NAND 플래시 메모리 상에서 페이지 공유를 통한 대용량 멀티미디어 파일 고속 편집저장 기법

고석영*○, 정승완*, 남영진**, 서대화*
*경북대학교 전자전기컴퓨터학부
**대구대학교 컴퓨터·IT공학부
syko@ee.knu.ac.kr, tmdrod@ee.knu.ac.kr, yjnam@daegu.ac.kr, dwseo@ee.knu.ac.kr

The Fast Editing and/or Storing Technique for Large-sized Multimedia Files with Page Sharing on NAND Flash Memory

Seok Young Ko*, Seung Wan Jung*, Young Jin Nam**, Dae-Wha Seo*
*School of Electrical Eng. & Computer Science, Kyungpook National University
**School of Computer & Information Technology, Daegu University

요 약

본 논문은 NAND 플래시 메모리를 기반으로 한 멀티미디어 휴대 장치에서 동영상 파일을 고속으로 편집하고 동시에 필요한 저장 공간 소모를 최소화 하는 기법을 제안한다. 동영상 파일 편집 후 저장에 있어서 현재의 NAND 플래시 파일 시스템들은 편집 내용을 빈 공간에 새로 저장하기 때문에, 대용량의 파일일수록 많은 시간과 저장 공간이 소모 된다. 본 논문에서 제안하는 기법은 동영상 편집 후 파일 간에 중복되는 데이터를 새로 저장하지 않고 공유하도록 하면서 소모되는 시간과 공간을 최소화 한다. 이를 위해 파일 간 공유 되는 데이터를 관리하는 페이지 공유 맵을 설계하고 이를 이용한 동영상 고속 편집저장 기법을 제시한다.

1. 서론

최근 컴퓨터의 이동성이 강조되면서 MP3P, PDA, 휴대폰, PMP 등의 휴대용 멀티미디어 기기가 증가하고 있고, 이러한 휴대용 장치에서는 대용량의 멀티미디어 콘텐츠를 저장하고 처리하기 위한 효율적인 저장장치가 필요하다[1]. 휴대용 기기를 위한 저장 장치로는 플래시 메모리와 하드 디스크가 주로 이용되는데, 그 중 NAND 플래시 메모리의 저전력 및 내구성의 장점과 최근 대용량화 및 가격 하락으로 인해 그 수요가 급속도로 증가하고 있다[2].

휴대용 멀티미디어 기기의 고성능화로 인해 캠코더 뿐 아니라 디지털 카메라, 휴대폰 등에서도 동영상 기능이 널리 제공되고 있다. 최근 UCC의 활성화로 인해 많은 사람들이 이런 휴대용 멀티미디어 기기를 이용해 동영상을 촬영하고 있다. 그리고 인터넷을 통하여 주위 여러 사람들과 촬영한 동영상을 공유하기도 한다. 이런 동영상 기능의 대중성은 휴대용 기기에 있어 멀티미디어 성능의 중요성을 부각시킨다.

동영상 촬영 뒤, 보관 또는 공유 할 때 동영상 편집 과정을 거칠 수 있다. 전문가와 달리 일반 대중들은 편집에 필요한 컴퓨터의 사용을 어려워하기 때문에, 이점을 고려해 휴대용 멀티미디어 기기 업체들은 장치 내에 간편한 동영상 편집 기능을 제공한다.

동영상 파일 편집저장에서, 편집은 동영상 파일을 저장 구간과 삭제 구간으로 나누는 것이다. 이는 파일 내용을 수정하는 일반적인 파일 편집과 틀리다. 그리고 저장은 편집한 내용을 기존 파일에 덮어 쓰거나 새 파일로 저장하는 것이다. 기존 NAND 플래시 파일 시스템에서는 먼저 편집된 내용 중 저장 구간에 해당하는 데이터를 NAND 플래시 메모리의 빈 공간에 복사한 뒤, 새 파일로 만든다. 그런 뒤, 만약 덮어 쓰기를 할 경우에는 기존 파일을 삭제하는 과정이 추가 된다. 기존 파일 시스템에서의 이런 편집 과정은 동영상 파일 편집에서 효율적이지 않다. 일반적인 파일 편집은 파일 내용 자체를 수정하기 때문에, 편집된 내용을 새로 저장해야 한다. 하지만 동영상 파일 편집에서는 내용은 수정되지 않고, 단지 저장 구간과 삭제 구간으로 파일을 나눌 뿐이다. 따라서 기존 NAND 플래시 파일 시스템의 동영상 파일 편집은 수정하지 않은 파일 내용을 복사하기 때문에, 중복되는 데이터가 존재하게 된다. 예를 들면 A파일의 내용 중 필요한 부분만을 선택하여 B파일에 저장한다고 하자. 여기서 B파일의 내용은 A파일에 포함되는 중복 데이터가 된다. 따라서 본 논문에서는 이런 동영상 파일 편집 특성을 고려하여, 파일 간 데이터 공유를 통해 동영상 편집을 고속으로 처리하고 저장 공간의 소모를 최소화하는 기법을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 NAND 플래

▪ 본 연구는 2008년도 경북대학교 BK21사업과 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2008-C1090-0801-0045)

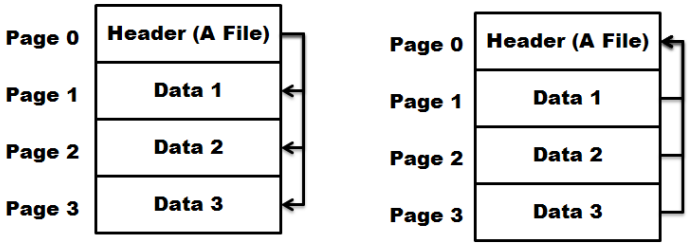
시 메모리의 특성과 기존의 NAND 플래시 파일 시스템에 대해 기술한다. 3장에서는 본 논문에서 제안하는 동영상 파일 편집을 고속으로 처리하고 저장 공간 소모를 최소화 하는 기법을 설명한다. 4장에서는 이 논문의 결론과 향후 과제를 제시한다.

2. 관련 연구

NAND 플래시 메모리는 읽기/쓰기 연산이 페이지 단위로 이루어지고, 블록을 먼저 삭제하고 쓰기를 수행할 수 있다. 이는 동일 페이지에 덮어 쓰기가 되지 않기 때문에 유의해야 하며, 쓸모가 없어진 페이지들은 더티(Dirty) 페이지로 설정되고 이후 블록 삭제를 통해 재사용할 수 있다[1].

NAND 플래시 파일 시스템에서는 일반적으로 파일 헤더와 파일 데이터로 하나의 파일이 구성된다. 여기서 파일 헤더와 데이터 간의 관계에 따라 그림 1에서처럼 직접 사상 또는 역 사상 방식으로 구분할 수 있다. NAND 플래시 메모리를 위한 대표적인 파일 시스템인 YAFFS[2]는 이 중에서 역 사상 방식으로, 이는 파일 데이터가 파일 헤더를 가리키는 그림 1.(b)와 같은 형태이다. 반면에 최근에 나온 NAND 플래시 메모리 파일 시스템 중 CFFS[3]는 그림 1.(a)와 같이 파일 헤더가 관련 데이터들을 가리키는 직접 사상 방식을 따르고 있다. 파일의 데이터 위치, 순서와 같은 파일의 구조를 직접 사상 방식에서는 파일 헤더에서 모두 관리하는 반면, 역 사상 방식에서는 각 데이터 페이지의 스페어 영역 정보로 관리하게 된다.

사상 방식과 관계없이 기존의 NAND 플래시 파일 시스템은 일반적인 파일 편집과 마찬가지로 동영상 파일 편집저장도 편집 내용을 빈 공간에 새로 복사하는 방법을 취하고 있다. 예를 들어 그림 1.(a)에서 파일 A 편집 후 저장 구간이 데이터 1, 3이라 하면, 이 데이터를 NAND 플래시 메모리 빈 공간에 기록하고 새 파일로 만든다. 그런 뒤, 덮어 쓰기로 저장한다면 기존 파일 A를 삭제하게 된다.



(a) 직접 사상 (b) 역 사상

그림 1. 파일 헤더와 관련 데이터와의 관계

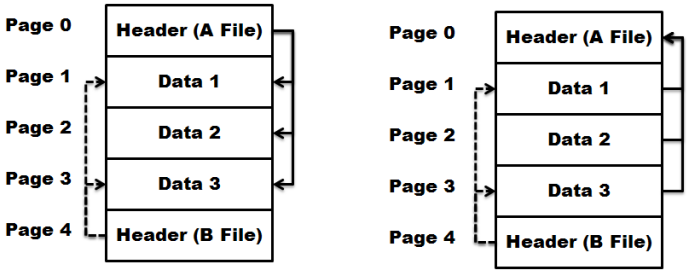
3. 페이지 공유 기반의 고속 편집저장 기법

본 논문에서 제안하는 고속 편집저장 기법은 파일 편집 후 저장에 있어 파일 간 중복되는 데이터를 서로 공유도록 하여, 고속 처리 및 공간 소모의 최소화가 가능한 동작이다. 이 동작은 먼저 파일 공유를 위해 파일 간 중복되는 데이터 페이지들을 파일 헤더를 통해 공유하고 페이지 공유 맵을 만들어 공유 정보를 관리하게 된다.

3.1 직접 사상과 역 사상에서 파일 공유

파일 간 데이터 공유를 위해서는 각 파일 헤더에서 공유 데이터 페이지를 동시에 가리켜야 한다. 하지만 NAND 파일 시스템에서 직접 사상과 역 사상 방식의 적용에는 고려해야 할 차이점이 있다. 그림 1.(a)에서 직접 사상인 경우, 데이터 1을 공유하기 위해서는 새로운 파일 헤더에서 단순히 그 데이터 페이지를 가리켜 주면 된다. 반면에 그림 1.(b)와 같이 역 사상인 경우, 데이터 1에서 기존 파일 헤더와 새로운 파일 헤더 모두를 가리켜야 한다. 이를 위해서 데이터 페이지의 스페어 영역을 수정해 주어야 한다. 하지만 NAND 플래시 메모리 특성상 동일 페이지에 덮어 쓰기가 되지 않는다. 결국 수정한 데이터를 빈 공간에 새로 기록해야 해야 하고 이는 결국 기존의 NAND 파일 시스템에서 편집 내용을 새로 기록하는 것과 같다. 또한 페이지 스페어 영역 용량의 제한으로 인해 여러 공유 정보를 기록하는데 한계가 있다. 따라서 역 사상 방식은 파일 공유에 있어 적합하지 않기에 파일 시스템에서 추가적인 보완이 필요하다.

본 논문에서는 역 사상 방식의 파일 시스템에서 추가적인 보완으로 직접 사상 방식의 파일 헤더 도입을 제안한다. 그럴 경우 데이터 공유는 그림 2와 같이 된다. 직접 사상인 경우에는 그림 2.(a)와 같이 단순히 공유되는 내용을 여러 파일 헤더에서 동시에 가리키면 된다. 역 사상인 경우에는 그림 2.(b)와 같이 직접 사상 방식의 파일 헤더를 도입하여, 그 파일 헤더에서 직접 공유 데이터를 가리키도록 한다.



(a) 직접 사상 (b) 역 사상

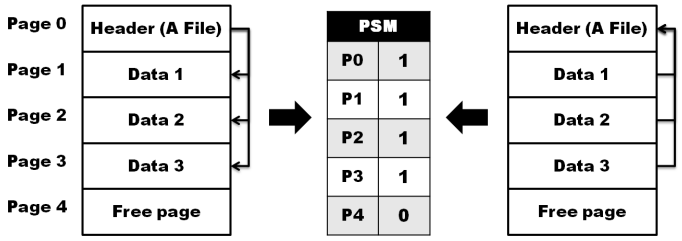
그림 2. 파일 헤더를 통한 데이터 공유

3.2 페이지 공유 맵 생성

페이지 공유 맵(PSM)은 NAND 플래시의 모든 페이지들에 대한 공유 정보를 담는다. 빈 페이지거나 더티 페이지일 경우 0으로 표시되고, 사용되고 있지만 공유되지 않는 페이지는 1로 표시된다. 공유 되고 있는 페이지는 2 이상으로 표시되고, 이 때 공유 값은 그 페이지를 공

유하는 파일의 개수이다. 파일 삭제에 있어서 공유 값이 1인 경우에는 0으로 만들고 그 페이지를 더티로 만든다. 하지만 2 이상인 경우에는 다른 파일에서 공유하고 있기 때문에, 따로 삭제하지 않고 공유 값만 1 감소시킨다.

페이지 공유 맵은 파일 시스템 마운트 시에 만들어진다. NAND 플래시 파일 시스템은 마운트 시에 모든 페이지들을 스캔 하거나, 또는 파일 헤더 정보만 스캔하기도 한다. 스캔하는 과정에서 각 페이지를 공유하는 파일의 개수를 페이지 공유 맵에 갱신 하게 된다. 그림 3을 보면, 파일 A가 있고 이 파일은 데이터 1-3으로 구성되어 있다. 파일 시스템 마운트 시에 각 페이지들을 스캔하면서, 페이지 공유 정보를 페이지 공유 맵에 갱신 한다. 그림 3에서 페이지 공유 맵을 보면 사용되지 않는 페이지 4는 0으로 표시되고 사용 중인 페이지 0-3은 모두 1로 표시된다. 이후에, 다른 파일에서 특정 파일 데이터 페이지를 공유하게 되면, 그 공유되는 페이지는 공유 파일 개수에 따라 공유 값이 2 또는 그 이상이 된다.



(a) 직접 사상 (b) 역 사상

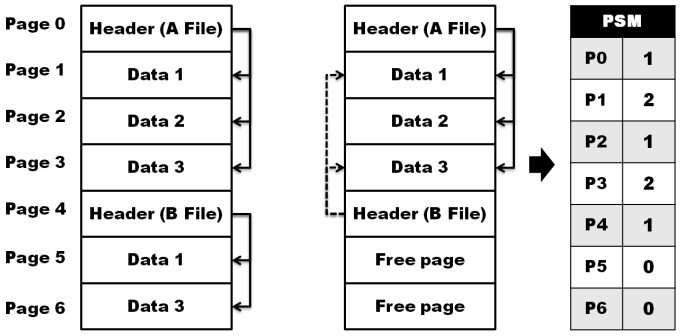
그림 3. 페이지 공유 맵(Page shared map)

3.3 고속 편집저장 기법

본 논문에서 제안하는 기법에서 동영상 파일 편집저장은 먼저 저장 구간과 삭제 구간으로 파일 내용을 나누게 된다. 편집 후 저장 되는 방법은 동일 파일에 덮어 쓰거나 새로운 파일로 저장하는 두 가지가 있다. 덮어 쓰기는 기존 동영상 파일에서 삭제 구간을 지우는 것이고 새로 쓰기는 새 파일에서 편집 파일의 저장 구간을 공유하는 것이다.

그림 4.(a)은 기존 NAND 플래시 파일 시스템의 동영상 파일 편집에서 저장 구간이 데이터 1, 3이고 삭제 구간이 데이터 2이라고 가정 했을 때, 편집 후 새로 쓰기를 한 결과이다. 새 파일 헤더를 생성하고 저장 구간을 빈 공간에 복사 한 뒤, 데이터 위치를 새 파일 헤더로 가리키게 된다. 이처럼 기존 NAND 파일 시스템은 데이터 복사로 인한 시간적, 공간적 소모가 발생하게 된다.

그림 4.(b)와 같이 본 논문에서 제안하는 기법은 저장 구간을 복사 하는 대신 새 파일 헤더로 저장 구간을 가리키면서 원본 파일과 함께 저장 구간을 공유하도록 한다. 먼저 새 파일 헤더를 생성하고 저장 구간에 해당하는 페이지들을 가리키게 한다. 그리고 공유 정보를 페이지 공유 맵에 그림과 같이 갱신한다. 데이터 1, 3은 두 파일에서 공유하게 되어서 공유 값이 2가 된다. 이런 페이지 공유 기법을 통해 기존 NAND 파일 시스템의 데이터 복사 과정이 없어진다. 동시에 데이터 복사에 필요한 시간과 공간 소모도 줄어들게 된다.



(a) 기존 NAND 파일 시스템 (b) 제안 기법

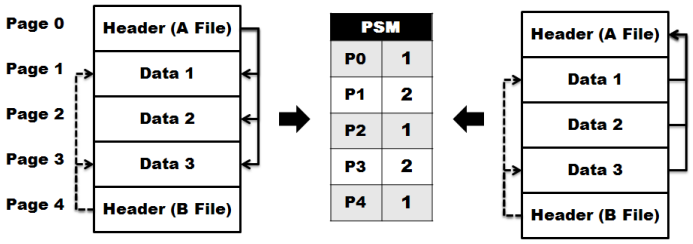
그림 4. 페이지 공유 맵을 통한 데이터 공유

앞에서 제안한 고속 편집저장 기법의 알고리즘을 그림 5에서 보여준다. 편집 후 새로 쓰기는 새 파일 헤더를 생성한 뒤, 저장 구간 페이지들을 가리키면서 공유 한다. 그리고 새 파일 헤더가 위치한 페이지와 저장 구간 페이지들의 공유 값을 1 증가시킨다. 그림 6.(a)에서 편집 후 새로 쓰기의 결과를 한 예로 보여준다. A파일과 B파일은 데이터 1, 3을 함께 공유 하고 공유 정보를 페이지 공유 맵에 갱신하게 된다. 편집 후 덮어 쓰기는 먼저 앞에서 설명했던 편집 후 새로 쓰기와 같은 과정을 수행한다. 그러면 편집된 내용으로 새 파일이 생성된다. 그런 뒤에, 이전 원본 파일을 삭제하면 된다. 원본 파일을 삭제하기 위해 원본 파일 헤더가 위치한 페이지와 삭제 구간 페이지의 공유 값을 1 감소시킨다. 이 때, 공유 값이 0이 되는 페이지는 사용되지 않기 때문에 더티로 만든다. 그림 6.(b)에서 편집 후 덮어 쓰기의 결과를 한 예로 보여준다. 데이터 1, 3을 새 파일 헤더로 가리키게 하고 이전 파일에 해당하는 페이지 중 공유 값이 0인 사용되지 않는 페이지는 더티로 설정한다. 하지만 공유되는 데이터 1, 3은 공유 값이 1이기 때문에 더티로 설정하지 않는다.

```
// n : page number
// S : page numbers of file's saved range
// D : page numbers of file's deleted range
// Hori : original file header
// Hnew : new file header
// PSMn : the number of sharing page n

procedure Fast file editing/storing
  create Hnew
  for each n ∈ S do
    register n to the data position of Hnew
    increase PSMn
  endfor
```

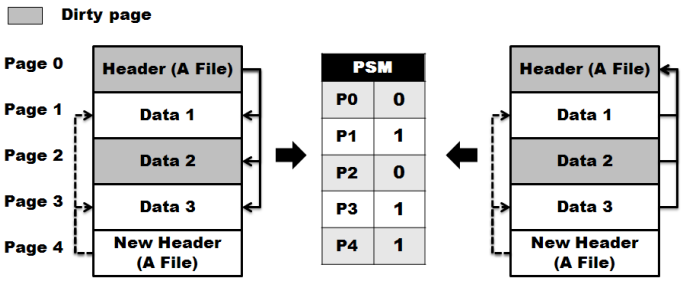
```
if overwrite the the original file
  for each n ∈ D do
    decrease PSMn
    if PSMn = 0
      make page n of dirty page
    endif
  endfor
  n = Hori's page number
  make page n of dirty page
  decrease PSMn
endif

save Hnew
n = Hnew's page number
increase PSMn
end
```

그림 5. 고속 편집 저장 알고리즘



가) 직접 사상 나) 역 사상

(a) 파일 편집 후 새로 쓰기



가) 직접 사상 나) 역 사상

(b) 파일 편집 후 덮어 쓰기

그림 6. 파일 편집 후 페이지 공유 맵 상태

3.4 제안기법 성능 분석

본 논문이 제안하는 기법이 동영상 편집저장에서 보여주는 성능 향상을 분석한다. 먼저 현재 NAND 플래시 파일 시스템으로 널리 쓰이는 YAFFS에서 동영상 편집저장 성능을 실험한다. 실험 환경은 리눅스 운영체제에서 520Hz ARM 프로세서, 64 MB 메인 메모리 그리고 64 MB NAND 플래시 메모리를 사용했다. 16 MB 동영상 파일을 각각 3.2 MB, 8 MB, 12.8 MB에 해당하는 영역을 저장 구간으로 설정한 뒤, 편집저장을 실험하였다. 이에 걸리는 시간은 각각 5.61초, 14초, 22.43초이다. 이를 동영상 파일이 1 GB라고 가정하고 각각 200 MB, 500 MB, 800 MB를 저장 구간으로 편집한다면, 이에 걸리는 시간은 5분 36초, 14분, 22분 26초가 된다.

논문에서 제안하는 기법은 현재 상기 실험환경 상에 구현 중에 있으며, 예측되는 성능은 다음과 같다. 파일 편집저장 과정은 파일 헤더 관리와 파일 데이터 복사 둘로 나뉜다. 그 중 파일 헤더 관리는 파일 입출력이 거의 없고 대부분 프로세서 연산이기 때문에 소모되는 시간이 매우 적다. 반면에 파일 데이터 복사는 대부분 파일 입출력이기 때문에 많은 시간을 소모하게 된다. 본 논문에서는 이 파일 데이터 복사 과정이 파일 공유를 통해 없어졌다. 따라서 파일 편집저장에서 대부분의 시간을 소모하는 파일 데이터 복사 과정이 없어졌기 때문에, 기존 낸드 플래시 파일 시스템에 비해 고속으로 편집이 가능하다. 또한 파일 공유를 통한 중복 데이터도 존재하기 않기 때문에, 소모되는 공간도 최소화 된다.

4. 결론 및 향후 연구 과제

본 논문에서는 NAND 플래시 메모리를 사용하는 멀티미디어 휴대 장치에서 동영상 편집을 고속으로 처리하며 동시에 저장 공간 소모를 최소화 하는 기법을 제안하였다. 제안한 기법은 동영상 파일 편집저장 후 새로운 파일 헤더로 편집 데이터를 가리키고, 공유되는 데이터를 페이지 공유 맵을 통해 관리함으로써 따로 편집 데이터를 새로운 공간에 저장하는 과정을 필요로 하지 않는다. 그로 인해 데이터 복사에 따른 시간 및 저장 공간 소모도 동시에 없어지는 효과를 가져며, 이는 특히 대용량 동영상 파일에 대해 그 성능이 더 우수하다. 앞으로 동영상 파일이 대용량화 되는 추세이기 때문에, 본 논문에서 제안하는 기법은 멀티미디어 장치에서 핵심적인 기술로 응용될 수 있을 것이다.

참고문헌

[1] F. Douglis, R. Caceres, F. Kaasho, K. Li, B. Marsh, and J.A. Tauber, "Storage Alternatives for Mobile Computers," *Proceedings of the 1st USENIX Symposium on Operating System Design and Implementation*, pp. 25-37, 1994.

[2] G. Lawton, "Improved Flash Memory Grows in Popularity," *IEEE Computer*, vol. 39, no.1, pp. 16-18, 2006.

[3] Aleph One, "YAFFS: Yet another flash filing system," http://www.aleph1.co.uk/yaffs/, 2002.

[4] Seung-Ho Lim and Kyu-Ho Park, "An Efficient NAND Flash File System for Flash Memory Storage," *IEEE Transactions on Computers*, Vol. 55, No. 7, pp. 906-912, July 2006.