

멀티미디어 객체 처리 성능 향상을 위한 웹 캐싱

나 윤 지* · 고 일 석**

요 약

웹 트래픽의 증가에 따라, 사용자에게 제공되는 웹 객체의 지연이 증가하고 있다. 웹 캐싱은 웹 기반 시스템의 성능을 향상시키는 하나의 방법이며, 캐시 저장영역의 효율적인 관리를 위한 많은 연구가 활발히 이루어지고 있다. 웹의 성능을 평가하는 중요한 척도는 사용자가 원하는 웹 객체를 제공하는 속도라 할 수 있다. 또한 웹 캐싱의 중요한 이슈는 웹 객체의 효율적인 관리를 통한 캐싱 시스템의 효율 향상이다. 이를 위한 해결책은 웹 객체의 이질성을 고려한 캐싱 전략을 세우는 것이다. 본 연구에서는 멀티미디어 객체 처리의 성능 향상을 위한 웹 캐싱 기법을 제안한다. 제안 기법은 멀티미디어 객체의 이질성을 기반으로 하고 있으며, 멀티미디어 객체의 캐싱 효율을 높여 캐싱 시스템의 성능을 높이고자 하는 것이다. 또한 실험에서 제안 시스템의 성능 분석을 통해 캐싱 시스템의 성능 향상을 검증하였다.

A Web Caching for Improvement of Processing Performance on a Multimedia Object

Yun Ji Na* · Il Seok Ko**

ABSTRACT

As traffic on the web increases, users are faced with increasing delays. A key performance measure for the web is the speed with which object is served to users. Web caching is the one of the key method that improve performance of Web base system. Also, many studies for efficient management of storing scope on cache are performed actively. An important issue in web caching is how to improve a efficiency of a caching system. For this, efficient web cache management is required. A solution for this is to set up the caching strategy that consider a heterogeneity of Web object. In this paper, We propose a new web caching technique for a performance improvement of a multimedia object process. The proposed method is based on a heterogeneity of Web object, and it increases performance of a caching system by increasing the caching efficiency of a multimedia object. In the experiment, We verified performance improvement of proposed web caching system through performance analysis of a proposed system.

키워드: 웹 캐싱(Web Caching), 멀티미디어 객체(Multimedia Object), 캐싱 시스템(Caching System)

1. 서 론

웹을 처리하는 HTTP 요청의 효율적인 처리는 네트워크 관리의 효율성을 결정하는 요인이 되고 있으며 이를 위해 HTTP의 요청만을 전담하여 처리하는 웹 캐시 서버의 사용이 점차 늘고 있다. 웹 캐싱은 자주 사용되는 인터넷의 객체를 캐시영역에 저장하여 인터넷의 성능을 향상시키며 웹 서버의 부하와 전체 네트워크의 트래픽을 감소시키고, 인터넷 사용자에게는 캐싱된 문서의 빠른 응답 시간을 제공한다 [1-4].

웹 캐싱 기법의 성능과 관련된 이슈는 한정된 저장공간의 효과적인 운영을 통한 웹 시스템 사용자의 응답속도 향상이

라 할 수 있다. 기본적으로 웹 캐시의 저장 공간은 한정되어 있으므로 사용 빈도가 높은 객체를 캐시의 저장영역에 저장하여야 사용자 응답속도를 향상시킬 수 있다. 하지만 한가지 더 고려해야할 사항은 객체가 웹 트래픽에 미치는 영향이다. 이런 관점에서 보면 멀티미디어 객체에 대한 관리 중요성이 부각된다. 일반적으로 다른 객체들에 비해 멀티미디어 객체의 크기가 크다. 이에 따라 멀티미디어 객체의 사용증가는 웹 트래픽의 증가를 가져오게 된다. 따라서 멀티미디어 객체 중심의 웹 캐싱 기법이 필요하다. 지금까지 많은 웹서버가 텍스트와 이미지 객체를 주로 서비스하였다. 하지만, 웹은 점차 멀티미디어 환경의 서비스로 발전하고 있으며 향후 웹 서비스의 50% 이상이 멀티미디어 서비스가 될 것이라 예측하고 있다[5].

기존의 연구 중에서 객체를 파일 타입별로 나누어 별도의 캐시 공간에 저장하고 관리하는 기법이 제안되었으나, 이 경

* 준 회 원 : 우송대학교 컴퓨터학과 교수

** 정 회 원 : 충북과학대학 전자상거래과 교수

논문접수: 2004년 3월 10일, 심사완료: 2004년 9월 9일

우 캐시 관리자의 구성과 관리에 대한 부담이 증가하게 된다. 일반적으로 사용 빈도가 비슷하고 크기가 비슷한 객체를 각각의 구분된 영역에 저장함으로써 한정된 공간의 캐시 영역의 할당을 복잡하게 만드는 단점이 발생하게 된다.

따라서 본 연구에서는 멀티미디어 객체 처리 성능 향상을 위한 웹 캐싱 기법을 제안한다. 제안기법은 멀티미디어 객체를 위한 캐시의 저장 공간을 할당하고, 이것의 관리를 통하여 멀티미디어 객체의 처리 성능을 향상시키는 기법이다. 제안 기법에 대한 실험 결과 멀티미디어 객체의 참조가 많은 웹 시스템에 대한 적중률 향상으로 사용자 응답속도의 성능 개선이 가능함을 확인할 수 있었다.

본 논문의 구성은 다음과 같다. 1장에서는 본 논문의 배경과 제안 기법의 기술로 본 논문의 주장사항과 효용성을 나타내었고, 2장에서는 대표적인 웹 캐싱 기법에 대해 살펴보았다. 3장에서는 실험을 통한 분석으로 웹 트래픽의 특징과 웹에서 차지하는 멀티미디어 객체의 중요성에 대해 다루었다. 4장에서 제안 기법에 대해 나타내었고, 5장에서는 제안 기법의 실험과 이에 대한 분석을 다루었다. 그리고 6장에서는 결론을 통해 본 연구의 효용성을 보인다.

2. 웹 캐싱

캐싱의 공통적인 관심사는 한정된 저장 영역의 효율적인 운용이다. 일반적으로 사용 빈도가 높은 객체를 캐시의 저장영역에 저장해야 캐시의 성능을 높일 수 있다. 따라서 효율적인 교체 기법은 캐싱의 성능을 향상시키는 중요한 요인이다[6, 7]. 현재까지 FIFO(First In First Out), LRU(Least Recently Used), LFU(Least Frequently Used), LUV, SIZE, LRUMIN과 같은 여러 교체 기법들이 연구되어 왔다. LRU는 새로운 객체가 저장 될 공간을 마련하기 위하여 가장 최근에 사용되지 않은 객체를 저장 영역에서 교체하는 기법이다. LRU는 최근에 오랫동안 사용되지 않은 객체가 미래에 참조되지 않을 가능성이 가장 높은 객체라는 특성을 이용하는 방식이며, 객체 참조의 국지성[4]을 이용한다. LFU는 참조의 빈도수를 기반으로 자주 사용되지 않은 객체를 우선적으로 교체하는 방식이다. SIZE는 새로운 객체가 저장 될 공간을 마련하기 위하여, 저장 영역에 저장된 객체 중에서 가장 큰 객체를 교체하는 기법이다. 인터넷 캐시는 하드웨어 캐시, 파일 시스템 캐시와 차이가 있으며, 인터넷 캐시는 객체의 크기가 가변적이고 교체의 단위가 웹 객체이다[8]. 따라서 크기가 큰 하나의 객체에 의해 크기가 작은 많은 객체가 저장 영역에서 제거되는 경우가 발생한다[9]. SIZE의 경우는 캐시 저장영역의 객체 중에서 가장 큰 객체를 교체함으로써 이 문제점을 개선하였다. LRUMIN은 LRU의 변형이다. 현재 연구되고 있는 대부분의 웹 캐싱 기법은 이러한 전통적인 기법을 기반으로 웹의 특성을 반영한 변형기법들이다.

〈표 1〉 객체 그룹 특성

| 구 분 | 참 조 특 성 | 참 조 빈도수 비율 | 총전송량 비율 |
|------|---|------------|---------|
| 그룹 1 | <ul style="list-style-type: none"> ◦ 텍스트 위주의 HTML 문서 ◦ 뉴스그룹, 전문가 그룹, 텍스트 위주의 커뮤니티·자료 공유 웹 사이트의 객체 | 높다 | 낮다 |
| 그룹 2 | <ul style="list-style-type: none"> ◦ 아이콘이나 웹 화면에 사용되는 GIF, JP(E)G, AVI 등의 비교적 작은 크기의 그래픽 및 멀티미디어 객체 ◦ 웹 화면을 통해 전송 받는 크기가 큰 그래픽 객체 및 MPEG 등의 동영상 객체 ◦ 젊은 연령의 사용자를 주된 대상으로 하는 웹사이트의 객체 | 낮다 | 높다 |

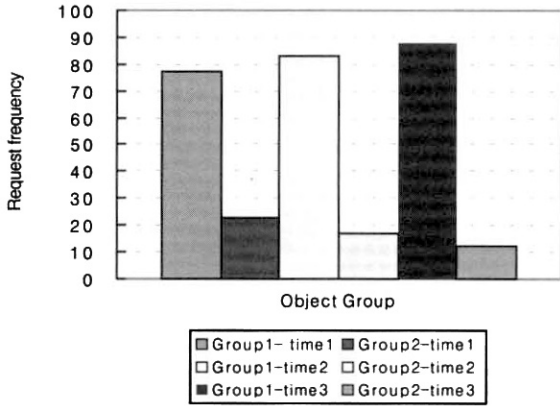
3. 웹에서의 멀티미디어 객체 특성

콘텐츠 제작 기술과 네트워크의 발전은 웹 상에서 더욱 다양한 콘텐츠의 서비스를 가능하게 하고 있다. 하지만, 이러한 발전은 한 편으로는 사용자의 멀티미디어 콘텐츠 요구를 증가시켜 웹 트래픽의 폭주를 가져오고 있다. 웹에서 다루고 있는 콘텐츠는 객체 단위로 이루어지고 있으며, 멀티미디어 콘텐츠의 경우 멀티미디어 객체 단위로 다루어진다. <표 1>은 객체의 그룹을 특성을 중심으로 나눈 것이다.

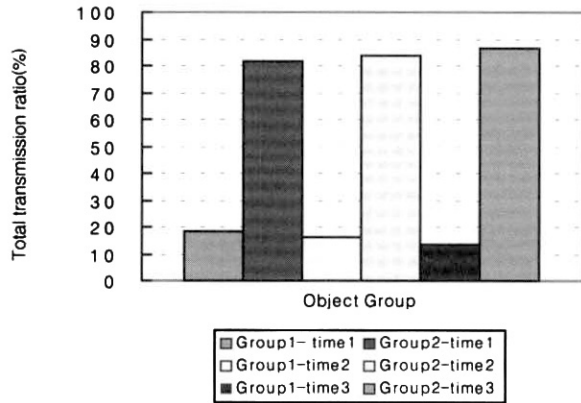
(그림 1)과 (그림 2)는 객체 그룹별 참조빈도 비율과 총전송량 비율을 나타낸 것이다. 웹 객체의 참조특성과 멀티미디어 객체 참조특성 분석을 위한 환경은 다음과 같다. 서버 기종은 IBM server xSeries 345이고, 기간은 2003년 4월 3일에서 6월 10일 사이에 실험과 분석을 수행하였다. 실험 시간은 주 4회 수업을 통해 주당 약 15시간이며, 31명을 대상으로 하였다.

(그림 1)은 세 개의 시간 구간에 대해 2개의 객체 그룹별 총 참조 빈도수 비율을 나타낸 것이고, (그림 2)는 (그림 1)에 대한 객체 그룹별 총 전송량 비율을 나타낸 것이다. 그룹 2는 아이콘이나 웹 화면에 사용되는 GIF, JP(E)G, AVI 등의 비교적 작은 크기의 그래픽 및 멀티미디어 객체와 웹 화면을 통해 전송 받는 크기가 큰 그래픽 객체, MPEG 등의 동영상 객체와 같은 크기가 큰 멀티미디어 객체 그룹이다. 전체 웹 객체 참조 빈도수 면에서는 멀티미디어 객체가 차지하는 비율이 그룹 1의 객체에 비해 작다. 하지만 총전송량 면에서는 80% 이상의 높은 비율을 차지한다. 그룹 1은 텍스트 위주의 HTML 문서와 뉴스그룹, 전문가 그룹, 텍스트 위주의 커뮤니티·자료 공유 웹사이트의 객체와 같이 비교적 같은 웹 객체 위주의 객체 그룹이다. 이 객체는 그룹 2의 객체에 비해 그 편차가 작고 전체 객체 참조빈도수 면에서 80% 이상의 높은 참조 특성을 나타낸다. 하지만 그룹 2의

객체에 비해 총전송량에서 차지하는 비율은 작다.



(그림 1) 객체 그룹별 참조빈도 비율



(그림 2) 객체 그룹별 총전송량 비율

멀티미디어 객체의 경우 객체의 사용 빈도가 낮은 경우라도 객체의 크기에 의해 웹 트래픽의 급격한 증가와 작은 크기의 객체를 캐시 밖으로 한꺼번에 교체시킨다. 캐시 적중률 관점에서 보면 작은 크기의 객체를 캐시에 저장하는 것이 캐시의 적중률을 향상시킨다. 따라서 멀티미디어 객체의 캐시부재는 캐시의 효율을 떨어뜨리는 주된 원인이 된다.

웹 캐싱의 효율성을 판단할 수 있게 하는 또 하나의 관점은 네트워크 트래픽 관점이다. 웹 캐싱은 전통적인 캐싱 기법과는 다르게 웹이라는 환경을 기반으로 하고 있어, 네트워크의 트래픽은 사용자 응답 시간에 직접적인 영향을 미치게 된다. 즉, 트래픽 관점의 웹 캐싱 효율은 사용자 응답속도의 개선이라 할 수 있다. 일반적으로 그룹 2의 멀티미디어 객체가 그룹 1의 객체에 비해 네트워크 트래픽을 증가시킨다.

따라서 멀티미디어 웹 객체의 크기 편차는 매우 크다. 따라서 멀티미디어 객체를 효율적으로 관리할 수 있는 웹 캐

싱 알고리즘이 필요하다. 또한 웹 객체는 사용자 개인의 참조 특성에 따라 가변적인 객체 참조 특성을 가진다. 이러한 웹 객체의 참조 특성은 객체의 특성을 반영한 웹 캐싱 알고리즘을 필요로 한다.

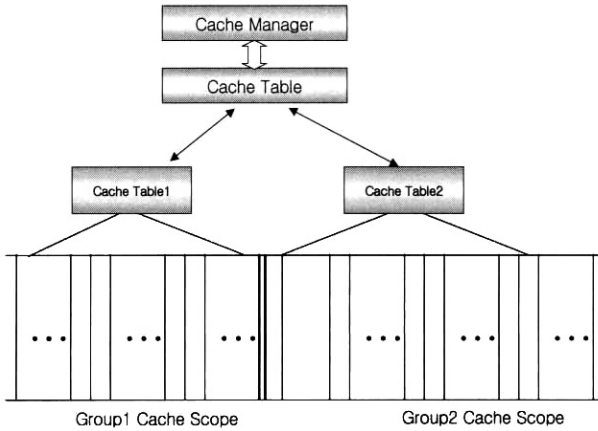
4. 제안 알고리즘

웹 캐싱 기법의 성능과 관련된 이슈는 한정된 저장공간의 효과적인 운영을 통한 웹 시스템 사용자의 응답속도 향상이라 할 수 있다. 또한 3장에서 살펴보았듯이 멀티미디어 객체가 웹에서 차지하는 비율이 점차 증가하고 있고 웹 트래픽에 미치는 영향이 크기 때문에, 본 연구는 멀티미디어 객체의 효과적인 관리를 통한 트래픽 감소로 인한 사용자 응답속도의 개선이 주안점이다. (그림 3)은 제안 기법의 캐시 구조를 나타낸 것이다. 캐시의 영역은 3장의 실험결과와 같이 그룹 1과 그룹 2, 두 개의 영역으로 구분된다. 캐시 테이블 1과 캐시 테이블 2는 이 두 개의 영역을 각각 관리하는 테이블이다.

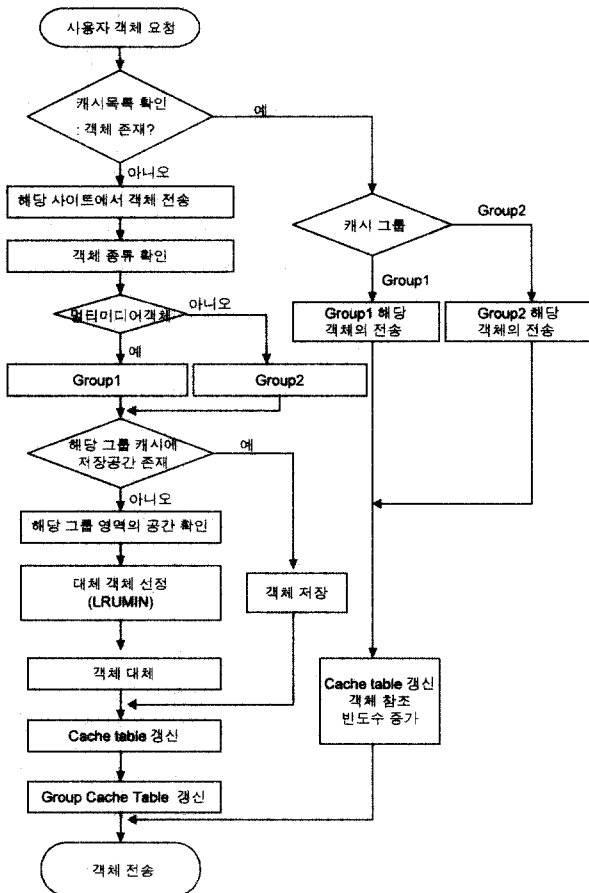
이 경우 발생할 수 있는 트레이드 오프는 ① 한정된 캐시 공간에 대한 멀티미디어 객체 이외의 객체에 대한 저장 공간 감소로 인한 객체적중률의 감소, ② 캐시 관리자의 복잡성의 증가이다. 하지만 제안 기법의 실험에서 볼 수 있듯이 사용자의 응답속도 개선을 통해 이 두 가지 트레이드 오프를 보상할 수 있다.

(그림 4)는 캐시 교체 알고리즘을 나타낸 것이다. 클라이언트의 객체 요청이 들어오면 캐시 관리자는 해당 객체그룹의 목록에서 존재 여부를 확인한다. 이때 캐시의 저장 영역에 해당 객체가 있을 경우 사용자가 요구한 객체를 사용자의 IP로 전송하게 된다. 이때 해당 객체는 캐시 테이블에서 참조빈도수가 1 증가한다. 캐시 목록에서 요청된 객체를 찾을 수 없을 경우 해당 URL의 인터넷 서버에 객체 서비스를 요청하여 전송 받는다. 전송 받은 객체는 객체에 따라 그룹 1 또는 그룹 2로 분류되고 캐시 관리자는 해당 그룹의 캐시 영역에 이 객체를 저장할 공간이 있는지를 확인한다. 저장할 공간이 있을 경우 객체를 해당 캐시 영역에 저장하고 캐시 목록을 갱신하고, 없을 경우 해당 그룹의 캐시 영역에 대해 캐시 교체 기법을 이용하여 객체를 교체한 후 캐시 목록을 갱신한다. 이때 사용할 수 있는 웹 캐시 교체 기법은 다양하지만 여기에서는 LRUMIN을 사용한다. LRUMIN 기법은 LRU 기법의 변형이며 SIZE 기법과 같이 LRUMIN은 크기가 큰 객체에 의해 크기가 작은 많은 객체가 저장 영역에서 제거되는 경우를 줄일 수 있다고 알려져 있다. LRUMIN의 동작은 다음과 같다. 먼저, 이 기법은 새롭게 들어오는

객체의 크기를 S로 설정한다. 다음으로, S보다 큰 여유 공간이 없으면 S/2 이상의 큰 객체들 중에서 LRU로 제거한다. 이때, S/2 이상의 큰 객체가 없으면 S/4 이상의 큰 객체들 중에서 LRU로 제거한다. LRUMIN은 이러한 방식으로 새로운 객체의 저장을 위한 여유 공간이 생길 때까지 이 과정을 계속해서 반복한다.



(그림 3) 캐시 구조



(그림 4) 캐시 교체 알고리즘

5. 실험 및 분석

사용자 응답속도 실험을 통해 제안 알고리즘의 성능을 평가할 수 있다. 실험은 3장의 실험 결과 도출한 세 가지 그룹별 참조빈도수비율, 총전송량비율에 대해 이루어졌다. <표 2>는 이 실험에 사용된 그룹 1과 그룹 2의 참조에 의한 총전송량을 기준으로 7.7 : 2.3, 8.3 : 1.7, 8.8 : 1.2의 세 가지 모델에 대해 이루어졌다.

<표 2> 실험모델 특성

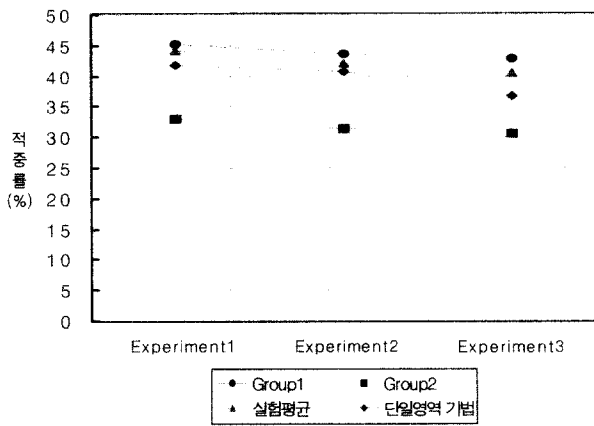
| 구 분 | 그룹별 비율 | |
|------|-----------------------------|-----------------------------|
| | 그룹 1 참조빈도 비율 (총전송량비율) | 그룹 2 참조빈도 비율 (총전송량비율) |
| 실험 1 | 7.7(1.8) | 2.3(8.2) |
| 실험 2 | 8.3(1.6) | 1.7(8.4) |
| 실험 3 | 8.8(1.4) | 1.2(8.6) |

(그림 5)는 객체 적중률을 나타낸 것이다. 그룹 1의 경우 멀티미디어 객체가 많은 실험 3의 객체 적중률이 실험 1보다 낮게 나왔다. 이것은 실험 1이 실험 3에 비해 크기가 작은 객체의 요구가 차지하는 비율이 적기 때문이다. 이것은 기존의 연구에서도 작은 크기의 객체를 캐시영역에 많이 저장하는 것이 객체 적중률을 높인다는 실험 결과를 뒷받침하는 결과라 할 수 있다. 또한 실험 평균 적중률 값을 보면 40% 이상의 값이 나온다. 이것은 각 실험에서 그룹 1의 객체 참조빈도수가 그룹 2의 객체참조빈도수에 비해 높기 때문이다. 또한 단일 영역 기법의 실험에서는 실험의 평균값보다 적중률이 낮아진 것을 알 수 있다. 이것은 단일 영역 기법의 경우 그룹 2의 멀티미디어 객체의 부재에 의한 그룹 1 객체의 부재가 제안 알고리즘 보다 많이 발생하기 때문이다. 또한 단일 영역 기법의 경우 그룹 2 객체가 많이 포함된 실험 3에 대해 가장 낮은 객체 적중률을 보임을 알 수 있다.

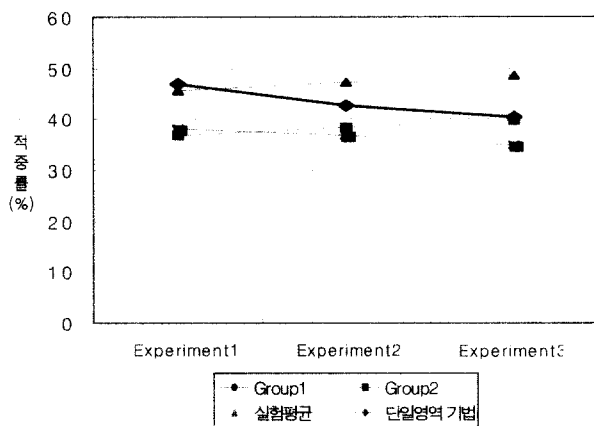
(그림 6)은 객체의 크기를 고려한 적중률을 나타낸 것으로 각 객체의 크기 평균값을 각 객체의 적중률에 곱한 값의 평균을 낸 것이다. 그룹 1에 대한 실험결과, 그룹 1의 비율이 가장 높은 실험 1의 경우 앞의 객체적중률 실험보다 다소 높은 적중률을 보인다. 하지만 그룹 1 객체가 감소할수록 앞의 객체적중률보다 적중률이 낮아짐을 알 수 있다. 그룹 2의 경우는 이 그룹 1에 대한 실험과는 다른 결과를 얻었다. 전체적으로는 앞의 실험에 비해 적중률이 높아졌으며, 특히 그룹 2 객체의 비중이 높을수록 적중률이 더욱 높아졌음을

알 수 있다. 결과적으로 객체의 크기를 반영할 경우 제안 알고리즘 전체의 객체적중률이 높아졌다. 하지만 단일 영역 기법의 경우 제안 기법보다 그룹 2 객체의 적중률이 낮기 때문에 전체적으로 적중률이 감소했다.

객체의 크기는 웹 기반 시스템의 트래픽에 영향을 미치는 요인이 된다. 크기가 큰 객체에 대한 캐싱 능력 향상은 웹 트래픽을 감소시켜 사용자 응답 시간을 감소시킬 수 있기 때문에 이 결과는 웹 캐싱 시스템에서 앞의 객체 적중률에 비해 더욱 큰 의미를 갖는다. 또한 제안 기법이 적중률 향상을 통해 멀티미디어 객체의 처리 성능을 개선할 수 있음을 기대할 수 있다.



(그림 5) 객체적중률 비교



(그림 6) 객체크기 반영한 적중률

6. 결 론

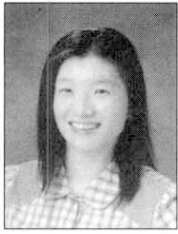
웹 트래픽에서 멀티미디어 객체가 차지하는 비중이 증가함에 따라 멀티미디어 객체에 대한 관리의 중요성이 부각되었다. 본 연구에서는 멀티미디어 객체 처리 성능 향상

을 위한 웹 캐싱 기법을 제안하였다. 제안기법은 멀티미디어 객체를 위한 캐시의 저장 공간을 할당하고, 이것의 관리를 통하여 멀티미디어 객체의 처리 성능을 향상시키는 기법이다.

제안 기법에 대한 실험 결과 멀티미디어 객체 비중이 높아질수록 제안 기법의 적중률 향상을 확인 할 수 있었고, 이에 따라 제안 알고리즘이 멀티미디어 객체 처리 성능 향상으로 사용자 응답속도를 개선함을 기대할 수 있었다.

참 고 문 헌

- [1] G. Barish, K. Obraczka, World Wide Web Caching : Trends and Techniques. IEEE Communications, Internet Technology Series, May, 2000.
- [2] H. Bahn, S. Noh, S. L. Min and K. Koh, "Efficient Replacement of Nonuniform Objects in Web Caches," IEEE Computer, Vol.35, No.6, pp.65-73, June, 2002.
- [3] N. Niclausse, Z. Liu, P. Nain, "A New and Efficient Caching Policy for the World Wide Web," Proc. Workshop on Internet Server Performance(WISP 98), pp.94-107, 1998.
- [4] C. Aggarwal, J. Wolf and P. Yu, "Caching on the World Wide Web," IEEE Trans. Knowledge and Data Engineering, Vol.11, No.1, pp.94-107, 1999.
- [5] S. Sahu, P. Shenoy, D. Towsley, "Design Considerations for Integrated Proxy Servers," Proc. of IEEE NOSSDAV'99, pp.247-250, June, 1999.
- [6] J. Wang, "A Survey of Web Caching Schemes for the Internet," ACM Computer Communication Review, 29, pp.36-46, October, 1999.
- [7] C. D. Murta, Virgilio A. F. Almeida, Jr. W. Meira, "Analyzing Performance of Partitioned Caches for the WWW," In Proceedings of the Third International WWW Caching Workshop, Manchester, England, June, 1998.
- [8] Annie P Foong, Yu-Hen Hu, Dennis M Heisey, "Web Caching : Locality of References Revisited," IEEE International Conference on Networks (ICON'00), Singapore, September, pp.05-08, 2000.
- [9] L. Rizzo, L. Vicisano, "Replacement Policies for a Proxy Cache," IEEE/ACM Trans. Networking, Vol.8, No.2, pp. 158-170, 2000.
- [10] S. Williams, M. Abrams, C. R. Standridge, G. Abhulla and E. A. Fox, "Removal Policies in Network Caches for World Wide Web Objects," Proc. 1996 ACM Sigcomm, pp.293-304, 1996.



나 윤 지

e-mail : yjna2967@korea.com

충북대 컴퓨터공학(공학박사)

뉴욕공대(NYIT) 대학원 Communication

ART전공(컴퓨터그래픽스, 방송기획)

충북대 컴퓨터공학(공학석사)

경북대 생명공학(이학사)

현재 우송대학교 컴퓨터학과 초빙교수

관심분야 : 멀티미디어개발 및 멀티미디어시스템 성능향상, 멀티미디어데이터베이스, 웹기반 시스템



고 일 석

e-mail : isko@ctech.ac.kr

연세대 컴퓨터산업시스템공학(박사수료)

미)USIU 경영학과(MBA)

경북대 컴퓨터공학(공학석사)

경북대 컴퓨터공학(공학사)

현재 충북과학대학 전자상거래과 조교수

관심분야 : 웹시스템개발&성능향상, 게임기술, 지능형웹시스템, 기업기술평가, 기업전략경영컨설팅