

Huffman 부호에 기초한 비대칭적 양방향 가변길이 부호기 설계 방법

정육현*, 이호경**, 호요성*

*광주과학기술원 정보통신공학과, **홍익대학교 전자전기공학부

Design Algorithm for Asymmetrical Reversible Variable-Length Codes from the Huffman Code

Wook-Hyun Jeong*, Ho-Kyoung Lee**, Yo-Sung Ho*

*Kwangju Institute of Science and Technology (K-JIST)

**Dept. Electronics and Electric Engineering, Hongik University

{whjeong, hoyo}@kjist.ac.kr, hkleee@hongik.ac.kr

Abstract: 양방향 가변길이 부호는 손상된 비트열에 대해 탁월한 복구 능력을 가지므로 오류 내성 부호화 기법의 하나로 사용되고 있다. 양방향 가변길이 부호는 비대칭적 양방향 가변길이 부호와 대칭적 양방향 가변길이 부호로 나눌 수 있으며, 일반적으로 비대칭적 양방향 가변길이 부호는 두 개의 부호 테이블을 필요로 하지만, 대칭적 양방향 가변길이 부호보다 더 나은 효율을 제공한다. 이러한 비대칭적 양방향 가변길이 부호를 설계하기 위해 Huffman 부호를 기반으로 생성하는 방법들이 제안되었지만, 기존의 방법들은 복잡도 측면과 효율성 측면에서 개선의 여지가 많다. 본 논문에서는 최적화된 Huffman 부호 테이블을 이용하여 비대칭적 양방향 가변길이 부호를 설계하는 새로운 알고리즘을 제안한다. 제안된 알고리즘은 기존의 알고리즘보다 부호의 생성과정이 간단하며 평균부호길이가 줄어든다.

Keywords: Error resilience, Huffman Code, H.263, MPEG-4, Reversible variable-length codes(RVLCs)

1. 서론

Huffman 부호[1]나 산술 부호[2]와 같은 가변길이 부호는 정보원(source)의 통계적인 특성을 이용하여 압축 효율을 높이지만, 잡음이 심한 전송 환경의 비트 오류에 대하여 매우 취약한 단점이 있다. 따라서 가변길이 부호로 부호화 된 비트열(bit stream)에 단일 비트 전송 오류가 발생하더라도 바로 다음에 위치한 동기점(resynchronization marker)까지 남은 비트열을 손상시켜 두 동기점사이의 모든 데이터를 버리게 되는 심각한 문제를 발생시킨다.

최근 압축된 비트열에 대한 전송 오류의 영향을 줄이기 위한 오류 내성 부호화 기법의 하나로써 양방향 가변길이 부호(reversible variable-length code, RVLC)가 제안되었다. RVLC는 전송 오류가 발생했을 때 비트열을 순방향이나 역방향으로 복호하여, 수신된 비트열로부터 손상되지 않은 멀티미디어 데이터를 가능한 한 많이 복원할 수 있으며, 손상된 영역을 줄여 복호기에서 사용되는 오류 은닉 기법을 효과적으로 지원할 수 있다[3].

RVLC는 부호어(codeword) 구성의 비트 패턴(pattern)에 따라서 대칭적(symmetrical) RVLC와 비대칭적

(asymmetrical) RVLC로 나눌 수 있다. 대칭적 RVLC는 비대칭적 RVLC보다 평균부호길이가 길지만, 하나의 부호어 테이블을 요구한다는 장점을 가진다. 반면에 비대칭적 RVLC는 부호 할당이 유연하여 Huffman 부호와 평균부호길이가 근접하고, 대칭적 RVLC보다 높은 부호화 효율을 가지지만 순방향과 역방향의 복호를 위해 두 개의 부호어 테이블을 사용해야 한다. 대칭적 RVLC는 H.263+에 채택되었고, 비대칭적 RVLC는 MPEG-4 표준에 채택되어 사용되고 있다[3, 4].

Fraenkel[5]은 주어진 부호 비트 길이 집합을 기반으로 완전한 RVLC를 생성하는 알고리즘을 발표하였다. Wen[6]은 Golomb-Rice 부호와 exponential Golomb 부호와 동일한 비트 길이를 갖는 RVLC에 대한 설계 기법을 제안하였다. Takishima[7]은 주어진 Huffman 부호에 기초하여 평균부호길이를 줄일 수 있는 RVLC의 생성 방법을 처음으로 제안하였고, Tsai[8]는 이 알고리즘을 수정하여 보다 짧은 평균부호길이를 갖는 RVLC를 생성하는 방법을 제안하였다. 그런데, Huffman 부호를 기반으로 RVLC를 설계하는 이 두 가지 알고리즘의 분석 결과는 몇 가지 개선의 여지를 보여준다. 이 알고리즘들은 전체 이진나무(binary tree)에서 부호를 검색하고, 부호 생성 이전에 각 레벨에서 사용 가능한 RVLC의 수를 미리 계산하며, Huffman 부호의 각 레벨에 이 값을 적용시킬 때에 발생하는 제한 사항으로 사용 가능한 RVLC를 선택하지 못하는 경우가 종종 발생하게 된다.

그래서 전체 이진 나무의 반쪽 영역에서 부호를 검색하고 주어진 Huffman 부호에 대한 적응 과정을 단순화 시켜 복잡도를 줄이는 동시에 부호화 효율을 높여 이러한 문제점들을 극복할 수 있는 대칭적 RVLC를 설계하는 방법이 제안되었다[9].

본 논문에서는 최적화 되어 있는 Huffman 부호를 기반으로 비대칭적 RVLC를 설계하는 새로운 기법을 제안한다. 제안된 기법 역시 전체 이진나무의 한쪽 영역에서 부호를 검색하고, Huffman 부호와 심볼의 확률 분포에 대한 적응과정을 단순화 시켜 복잡도를 줄인다. 그리고 적응과정에서 발생할 수 있는 제한 사항을 해결하여 RVLC의 효율성을 향상 시키고 가장 긴 부호어의 비트 길이를 줄여 한정된 시스템에서도 RVLC가 적용될 수 있도록 한다.

II. 기존의 RVLC 생성 방법

Tsai가 제안한 방법은 Takishima의 RVLC 중에서 효율성을 향상시킨 특별한 경우에 해당한다. Tsai의 RVLC는 주어진 최적의 허프만 부호에서 출발하여 그 부호들을 상위 레벨에서부터 비대칭적 RVLC로 바꾸어 가며 생성된다. 주어진 가변길이 부호 혹은 설계된 RVLC의 비트 길이 벡터는 $(n(1), n(2), \dots, n(L_{max}))$ 로 구성되는데 여기서 $n(i)$ 는 비트 길이 i 를 갖는 부호어 수를 나타내고, L_{max} 는 이 부호 안에서 최대 부호어 길이를 나타낸다. Huffman 부호의 비트 길이 벡터는 $n_{Huff}(i)$ 이고 RVLC의 비트 길이 벡터는 $n_{RVLC}(i)$ 이다. 부호화 되어야 할 전체 심볼(symbol)수를 S 로 놓으면, Tsai의 방법은 먼저 각 레벨의 가능한 비대칭적 부호어 수, $avail(i)$ 를 미리 계산한 후에 비대칭적 RVLC를 설계한다.

평균부호길이를 줄이기 위해서 Tsai는 각 레벨에서 더 많은 비대칭적인 부호어를 보장할 수 있는 부호어들에게 우선 순위를 두고 선택하는 방식을 사용하였다. 이와 같이 우선 순위에 따라 부호어들이 선택된다면 하위 레벨로 내려갈수록 많은 부호어들을 보장하여 설계된 RVLC의 효율성이 향상된다.

각 부호어들의 우선 순위는 최소반복간격(minimum repetition gap, MRG)[8]을 기반으로 하여 정해진다. m 비트로 구성된 부호어 C 와 이진 심볼 0과 1로만 구성된 비트 패턴 X 가 존재하면 C 뒤에 k 비트 길이의 X 를 연결시켜 새로운 부호 C_{new} 를 만들 수 있다. 여기에서 C_{new} 의 비트 길이는 $m+k$ 가 된다. C_{new} 의 어미 부분의 m 비트 패턴 $Suff_m$ 은 X 에 따라서 C 와 같은 형태가 나올 수 있다. MRG는 $Suff_m$ 과 C 를 같도록 하는 최소한의 k 값을 말한다. 그리고, MRG 값이 작을수록 하위 레벨에 보다 많은 $avail(i)$ 를 보장한다. 표 1은 3비트 길이를 갖는 부호어들의 MRG 값과 더불어 하위 레벨에 존재하는 가능한 RVLC 수를 보여주고 있다.

표 1. 레벨 3에 존재하는 부호어에 대한 MRG 값과 하위 레벨에 존재하는 $avail(i)$

부호어	MRG	$avail(i)$	
		레벨 4	레벨 5
000	1	13	24
111	1	13	24
010	2	12	21
101	2	12	21
001	3	12	20
110	3	12	20
011	3	12	20
100	3	12	20

Tsai의 비대칭적 RVLC를 생성하는 알고리즘을 정리하면 다음과 같다.

(과정 1) 비대칭적 RVLC의 비트 길이 벡터를 초기화한다. 그리고 각 레벨에서 MRG를 기반으로 $avail(i)$ 를 계산한다.

$$n_{RVLC}(i) = n_{Huff}(i) \quad (\forall i \geq 1) \quad (1)$$

(과정 2) $n_{RVLC}(i) \leq avail(i)$ 이면 $n_{RVLC}(i)$ 를 그대로 유지하고 그렇지 않으면 다음과 같이 한 비트가 추가된다. 결정된 $n_{RVLC}(i)$ 에 따라 부호어를 검색한다.

$$n_{RVLC}(i+1) = n_{RVLC}(i+1) + n_{RVLC}(i) - avail(i), \quad (2)$$

$$n_{RVLC}(i) = avail(i) \quad (3)$$

(과정 3) S 에 대한 최종 비대칭적인 RVLC가 할당될 때까지 과정 2)를 반복한다.

그런데, (과정 2)와 같이 주어진 Huffman 부호에 대한 적응과정이 모든 레벨에 걸쳐 진행되기 때문에 몇몇 레벨에서 RVLC의 수가 Huffman 부호의 수보다 클 수 없는 제한을 받게 된다. 과정 2)에서 $\sum_i n_{Huff}(i) = \sum_i m(i)$ 이

고 $n_{RVLC}(i+1) \leq avail(i+1)$ 이면 $n_{RVLC}(i)$ 는 그대로 유지되어 이 문제점이 발생하게 된다. 더욱이 효율성에 큰 영향을 미치는 최상위 레벨에서 $\sum_i n_{Huff}(i) = \sum_i m(i) = 0$ 이

로 가장 높은 발생확률을 갖는 심볼에 충분히 짧은 부호어를 할당할 수 없게 된다. 따라서 최종 RVLC의 효율성을 저하시킨다.

III. 제안된 RVLC 알고리즘

1. 비트 반전 기반의 비대칭적 RVLC의 검색

그림 1 (a)와 그림 1 (b)는 Huffman 부호와 그에 따른 이진나무의 한 예를 보여준다. 최상단의 절점에서 각각 오른쪽과 왼쪽으로 향하는 가지(branch)에 0과 1을 할당하고, 오직 단점(leaf)만이 부호어로 선택된다.

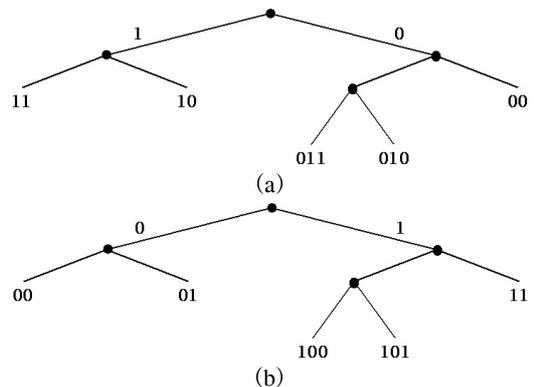


그림 1. Huffman 부호와 이진나무

만약 대칭적 RVLC만 고려한다면 부호어의 대칭적인 특징 때문에 그림 1 (a)와 그림 1 (b)에서 00과 11, 010과 101처럼 비트 반전을 통해 같은 부호 길이를 가지는 또 하나의 대칭적 RVLC를 얻게 된다. 하지만 이러한 특성은 대칭적 RVLC에게만 국한되지 않는다. 비트 반전을 이용한 한쪽 영역 검색[9]은 비대칭적 RVLC에도 적용될 수 있다. 이를 이용하여 제안된 알고리즘은 전체 이진나무에서 부호어를 검색하지 않고 특별히 0으로 할당된 한쪽 영역에서 부호어를 검색한다. 대칭적인 부호어는 비대칭적인 부호어의 특별한 경우이므로 대칭적인 부호어와 비대칭적인 부호어를 모두 검색한다. 그리고 이 영역에서는 S 만큼이 아니라 $\lceil S/2 \rceil$ 만큼 부호어를 검색한다. 여기에서 $\lceil x \rceil$ 는 x 와 같거나 큰 수 중 가장 작은 정수를 말한다. 최종적인 비대칭적 RVLC는 0으로 할당된 한쪽 영역에서 선택된 부호어들과 그 부호어들을 비트 반전시킨 부호어들로 구성된다.

처음에 0으로 할당된 한쪽 영역에서 선택된 비대칭적인 부호어들은 제거과정을 거치며 어두 조건과 어미 조건을 만족시키게 된다. 이 영역은 Huffman 이진나무의 첫 번째 sub-tree와 같으므로 식 (4)의 Kraft 부등식은 식 (5)와 같이 개정된다.

$$\sum_{i=1}^S 2^{-l_i} \leq 1, (l_1 \leq l_2 \leq \dots \leq l_S) \quad (4)$$

$$\sum_{i=1}^{\lceil S/2 \rceil} 2^{-l_i} \leq \frac{1}{2}, (l_1 \leq l_2 \leq \dots \leq l_{\lceil S/2 \rceil}) \quad (5)$$

부호어들에 비트 반전 연산이 취해지면 어두와 어미가 비트 반전되므로 어두 조건과 어미 조건을 동시에 만족하게 된다. 따라서 1로 할당된 영역 내에서 역시 유일하게 복호가 가능하게 된다. 그리고, 비트 반전 연산을 통해 얻어진 부호어들의 비트 길이는 당연히 비트 반전 이전의 부호어들의 길이와 동일하기 때문에 개정된 Kraft 부등식은 식 (6)과 식 (7)로 확장된다.

$$\sum_{i=1}^{\lceil S/2 \rceil} (2^{-l_i} + 2^{-l_i}) = \sum_{i=1}^{\lceil S/2 \rceil} (2 \times 2^{-l_i}), (l_1 \leq l_2 \leq \dots \leq l_{\lceil S/2 \rceil}) \quad (6)$$

$$2 \times \sum_{i=1}^{\lceil S/2 \rceil} 2^{-l_i} \leq 2 \times \frac{1}{2}, (l_1 \leq l_2 \leq \dots \leq l_{\lceil S/2 \rceil}) \quad (7)$$

부호화 된 비트 길이 l_i 는 같거나 다를 수 있으므로 결국 식 (7)로부터 다음과 같이 완전한 Kraft 부등식을 얻게 된다.

$$\therefore \sum_{i=1}^S 2^{-l_i} \leq 1, (l_1 \leq l_2 \leq \dots \leq l_S) \quad (8)$$

따라서 식 (4) ~ 식 (8)을 통해, 제안된 비트 반전을 이용한 부호 검색 방법이 즉각적으로 복호가 가능한 RVLC를 제공하는 사실을 알 수 있다.

2. H_L 적응법

대칭적 RVLC의 경우를 보면 효율성은 최상위 레벨에 어떠한 부호어들을 선택되느냐에 의존한다. 그래서 대칭적 RVLC의 효율성을 높이며 동시에 적응 과정을 단순화 시킨 Z_L 적응법이 제안되었다[9]. Z_L 적응법은 주어진 Huffman 부호에서 최상위 레벨, L_{min} 의 비트 길이를 갖는 모두 0으로만 이루어진 부호어를 강제로 선택하고 나머지 부호어를 차례대로 선택하는 방법이다. 이 방법은 각 레벨에서 선택 가능한 부호어를 놓치는 일이 없고, 최상위 레벨에 보다 많은 대칭적 부호어를 할당하여 평균부호길이를 줄이는 데에 크게 기여를 하는 방식이다.

하지만 비대칭적 RVLC에서는 부호어 구성 방식에 상관하지 않기 때문에 부호어 모양이 비대칭적이던 대칭적이던 구분을 하지 않는다. 달리 말하면, 대칭적 RVLC에서는 부호의 효율성이 어떠한 대칭적 부호어가 선택되는가에 기반하였다면 비대칭적 RVLC에서는 최상위 레벨에 얼마나 많은 부호어들이 선택되는가에 기반하고 있다. 물론 비대칭적 RVLC에서도 특정 부호어의 선택 여부에 효율성이 달라질 수 있지만 우선적으로 고려해야 할 사항은 바로 부호어의 수이다.

본 논문에서는 Huffman 부호로부터 비대칭적 RVLC를 설계하기 위해 H_L 적응법을 제안한다. 적응법은 주어진 Huffman 부호의 최상위 레벨 L_{min} 에 존재하는 전체 부호어 수의 반, 즉 $2^{(L_{min}-1)}$ 만큼 부호어를 선택하고 나머지 비대칭적 RVLC들을 하위레벨부터 차례차례 선택하는 방식이다. H_L 은 $2^{(L_{min}-1)}$ 을 의미한다. 제안된 알고리즘에서는 비트 반전을 이용한 반 영역 검색이 사용되므로 처음에 L_{min} 에서 선택되는 부호어의 수는 실제로 $2^{(L_{min}-2)}$ 이다.

그런데, 효율성은 L_{min} 에서 선택되는 부호어 수에 따라 크게 영향을 받지만, 또한 선택되는 부호어에 따라서도 영향을 받게 된다. 따라서, 0으로 할당된 한쪽 영역의 L_{min} 에서 $2^{(L_{min}-2)}$ 만큼 부호어들을 선택할 때 MRG[8]가 작은 부호어부터 선택을 한다.

본 논문에서 제안된 비대칭적 RVLC의 설계 방법은 다음과 같다.

(과정 1) Huffman 이진나무의 반쪽 영역에서 H_L 을 결정한다. 0으로 할당된 한쪽 영역이므로 실제 선택되는 수는 $2^{(L_{min}-2)}$ 이며, 부호어는 MRG가 작을수록 먼저 선택된다. 이 경우 Z_L 이 먼저 선택된다.

(과정 2) $\lceil S/2 \rceil$ 만큼 RVLC를 찾을 때까지 사용 가능한 모든 부호어를 검색한다. 모든 검색과정은 어두 조건과 어미 조건을 위배하는 부호어의 제거과정을 수반한다.

(과정 3) 한쪽 영역에서 먼저 선택된 부호어들과 이들을 비트 반전시킨 부호어들로 최종 비대칭적 RVLC를 구한다.

IV. 실험 결과 및 분석

1. RVLC 생성과정의 복잡도 분석

제안된 알고리즘에 대한 생성 과정의 복잡도 측정을 평가하기 위해서 최악의 경우를 가정하여 알고리즘의 성능을 보여주는 Big-Oh 표기법을 사용한다. Tsai 생성방법[8]의 복잡도 함수 $T_{conv}(N)$ 은 다음과 같다.

$$T_{conv}(N) = O\left(\frac{S^2}{2} \cdot (\sqrt{2})^S + \frac{S^2}{2} + S\right) = O(N^2 \cdot (\sqrt{2})^N) \Big|_{N=S} \quad (9)$$

또한 제안된 알고리즘 복잡도 함수 $T_{new}(N)$ 은 식 (10)과 같다.

$$T_{new}(N) = O\left(\frac{S^2}{4} \cdot (\sqrt[4]{2})^S + S + 1\right) = O(N^2 \cdot (\sqrt[4]{2})^N) \Big|_{N=S} \quad (10)$$

그리고, 기존의 알고리즘과 제안된 알고리즘 사이의 복잡도 차이를 식 (11)과 같이 복잡도 함수의 비로써 표현할 수 있다.

$$\frac{T_{conv}(N)}{T_{new}(N)} = O((\sqrt[4]{2})^N) \Big|_{N=S} \quad (11)$$

따라서, 최악의 경우에 부호화 되는 심볼의 수 S에 따라 약 $(1.19)^S$ 정도로 제안된 알고리즘의 성능이 우수하다. 즉, 제안된 알고리즘은 검색영역을 50%정도 줄이고, H_L 적응법으로 주어진 확률분포에 대한 적응과정을 간단히 하여 복잡도 성능이 개선되었음을 알 수 있다.

2. 부호화 성능의 비교

표 2는 Takishima와 Tsai의 방법과 제안된 방법을 가지고 영어 알파벳[7, 8]에 대하여 비대칭적 RVLC를 생성하여 평균부호길이를 비교한 결과를 보여준다. 표 3에서 C1, C2, C3은 각각 Huffman 부호, Takishima의 방법, Tsai의 방법을 통해 설계된 부호이다. Tsai는 MRG를 기반으로 부호어를 선택하여 평균부호길이가 2.7%정도 줄어든 RVLC를 구하였다. 그런데, C2와 C3의 결과를 보면 레벨 3과 레벨 9에서 사용가능한 부호어들이 더 존재하는데도 불구하고 찾지 못했음을 알 수 있다. 더욱이 L_{min} 에서 선택되는 비대칭적인 부호어의 수는 효율성에 큰 영향을 미치는데 $n_{Huff}(3)$ 와 같은 2개만이 선

표 2. 영어 알파벳에 대한 비대칭적 RVLC 의 부호화 성능 비교

발생 확률		Huffman 부호 C1		Takishima 알고리즘 : C2		Tsai 알고리즘 : C3		제안된 알고리즘 : C4	
		L	부호어	L	부호어	L	부호어	L	부호어
E	0.14878570	3	001	3	001	3	000	3	000 (H ₃)
T	0.09354149	3	110	3	110	3	111	3	111 (H ₃)
A	0.08833733	4	0000	4	0000	4	0101	3	010 (H ₃)
O	0.07245796	4	0100	4	0100	4	1010	3	101 (H ₃)
R	0.06872164	4	0101	4	0101	4	0010	4	0011
N	0.06498532	4	0110	4	1000	4	1101	4	1100
H	0.05831331	4	1000	4	1010	4	0100	4	0110
I	0.05644515	4	1001	5	10010	4	1011	4	1001
S	0.05537763	4	1010	5	01100	4	0110	5	00100
D	0.04376834	5	00010	5	00010	5	11001	5	11011
L	0.04123298	5	00011	5	00011	5	10011	5	01110
U	0.02762209	5	10110	5	10111	5	01110	5	10001
P	0.02575393	5	10111	5	11100	5	10001	6	001011
F	0.02455297	5	11100	5	11111	6	001100	6	110100
M	0.02361889	5	11110	6	111101	6	011110	6	011110
C	0.02081665	5	11111	6	101101	6	100001	6	100001
W	0.01868161	6	011100	6	011101	7	1001001	7	0010100
G	0.01521216	6	011101	6	111011	7	0011100	7	1101011
Y	0.01521216	6	011110	8	01110011	7	1100011	7	0111110
B	0.01267680	6	011111	8	11101011	7	0111110	7	1000001
V	0.01160928	6	111011	9	111010011	7	1000001	8	00101011
K	0.00867360	7	1110100	9	011110011	8	00111100	8	11010100
X	0.00146784	8	11101011	9	011110011	8	11000011	8	01111110
J	0.00080064	9	111010101	10	1110101011	9	100101001	8	10000001
Q	0.00080064	10	1110101000	10	11101010011	10	0011101001	9	001010100
Z	0.00053376	10	1110101001	12	1110101000111	10	1001011100	9	110101011
평균부호길이		4.15572392		4.42607344		4.30677804		4.236589	

택되었다.

C4는 제안된 방법에 따라 생성된 비대칭적 RVLC 이다. 이를 C2, C3와 비교하면 각각 4.3%와 1.7%정도 더 짧은 평균부호길이를 가진다. 그리고 제안된 알고리즘에 따라 비트 반전 쌍으로 구성되고 최상위 레벨의 부호어를 MRG를 기반으로 강제 선택한 후 나머지 하위 레벨에서 어떠한 부호어도 놓치지 않았다. 그리고 효율성에 큰 영향을 미치는 최상위 레벨에 적절한 수의 부호어들이 선택되어 더 효율적인 RVLC를 얻을 수 있음을 알 수 있다.

V. 결론

본 논문에서는 Huffman 부호를 기반으로 비대칭적 RVLC를 생성하는 새로운 알고리즘을 보였다. 제안된 알고리즘은 먼저 H_L적응법을 이용하여 한쪽 영역에서 부호어를 선택한 후 비트 반전 연산을 통해 최종 비대칭적 RVLC를 생산한다. 제안된 H_L적응법과 비트 반전 연산은 생성 과정의 복잡도를 줄이고 부호어를 놓치지 않아 효율성을 개선시킨다. 또한 가장 긴 부호어의 비트 길이를 크게 증가시키지 않아 한정된 용량의 시스템에 적합한 부호어를 설계한다. 실험 결과를 통해 효율성과 복잡도 측면에서 우수한 RVLC를 얻을 수 있다.

감사의 글

본 연구는 광주과학기술원(K-JIST)과 광주과학기술원 실감방송 연구센터를 통한 대학IT연구센터(ITRC), 그리고 교육부 두뇌한국21(BK21) 정보기술사업단의 지원에 의한 것입니다.

참고문헌

[1] D. Huffman , "A method for the construction of minimum redundancy codes," *Proc. Inst. Radio. Engr.*, vol. 40, pp. 1098-1101, Sept. 1952.

[2] J.J. Rissanen and G.G. Langdon. Jr., "Arithmetic coding," *IBM J. Res. Develop.*, 23, pp. 149-162, 1979.

[3] ISO/IEC 14496-2, "Information Technology – Coding of audio / video objects," *Final Draft Int. Std.*, Part 2 : Visual, Oct. 1998.

[4] ITU-T Rec. H.263, "Video coding for low bit rate communications," Annex V, 2000.

[5] A.S. Fraenkel and S.T. Klein, "Bidirectional Huffman coding," *Comp. J.*, vol. 33, no. 4, 1990.

[6] J. Wen and J. D. Villasenor, "A class of reversible variable length codes for robust image and video coding," *Proc. IEEE Int. Conf. Image Processing*, vol. 2, pp. 12-49, 1997.

[7] Y. Takishima, M. Wada and H. Murakami, "Reversible variable length codes," *IEEE Trans. Comm.*, vol. 43, pp. 158-162, Feb. 1995.

[8] C.W. Tsai and J.L. Wu, "On constructing the Huffman-code-based reversible variable-length codes," *IEEE Trans. Comm.*, vol. 49, pp. 1506-1509, Sept. 2001.

[9] W.H. Jeong and Y.S. Ho, "Design of symmetrical reversible variable-length codes from the Huffman code," *Picture Coding Symposium*, pp. 135-138, April 2003.