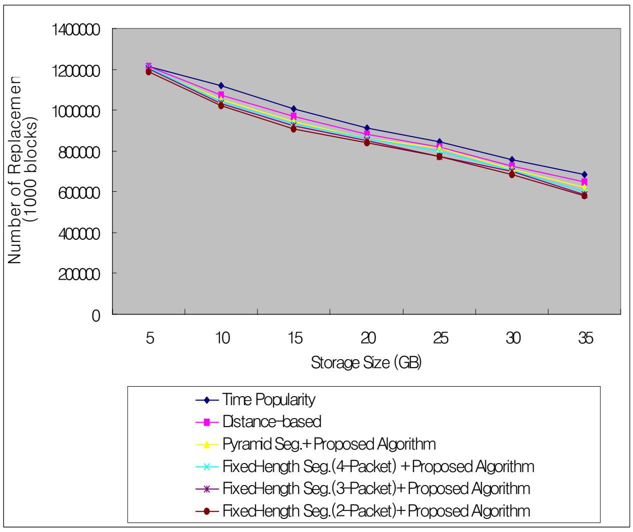

Figure 7. Comparison of number of replacement under various storage size

5. Conclusion

The storing technique of proxy server is one of the key solutions to improve the performance of multimedia applications on the Internet. By storing frequently accessed video data at a storage area, client perceived latency, central server load, and network traffic can be reduced significantly. However, since existing storing policies are for traditional data such as text and image, they are not suitable to continuous media data.

In this paper, we present the video proxy server management policy using virtual caching technique for an efficient multimedia streaming service. Thus, we can save storage space without degrading performance in proxy server. Through simulation, we evaluate the performance of our video proxy server management policy based on virtual caching technique and compared with other well-known algorithms. We demonstrate that the introduction of the concept of virtual caching leads to better performance.

Acknowledgement

This work was supported by grant No.(R01-2006-000-10876-0) form the Basic Research Program of the Korea Science & Engineering Foundation.

References

[1] Wanq. J. Z and Yu. P. S, “Fragmental Proxy Caching for Streaming Multimedia Objects,” *Trans. on Multimedia*, Vol. 9, pp. 147-156, Jan. 2007..

[2] Jun Pyo Lee and Sung Han Park, "A Cache Management Policy in Proxy Server for an Efficient Multimedia Streaming Service," *Proc. of IEEE Int. Conf. ISCE, pp. 64-68*, June 2005.

[3] Wei Liu, Chun Tung Chou, ZongKai Yang, and Xu Du, "Popularity-wise Proxy Caching for Interactive Streaming Media," *Proc. of IEEE Int. Conf. Local Computer Networks*, pp. 250-257, Nov. 16-18 2004.

[4] Kusmierek and David H.C. Du, "Optimizing periodic broadcast resource requirements with proxy," *Proc. of IEEE Int. Conf. Multimedia and Expo*, Vol. 2, pp. 1059-1062, June. 2004.

[5] Vakali A, and Pallis G, "Content Delivery Networks: Status and Trends," *IEEE Internet Computing*, Vol. 7, no 6, pp. 68-74, Dec. 2003.

[6] Wei-Hsiu Ma and David H. C Du, "Reducing Bandwidth Requirement for Delivering Video Over Wide Area Networks With Proxy Server," *Trans. on Multimedia*, Vol. 4, no. 4, pp. 539-550, Dec. 2002.

[7] Sonia Gonzalez, Angeles Navarro, Juan Lopez, and Emilio Zapata, "Load Sharing based on Popularity in Distributed Video on Demand Systems," *Proc. of IEEE Int. Conf. Multimedia and Expo*, Vol. 1, pp. 5–8, Aug. 26-29 2002.

[8] Kuan-Sheng Hsueh and Sheng-De Wang, "A packet-based caching proxy with loss recovery for video streaming," *Proc. of the 2002 Pacific Rim International Symposium on Dependable Computing*, pp. 185-190, Dec. 2002.

[9] Songqing Chen, Haining Wang, Xiaodong Zhang Shen B, and Wee S, "Segment-based proxy caching for Internet streaming media delivery," *IEEE Trans. on Multimedia*, Vol. 12, pp. 59-67, Sept. 2005.