문자 간 상관관계를 고려한 2 차 한글 허프만 부호 설계 및 최적화 기법

조용래, 윤자천, 박진형, 강신일[1], 설상훈
고려대학교 전자컴퓨터공학과
e-mail : {yrcho, jcyoon, jhpark, sikang, sull}@mpeg.korea.ac.kr,

Second Order Hangul Huffman Code Design and Optimization based on Character Correlation

Yong-Rae Cho, Ja-Cheon Yoon, Jin-Hyung Park, Sinil Kang, Sanghoon Sull
Dept. of Electronics and Computer Engineering, Korea University

요 약

국내 디지털 방송 방식이 결정되고 본격적인 디지털 방송 서비스가 시행되면서, 전자프로그램가이드(EPG: electronic program guide), 주식 및 일기예보등의 문자 방송이 활성화 되고 있다. 특히, 지상파 디지털 방송, 위성 방송, 및 최근 대두되고 있는 DMB (Digital Media Broadcasting) 지상파/위성 방송 등에서 기존의 단순 방송 서비스외에 양방향을 위한 새로운 서비스 개발을 확대하면서 문자방송의 역할은 점점 넓어지고 있다. 본 논문은 한글 데이터의 압축 방법에 관한 것으로, 디지털 방송의 문자 방송 서비스에서 사용되는 문자들을 효율적으로 압축하는 방법을 제안한다. 특히, 현재 서비스 되고 있는 방송의 문자 정보를 분석하고 한글의 특성과 글자간의 상관관계를 고려한 2 차 한글 허프만 부호 설계 기법을 제안한다. 본 논문에서 제안한 방법은 디지털 방송에서 점점 늘어나는 문자 방송의 문자를 효율적으로 압축함으로써 제한된 대역폭을 최대한 활용할 수 있는 방법을 제공한다.

1. 서론

TV 방송이 시작된 이후 디지털 TV(DTV)에 대한 연구가 1990 년대 이후 활발하게 이루어 졌으며, 크게 미국식 시스템인 ATSC (Advanced Television Systems Committee) 방식과 유럽식 시스템인 DVB (Digital Video Broadcasting) 방식으로 나뉘어 개발이 진행되었다. 최근 우리나라에서 디지털 텔레비전 전송 방식을 둘러싼 논쟁이 ATSC 를 채택함으로써 매듭지어졌고 [1], 디지털 방송이 본격적으로 활성화 됨에 따라 관련된 다양한 서비스를 우리나라 실정에 맞게 구현하기 위한 연구가 필요하다.

현재 ATSC 시스템에서는 시청채널 정보 외에 부가적인 정보를 전송하는 프로토콜로써 PSIP(Program and System Information Protocol)를 정의하여 사용한다. 이 표준에서는 영문 텍스트 데이터의 압축을 위하여 2 차로 정의된 허프만 코드를 사용하며, 제목과 내용에 따라 서로 다른 테이블을 사용한다. 여기에서 발생 빈도수가 적은 알파벳은 예외적으로 부호화된다. 그러나 ATSC 전송방식을 채택한 국내 DTV 방송 규격은 한글 코드 압축을 위한 테이블이 마련되어 있지 않아 현재까지 압축을 하지 않은 한글 텍스트를 전송하고 있다. 따라서 데이터 방송이 본격화되는 시점에서 한글 데이터가 차지하는 전송량의 증대는 새로운 문제로 드러나고 있다.

본 논문에서는 2 차 영문 허프만 부호화의 장점을 한글에도 적용할 수 있도록, [2]에서 제시한 한글과 ASCII 코드와의 상관 관계뿐만 아니라, 한글의 특성에 따른 글자간의 상관관계도 고려하도록 한 2 차 한글 허프만 부호화를 제안한다. 본 논문의 구성은 다음과 같다. 2 장은 ATSC 디지털 방송 표준의 PSIP 에서 정의한 다중 문자열에 대한 텍스트 데이터 관련 표준을 알아보고, 3 장에서는 본 논문에서 제안한 2 차 허프만 코드의 생성 과정에 대하여 알아본다. 4 장은 실험 결과를, 5 장에서는 결론에 대하여 기술한다.

[1] 현 ㈜ 팬택 중앙연구소

2. PSIP 에서의 다중 문자열

PSIP 표준에서는 텍스트 문자열을 나타내기 위해 표 1 과 같이 다중 문자열 구조(multiple string structure)를 사용한다[3].

표 1. 다중 스트링 구조

Syntax	No.ofBits	Format
multiple_string_structure()		
number_strings	8	uimsbf
for(i=0;i<number_strings;i++){		
ISO_639_language_code	24	uimsbf
number_segments	8	uimsbf
for(j=0;j<number_segments;j++){		
compression_type	8	uimsbf
mode	8	uimsbf
number_bytes	8	uimsbf
for(k=0;k<number_bytes;k++)		
compressed_string_byte[k]	8	bslbf
}		
}		
}		

표 1 에서 보듯이 압축된 문자열을 표현하기 위하여 압축 방법을 나타내는 *compression_type* 이 존재한다. 이 값은 다음 표 2 에서 설명되어 있듯이 압축 방식을 나타낸다. 영어의 경우 0x01 및 0x02 로써 두 가지의 부호/복호 방식을 제공하고 있으며, 다양한 허프만 테이블을 지원하기 위하여 0x03 에서 0xAF 까지 그 값을 비워놓았다. 한글의 경우 이 값 중 하나를 선택하여 한글 부호/복호를 위한 해당 허프만 테이블을 쓰게 할 수 있다.

표 2. 압축 방법

Compression_type	Compression Method
0x00	No compression
0x01	Huffman coding using standard encode/decode tables defined in Table C.4 and C.5 in AnnexC
0x02	Huffman coding using standard encode/decode tables defined in Table C.6 and C.7 in AnnexC
0x03 to 0xAF	Reserved
0xB0 to 0xFF	Used in other systems

표 3 은 압축에 사용한 텍스트 모드에 대한 설명으로 어떤 언어인지 구분할 수 있게 한다. PSIP 에서는 표준완성형과 유니코드에 대한 텍스트 모드를 할당해 두어 활용할 수 있게 하였다.

표 3. 텍스트 모드

Mode	Meaing
0x00-0x3E	ISO/IEC 10646-1 or reserved
0x3F	Select Unicode, UFT-16 Form
0x40-0x41	Assigned to ATSC standard for Taiwan
0x42-0x47	Reserved for future ATSC use
0x48	Assigned to ATSC standard for South Korea
0x49-0xDF	Reserved for future ATSC use
0xE0-0xFE	Used in other systems
0xFF	Not applicable

3. 2 차 허프만 코드 생성 과정

PSIP 표준에서 정의한 것과 같이 본 논문에서도 각 글자의 출현 빈도에 따른 허프만 부호화[4]를 제안하며, 한글의 경우 영어에서와 같이 글자와 글자 간에 적용되는 특성이 있음은 물론 띄어쓰기나 마침표 등에 의한 ASCII 코드와의 상관관계도 생각해야 한다.

3-1. 허프만 부호화와 완성형 한글

본 논문에서 대상으로 하는 한글의 표현 코드는 표준 조합형(KSX1001 : KSC 5601-1982), 표준 완성형(KSX 1001 : KSC5601-1987), 유니코드(KSX 1005-1 : ISO/IEC 10646-1) 세 가지 중 표준 완성형을 대상으로 하였다. 완성형 한글코드는 1987 년 정부가 한국표준으로 정한 것으로 가장 많이 사용되는 한글 음절마다 2 바이트의 2 진수를 1 대 1 로 대응하여 표현하는 방법이다. 이 코드는 한글을 나타내는 코드의 첫 번째와 두 번째 바이트의 MSB 가 모두 "1" 로 만들어져 있다. 따라서 아스키 코드의 영문자는 7 비트 코드이기 때문에 MSB 를 "0"으로 만들면, 한글과 영문 아스키 코드가 중복되지 않아 처리하기에 편리하다는 장점이 있다. 조합형은 자음과 모음으로 조합 가능한 모든 한글을 사용할 수 있으며, 심지어 우리나라 고어까지 취급할 수 있는 장점이 있으나, 출력시 다시 모아써야 하는 불편이 있다는 것이 단점있기 때문에 많은 논란 끝에 완성형 한글이 한국표준으로 제정되었다. 표준 완성형 한글코드로 표현할수 있는 한글은 2350 자이며, 각 음절당 2 바이트를 할당하여 사용한다.

그러나 한글 2350 자 및 ASCII 문자로 표현 가능한 문자 128 개 문자 모두에 대하여 모든 1 차 및 2 차 테이블을 구성할 경우 부호화/복호화를 위한 테이블의 양은 기하급수적으로 늘어나게 된다[5]. 따라서 본 논문에서는 1 차적으로 특정 확률 이상의 출현 빈도(k_1)를 가지는 문자들을 대상으로 한글 및 ASCII 문자의 1 차 테이블을 작성하였다. 또한, 앞선 과정에서 1 차 문자로 결정된 한글 문자에 대해서만 그에 이어지는 2 차 문자의 검색을 수행하며, 여기서 특정 확률 이상의 출현 빈도(k_2)를 가지는 한글 및 ASCII 문자에 대해 2 차 테이블을 작성하였다. 그림 1 은 간략화된 2 차 트리이며, 그림 2 는 2 차 허프만 테이블 생성 흐름도 이다.

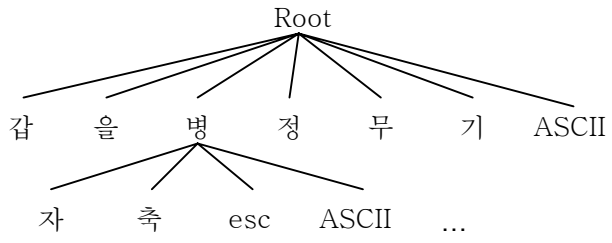

그림 1. 간략화 된 2 차 트리

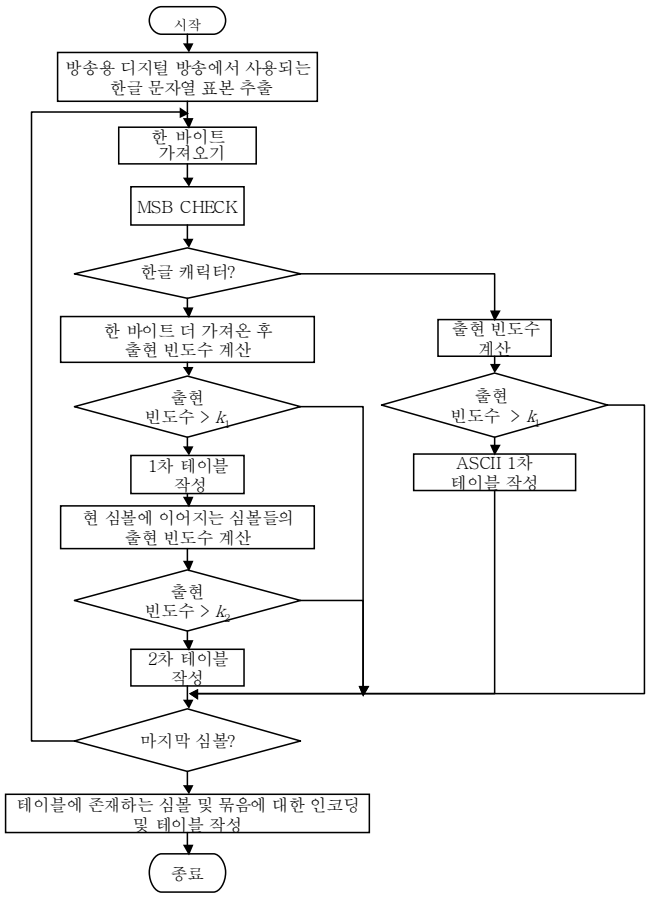

그림 2. 제안한 2 차 허프만 테이블 생성 흐름도

이 테이블을 근거로 허프만 부호화를 수행하며 출현 빈도가 적은 문자에 대해서는 PSIP 에서 수행하는 것과 같이 예외 처리를 하는 기법을 이용하여 테이블의 크기 및 수행시간 단축을 꾀하였다.

3-2. 한글 문법상의 특징

한글은 영어의 문법에서 단어마다 띄어쓰기를 하는 것처럼, 어절과 어절 사이에 띄어쓰기를 한다. 그러나 영어에서는 띄어쓰기가 모든 품사의 뒤에서 골고루 발하는데 반해, 한글에서는 주로 조사 이후에 띄어쓰기가 많이 발생하는 특징을 가진다. 따라서 이러한 특징에서 발생하는 한글과 ASCII 코드 사이의 상관관계를 이용하여 2 차 허프만 부호화를 수행할 수 있으며, 이는 [2]에서 제시한 방법과 동일하게 수행된다.

본 논문에서는 한글과 띄어쓰기와 같은 ASCII 코드와의 상관관계 외에, PSIP 에서 영어가 각 글자에 대한 상관관계를 이용하여 2 차 허프만 부호화를 수행하는 것과 같이 한글과 한글 사이의 상관관계를 이용하여 2 차 허프만 코딩을 수행할 것을 제안한다.

한글은 영어와 달리 가능한 문자의 수가 2350 개에 달하기 때문에 각 글자와 글자와의 상관관계가 크지 않다. 따라서 영어에서 2 차 허프만 부호화를 수행하는 것과 같은 방법으로 단순히 전 글자와의 상관관계를 이용해 부호화를 수행할 경우 허프만 부호화가 비효율적일 수가 있다. 그러므로 본 논문에서는 출현 비율을 정함에 있어 두 글자를 묶어 허프만 부호화를 수행하였다. 예를 들면 “하”라는 문자의 출현빈도를 살펴보면 샘플 문서 내 총 12775 회 출현하였으며, 이후 이어지는 문자의 출현 비중은 “는”이 27.45%, “고”가 17.80%, ”지”가 8.10% 와 같다. “하는”,”하고”,”하지” 에 대해 허프만 부호를 할당하고 이후 등장하는 글자부터는 다시 새로운 부호화를 실행하는 변형된 2 차 부호화 방법을 사용하였다.

3-3. 허프만 부호화의 최적화

앞선 논문에서는 한글과 ASCII 단위로만 허프만 부호화를 수행하기 때문에, 문장 내에서 문자 순서대로 허프만 부호화를 수행하면 언제나 최단의 허프만 부호 길이를 얻어낼 수 있다. 그러나 본 논문에서 제안한 알고리듬의 경우 순서를 고려하여 않을 경우 최적의 결과보다 긴 허프만 부호 길이를 얻는 경우가 생길 수 있다. 예를 들어 “저마다의_” 라는 문장을 허프만 부호화 할 경우를 생각해 보면, 모든 문자에 대해 2 차 테이블이 존재하다고 가정할 경우 “저마”, “다의”, “_” 로 허프만 부호화를 수행할 수 있을 것이다. 하지만 오히려 “저”, “마다”, “의_”로 허프만 부호화를 수행하는 경우가 각 단위의 출현 확률에 따라 허프만 부호 길이가 짧은 경우가 발생할 수 있다. 따라서 본 논문에서는 어절마다 허프만 부호화를 수행하되, 가능한 모든 순서를 고려하여 허프만 부호화를 수행하였다. 예를 들어, “저마다의_” 라는 문자열에 대한 한글 허프만 부호화의 과정은 다음과 같다. 우선 실험 데이터에서 각 문자의 빈도수는 다음과 같다.

표 4. 실험데이터에서 각 문자의 빈도수

1 차 심볼	횟수	이어지는 2 차심볼	횟수
“저”	7,793 회	“마”	65 회
“마”	39,074 회	“다”	2,001 회
“다”	215,121 회	“의”	271 회
“의”	229,812 회	“ ” (공백)	205,411 회
“ ” (공백)	539,812 회		

이를 이용하여 일반적인 방법으로 “저마다의_“라는 문자열을 허프만 코딩 했을 경우 생성되는 총 비트 스트림의 길이는 아래와 같다.

저마 - 1000 1010 1000 1111 1 (17 bits)
다의 - 1001 0110 0101 011 (15 bits)
Space - 0111 (4 bits)

그러나, 최적의 인코딩을 고려해보면 결과는 달라진다. 즉, “저”가 1 차 심볼 테이블에 존재하고 이어지는 “마” 역시 “저”의 2 차 심볼 테이블에 존재하지만 최적의 인코딩을 위해 “저”에 대해 “저” 한 글자에 대한 허프만 코드를 할당한 뒤 남은 “마다의_” 에서 “마다”를 묶어 인코딩을 하고 “의_”를 묶어 인코딩을 하면 결과는 아래와 같다.

저 - 1111 1001 0011 101 (15 bits)
마다 - 0111 0110 0000 0 (13 bits)
의_ - 0000 10 (6 bits)

따라서, 본 논문에서는 상기의 과정과 같이 두 글자 단위로 많이 이루어져 있는 한글 문자의 특성을 활용하기 위하여 단어 단위의 최적의 압축방법을 취하였다.

4. 실험결과

본 논문에서 사용된 한글 문장의 표본은 KBC, MBC, SBS 등 방송 3 사에서 2003 년 6 월부터 2005 년 2 월에 거쳐 방송된 뉴스, 드라마 줄거리, 프로그램 정보 등을 수집하여 사용하였다. 앞서 언급하였듯이 표준 조합형 한글은 국내 지상파 DTV 규격에서 제외되어 있고, 유니코드에서 표준 완성형 한글 이외의 문자는 발생 확률이 매우 낮기 때문에, 본 논문에서는 표준 완성형 한글만을 실험 대상으로 하였다. 실험에 사용된 데이터는 표준완성형 한글 1,981,237 문자, 1,068,230 ASCII 문자를 대상으로 하였으며, 이때 심볼의 수는 한글 2,207 개 및 ASCII 문자 95 개였다.

실험 중 테이블에 존재하지 않는 문자에 대해서는 PSIP 에서와 같이 예외 처리 부호화를 실행하였으며, 출현 확률 기준값을 정하여 기준값 보다 큰 출현 확률을 가지는 글자에 대해서 테이블을 작성하였다. 따라서 출현 확률 기준 값에 따라 허프만 부호화를 수행하게 되는 한글 및 그에 2 차 심볼로 추가되는 ASCII 문자 및 한글 문자의 수는 달라질 것이다. 그에 따른 압축률 및 디코딩 테이블의 크기는 표로 정리하였다.

표 5 에서 보면 알 수 있듯이 테이블의 크기는 출현 빈도수 임계치(k_1, k_2)에 의해 결정되며 주로 2 차 심볼의 개수에 영향을 준다. 표 5 의 마지막 열에서와 같이 실험 데이터 내 출현한 총 1,827 자의 한글과 95 자의 ASCII 모두에 대해 허프만 부호화를 실행할 경우 최대 54.92%의 압축률을 보인다. 전체적으로 제안된 기법은 기존 일차 테이블을 사용하는 경우보다 약 13~17% 정도의 압축율의 증가를 가져왔다.

표 5. 실험 결과

	1 차 한글 부호화					
		ASCII 문자를 고려한 2 차 한글 부호화 [4]				
			제안된 2 차 한글 부호화 (k_1,k_2)			
			(37,37)	(20,20)	(20,10)	(1,1)
1 차심볼수	891	891	1,019	1,150	1,150	1,827
2 차심볼수	-	5,951	7,071	11,230	17,425	47,080
ASCII 심볼수	95	95	95	95	95	95
압축률(%)	46.37	50.73	53.11	53.72	54.20	54.92
디코딩 테이블 크기(KBytes)	3.8	15.8	41.8	63.3	94.3	246.6

5. 결론

테이블 용량은 2 차 한글 심볼까지 생각한 관계로 약 2 배에서 10 배정도 증가하였으나, 최근 출시되고 있는 Set-Top Box 의 저장공간의 크기를 생각할 경우 큰 문제가 되지 않는다. 또한 테이블에 변화가 생길 경우에만 테이블을 재전송 해주면 되고, 그런 경우는 자주 있는 것이 아니기 때문에 디코딩 테이블 용량의 증가는 크게 문제가 되지 않을 것이다. 그러나 데이터 방송에서 한글 데이터가 차지하는 비중이 계속 증가될 것으로 예상되므로 적은 압축률 향상으로도 전송 효율 및 그로 인한 기타 정보의 전송 가능성을 생각하면 큰 효과를 볼 수 있다.

또한 기존의 한글과 ASCII 문자만을 고려한 2 차 허프만 부호화 기법에서는 한글의 특성중 조사 뒤에 띄어쓰기, 마침표 등과 같은 ASCII 코드와의 관계를 이용하여 높은 압축효율을 보였지만, 이와 같은 특징이 없는 일반적인 2 바이트의 문자에서는 압축효율이 떨어질 수 밖에 없다. 하지만 본 논문에서와 같이 일반적인 관점에서 2 바이트 심볼과 1 바이트 심볼을 동시에 고려할 경우 응용 분야는 더욱 넓어질 것이다.

참고문헌

[1] 정보 통신 단체 표준, 지상파 디지털 TV 방송 송수신 정합표준, 2000 년 12 월 20 일

[2] 강신일, 윤자천, 설상훈 “디지털 방송에서 문자 전송을 위한 2 차 한글 허프만 부호 설계”, 통신학회 추계종합학술발표회 논문지, pp.213 , Nov. 2004

[3] ATSC standard A/65B, Program and System Information Protocol for Terrestrial Broadcast and Cable, Mar.2003

[4] David A. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” Proceeding of the institute of Radio Engineers 40, Sept. 1952

[5] 황재정, 진경식, “디지털 방송용 한글 허프만 부호 설계 및 PSIP 구조”, 한국방송공학회논문지, vol. 6, no.1, pp 98-107, Jun.2001

.