

## MPEG-2 AAC를 위한 허프만 디코더 구현기법에 관한 연구

정종훈, 김병일, 장태규

중앙대학교 전자전기공학부

## A novel method of Huffman decoding for MPEG-2 AAC

J.H.Jeong, B.I.Kim, T.G.Chang

Department of Electrical and Electronics Engineering

Email : tgchang@jupiter.cie.cau.ac.kr

### 요 약

본 논문에서는 허프만 코딩의 기본 원리인 이진트리로부터 생성된 효율적인 일차원 검색테이블을 구성파, 프로세싱 효율을 증가시키기 위한 수치연산 기법의 도입을 통하여 도출된 효율적인 허프만 디코딩 방법을 제시한다. 제안된 방법에서는 허프만 코드화 된 비트스트림으로부터 읽어 들인 비트 데이터를 직접 이진트리를 검색하기 위한 주소연산에 적용함으로써, 기존의 이진트리 검색을 위하여 수행되던 비교 및 분기문의 제거를 가능토록 하였다.

본 논문에서 제안하는 허프만 디코딩 기법은 기존의 비교 및 분기구분에 의하여 수행되던 허프만 디코딩 기법대비 30%이상의 검색속도 증가 및 이진트리 방식의 검색을 수행하기 위하여 필요로 하는 검색테이블에서 존재하던 데이터 및 노드간의 연결주소를 저장하기 위하여 사용되던 메모리 공간을 효율적으로 제거할 수 있는 일차원 검색테이블을 구성함으로써, 기존 방법대비 1/3의 메모리 공간만을 사용하도록 하였다.

제안된 허프만 디코딩 기법의 성능평가를 위하여 MPEG-2 AAC의 허프만 디코더에 적용한 결과 기존의 순차검색방법에 의한 디코딩 기법대비 198%의 성능 향상을 확인할 수 있었다.

### 1. 서 론

허프만 코드가 가지는 고유한 특징으로 인하여 이진트리를 이용한 허프만 디코딩 기법은 최대 및 평균 검색시간, 또한 검색시간의 편차 측면에서 매우 효율적인 방법으로 간주되고 있다.[1][2] 하지만 이진트리 기반의 검색방법은 검색을 위하여 필요로 하는 자료구조를 생성하기 위하여 연결리스트 기반의 이진트리를 구성하는 복잡한 과정을 수행하여야만 한다.[3][4] 특히, 이진트리 검색에서 노드간의 이동을 위하여 수행하는

비교 및 분기 구문은 프로세서의 동작 중 파이프라인의 흐름을 방해하여 수행속도의 효율을 감소시키는 주요 원인이다.

자료 검색의 방법 중 데이터 hashing, content addressable memory등과 같은 수치연산 기법들은 자료 검색속도의 향상을 위한 잘 알려진 기법들이다. 이진트리 검색에 관한 최근 논문들에서 검색과정 중 수학적인 기법들을 부분적으로 적용함으로써 검색효율을 향상시키기 위한 노력들이 주목된다. [5][6]

본 논문에서는 허프만 디코딩 과정 중 노드간의 이동을 위하여 필요한 조건문의 사용을 완전히 제거하는 방법을 제시하고, 결과적으로 디코딩 과정에 있어 프로세싱 효율을 감소시키는 “비교 및 분기” 명령을 생략하여 효율적인 수행 가능토록 하였다. 노드간의 이동을 위한 다음 노드의 주소는 압축된 비트스트림으로부터 읽어 들인 1bit데이터는 현재 인덱스가 가리키는 위치의 테이블 값과 직접 수치연산을 수행함으로써 획득될 수 있다. 따라서 기존의 이진트리 검색에서 존재하던 조건문의 수행 없이 노드간의 이동을 가능토록 해준다. 결론적으로 본 논문에서 제안하는 코드워드의 수치적 해석방법에 의한 일차원 검색테이블의 생성과 수치연산 기법의 도입은 허프만 디코딩을 위한 효율적인 기법이라 할 수 있겠다. 허프만 트리가 가지는 non-skewed 특성은 위에서 제시한 검색기법의 적용을 가능토록 해준다.

### 2. 코드워드의 수치적 해석에 근거한 허프만 디코딩 기법

본 논문에서 제시하는 허프만 디코딩 기법은 검색테이블의 재구성에 의한 효율적인 일차원 검색테이블의 구성으로부터 기초한다. 그림 1에서는 허프만 디코딩을 위한 검색테이블의 구성과정을 보여준다.

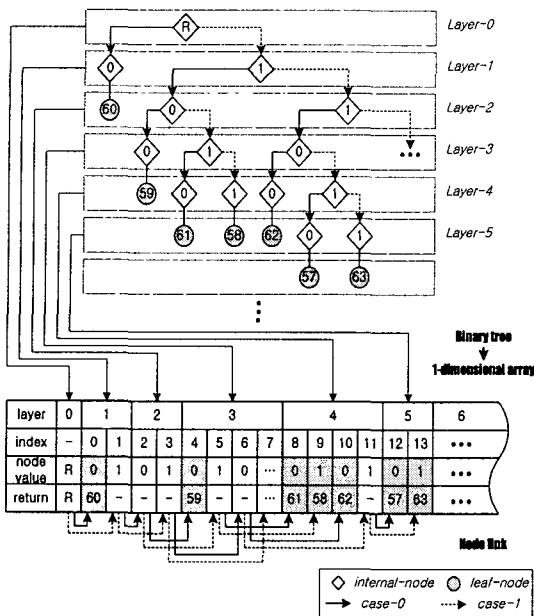


그림 1 허프만 트리로부터 일차원 검색테이블을 생성하는 과정

테이블의 변환과정은 동일 검색단계별로 이진트리의 노드들을 그룹화 하고, 코드워드의 크기 순으로 좌측에서 우측으로 일차원 배열상에 정렬을 실시한다. 이 검색테이블은 연결리스트에 의한 동적인 자료구조의 생성 없이, 직렬화된 정적인 검색테이블의 구성을 가진다. 위의 정렬과정에 의하여 생성된 일차원 배열은 코드워드의 크기 순으로 정렬된다. 이진트리 검색 후 최종적인 결과값을 돌려주는 leaf-node의 경우에는 테이블상에 실제 돌려줄 값을 저장한다.

재구성된 일차원 겹색 테이블에 기초하여 노드 이동을 위한 분기 주소는 식 (1)과 같이 받아들인 이진값을 직접 산술연산에 적용함으로써, 비교구문 없이 직접 주소값을 추출할 수 있게 된다.

*index* := *index* + *DATA(index)* + *new digit()* (1)

그림 1에서 보여진 테이블 재구성 법에 따라 구성된 허프만 코드워드 테이블을 표 1에서 나타내고 있다. 생성된 일차원 검색 테이블의 내부 값은 검색과정의 중간 노드(internal node)의 경우에는 현재 노드와 다음 이동할 노드의 상대적인 거리 값을 저장하고 있다. 결과값을 돌려주는 leaf노드의 경우 되돌려줄 결과값을 지니고 있게 된다. 표상의 "left-0"과 "right-1"로서 표시된 열은 최종적인 검색테이블에는 포함되지 않고 테이블을 생성하는 중간 과정을 나타낸다. 이것은 검색 과정에서 이동할 절대 주소 값을 나타내는 것으로 읽어

들인 1bit 레이터와 연동하여 각각의 경우 이동할 다음 검색단계의 노드 주소를 표시한다.

표 1 헤프만 트리로부터 생성된 일차원 배열의 구성

index	left-0	right-1	offset 0:right-1	return value
0	-	-	-	60
1	2	3	1+(1)	-
2	4	5	2+(1)	-
3	6	7	3+(1)	-
4	-	-	-	59
5	8	9	3+(1)	-
6	10	11	4+(1)	-
7	...	...	...	...
8	-	-	-	61
9	-	-	-	58
10	-	-	-	62
11	12	13	1+(1)	-
12	-	-	-	57
13	-	-	-	63
...	...	...	...	...

일반적인 이진트리 검색방식의 사용 시 각 노드들은 검색의 수행을 위한 테이블은 각 노드에 해당하는 데이터값, 왼쪽 혹은 오른쪽 노드로 분기 시 이동할 주소 값의 세가지 구성요소를 가지고 있어야 한다. 반면에 본 논문에서 제안한 방식을 사용하여 일차원 검색테이블의 생성시 각 노드들은 중간노드(internal node)의 경우 현재노드의 위치기준 다음 이동할 노드의 상대적인 주소값, leaf노드의 경우 되돌려줄 값을 가지게 되고 또한 이 두가지 값은 표 1의 굵은 선으로 표시된 부분에서 보여주는 바와 같이 항상 배타적으로 발생하므로 일차원 공간상에 배열할 수 있게 된다. 따라서 기존의 검색테이블대비 소요 메모리가 1/3로 감소하게 된다.

그림 2에서는 식 (1)을 이용하여 상대주소를 기록한 테이블로부터 허프만 디코딩을 실시하는 과정을 나타내고 있다. 그림 3에서는 전형적인 “비교 및 분기” 명령에 의하여 허프만 디코딩을 수행하는 과정을 나타내고 있다.

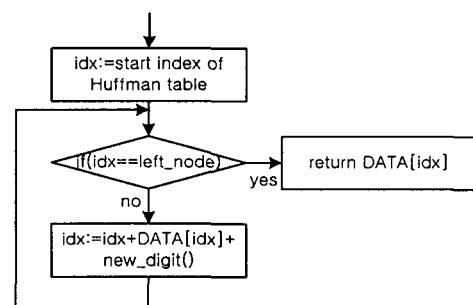


그림 2 수치연산 기법에 기반한 허프만 디코더의 흐름도

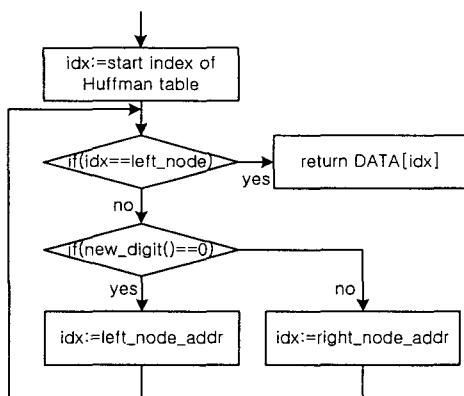


그림 3 “비교 및 분기” 방법에 기반한 허프만 디코더의 흐름도

그림 2와 그림 3에서 나타내고 있는 두가지 허프만 디코딩 방법들 수행과정을 비교하면 기존의 허프만 디코딩 방법에서 노드간의 이동을 위하여 존재하던 조건문을 제거함으로써, 약 30%의 프로세싱 효율을 증가시킬 수 있게 된다. 실제 연산을 수행시 “비교 및 분기” 명령에 의하여 발생하던 프로세싱상의 파이프라인 파괴에 의한 성능저하를 고려한다면 30%이상의 성능향상을 고려할 수 있다.

### 3. 실험결과 : MPEG-2 AAC

본 논점에서는 제안된 허프만 디코딩 방법의 성능 평가를 위하여 제안된 허프만 디코딩 방법을 MPEG-2 AAC를 통해 디코딩 알고리즘을 구현하고, 평균 검색시간을 측정함으로써 성능평가를 수행하였다. 테스트를 위한 입력파일로서 MPEG 웹사이트에서 제공하는 테스트 비트 스트림을 이용하여 측정을 실시하였다. 실질적인 성능향상을 측정하기 위하여, 구현의 용이성으로 인하여 널리 사용되는 순차 검색방식에 기반한 허프만 디코더와 성능을 비교하였다.

MPEG-2 AAC는 각 주파수 대역에 대한 신호의 특성에 따라 인간의 청각 심리모델(psychoaoustic model)을 반영하여 최적의 압축 효율을 얻기 위한 각기 특성이 다른 12가지의 허프만 테이블을 사용한다.

표 2에서는 본 논문에서 제안한 허프만 디코더의 수행결과와, 기존의 순차검색 방식에 의한 허프만 디코더의 수행결과를 비교하여 나열하였다. 실험결과에서 보여주는 바와 같이, 임의의 MPEG-2 AAC 테스트 파일을 시험한 결과 허프만 코드북의 종류에 따라 최소 30%에서 최대 777%의 성능향상을 확인할 수 있었다. 평균 검색횟수 측면에서도 제안된 허프만 디코딩 방법은 일반적인 순차검색방식의 허프만 디코더에 비하여 약75%의 성능향상을 보이고 있다.

표 1 각 허프만 테이블별 평균 검색횟수 측정결과

허프만 코드북	검색단계/ 테이블크기	총 검색 횟수	평균검색횟수		검색속도 향상을 (%)
			제안된 검색방법	순차 검색방법	
scalefactor	19/121	14022	3.91	5.09	30.01
1	16/81	5624	4.51	12.50	177.98
2	16/81	7857	5.39	18.61	245.11
3	16/81	16170	3.77	6.46	71.11
4	16/81	8680	4.77	11.42	139.32
5	16/81	28794	3.08	4.51	46.35
6	16/81	1086	5.17	14.04	172.58
7	16/64	2958	2.64	3.29	24.80
8	16/64	582	4.81	11.13	131.28
9	16/169	234	2.94	5.09	73.26
10	16/169	2594	6.51	33.60	415.63
11	16/289	7640	7.41	65.02	776.43

다음은 동일한 측정 환경을 유지하면서 6가지 서로 다른 MPEG-2 AAC 테스트 파일을 사용하여 측정을 수행하고 그 결과를 표 3에 기술하였다. 실험결과에서 보여주는 바와 같이 파일의 종류에 따라 최소 24%에서 최고 776%의 검색속도 향상을 확인할 수 있었다. 이러한 결과는 기존의 검색방식대비 상당한 성능향상이라 할 수 있겠고, MPEG-2 AAC의 디코딩 과정에서 허프만 디코더가 차지하는 비중을 고려 시 본 논문에서 제안한 허프만 디코딩 방법을 적용한다면 MPEG-2 AAC디코더 성능의 비약적인 향상을 가져올 수 있을 것이라 생각된다.

표 2 여섯 개의 MPEG-2 AAC 테스트 파일의 수행결과 비교

테스트파일	총 검색횟수		검색속도 향상률(%)
	제안된 검색방법	순차 검색방법	
test01.aac	407,591	1,168,714	187
test02.aac	404,608	1,238,404	206
test03.aac	360,727	1,236,965	243
test04.aac	371,423	941,376	153
test05.aac	174,997	511,710	192
test06.aac	239,362	740,700	209
평균	326,451	972,978	198

### 4. 결 론

본 논문에서는 이진트리 검색을 기초로 한 “비교 및 분기” 명령을 통한 허프만 디코딩 방법과 비교하여 프로세싱 효율의 상당한 성능향상의 결과를 나타낸 새로운 허프만 디코딩 방법을 제시하고 있다. 본 논문에서 제안한 허프만 디코딩 방식은 압축된 허프만 비트스

트림으로부터 받아들인 1bit데이터를 직접 검색테이블의 주소값의 연산에 적용함으로써, 기존의 이진트리 검색에서 존재하던 “비교 및 분기” 연산을 제거함으로써 검색효율을 향상시킬 수 있도록 하였다. 본 검색방법의 구현은 허프만 코드워드들을 수치적인 표현으로 해석함으로 검색을 위한 효율적인 일차원 검색 테이블을 구성하도록 하였다. 이론적인 구조의 복잡도 측면에서 약 30%의 효율성 증가를 가져왔고, 검색테이블의 구성을 위한 메모리 공간의 소요역시 기존의 “비교 및 분기” 구문에서 사용하는 테이블대비 1/3의 메모리 소요만으로 구현이 가능하도록 하였다. 성능측정을 위하여 MPEG-2 AAC의 허프만 디코더에 기존의 순차검색에 의한 허프만 디코더와, 본 논문에서 제안한 허프만 디코더를 적용, 실질적인 성능향상 정도를 측정하였다. 6개의 테스트 파일을 이용하여 성능측정을 수행한 결과 평균 기준의 순차검색방식대비 198%의 성능향상을 확인할 수 있었다. 이 결과는 허프만 디코더가 MPEG-2 AAC의 전체 구성요소 중 차지하는 비중을 고려할 때, 이러한 허프만 디코딩의 성능향상은 중요한 결과로서 판단할 수 있다.

#### 참고문헌

- [1] S.HO and P.LAW, Efficient hardware decoding method for modified Huffman code , Electronics letters, 1991, Vol.27, pp. 855-856
- [2] K. L. Chung and J. G. Wu, Level-compressed Huffman decoding , IEEE, 1999, vol. 47, pp. 1455-1457
- [3] W.Hamming, Coding and information theory , prentice-hall, 1986, pp. 51-78
- [4] Khalid sayood, Introduction to data compression, morgan kaufmann, 1996, pp 39-73
- [5] H.D.Lin, David G, High throughput reconstruction of Huffman-coded images , IEEE, 1989, pp. 172-175
- [6] Keshab K. Parhi, High speed Huffman decoder architectures , IEEE, 1991, pp. 64-68
- [7] ISO/IEC JTC1/SC29/WG11 N1650 "IS 13818-7 (MPEG-2 Advanced Audio Coding, AAC)"
- [8] M.bosi and K.Brandenburg, ISO/IEC MPEG-2 Advanced Audio Coding, J. Audio Eng. Soc, 1997, Vol.45, pp 789-814