

# 블록간 상대거리에 의한 정점기반 모양정보 부호화 기법

정희원 이진학\*, 정재원\*\*, 문주희\*\*\*, 김재균\*

## Vertex-based shape coding based on the inter-segment distance

Jin-hak Lee\*, Jae-won Chung\*\*, Joo-hee Moon\*\*\*, Jae-kyoon Kim\* *Regular Members*

### 요 약

본 논문에서는 정점기반 모양정보 부호화의 부호화 효율 향상을 위한 새로운 정점의 위치 정보 부호화 방법을 제안한다. 정점들을 일차원 화소열에 위치시키고, 일정한 길이의 블록으로 나누어 정점을 포함하는 블록사이의 상대거리를 가변길이 부호화 한다. 블록 사이의 상대 거리에 대한 확률 분포를 분석하고, 확률 분포가 정점의 수와 블록 수의 비율에만 의존한다는 것을 증명한다. 부호화 효율과 구현상의 복잡도를 고려한 비율을 결정하고 그 비율의 확률 분포에 맞는 허프만 부호 표를 구성한다. 입력된 영상은 그 비율에 따라 블록의 길이가 결정되며, 허프만 부호 표를 사용하여 블록 사이의 상대 거리를 가변길이 부호화 한다. 제안된 부호화 방법은 정점의 수가 많은 영상에서 부호화 효율이 좋아진다.

### ABSTRACT

In this paper, we propose a new coding method based on the distance between vertex segments for vertex positions in the vertex-based shape coding. The pixel lines are divided into the segments of a fixed length, and the segments that have vertex pixels are called vertex segments. We analyze the probability distribution of the relative distance between vertex segments and prove that it depends only on the ratio between the number of vertices and the number of segments. Considering the coding efficiency and implementation complexity, we choose a particular ratio to make a code table. For each input image, the segment size is chosen according to the ratio, and the relative segment distances are entropy coded. It is shown that the proposed method is efficient for the images with many vertices.

### I. 서 론

정점기반 모양정보 부호화(vertex-based shape coding)는 컴퓨터 비전, 패턴 인식 등 컴퓨터 영상 처리 분야에서 영상을 해석하고 표현하기 위하여 많이 연구되어 왔다<sup>[1,2]</sup>. 또한 동일한 성질을 갖는 영역으로 영상 부호화하여 영역별로 부호화하는 부호화 기반 부호화 방법(segment-based coding) 이나, 영상 내부의 물체들을 독립적으로 부호화하는 물체기반 부호화 방법(object-oriented coding)에서 영역이나 물체의 모양 정보(shape information)를 부호

화 하기 위하여 사용되어 왔다<sup>[3,4,9,14]</sup>. 최근에는 물체 단위의 처리를 기반으로 하는 MPEG-4의 표준화 과정에서 물체의 모양 정보를 부호화하기 위해 MMR(modified modified reed)<sup>[5,9]</sup>, CAE(context based arithmetic encoding)<sup>[6,9]</sup>, chain coding<sup>[7]</sup>, baseline-based shape coding<sup>[8,9]</sup> 등과 함께 많은 연구가 수행되었다.

정점 기반 부호화는 특히 손실 부호화에서 윤곽선(contour)을 부드럽게 재구성함으로써 주관적 화질(subject quality)에서 매우 우수한 성능을 보여주며<sup>[10]</sup>, 재현 오차에 따라서 정점들을 점진적으로 전송하는 신축적 모양정보 부호화(scalable shape

\* 한국과학기술원 전기및전자공학과 영상통신연구실(jhlee@visnet.kaist.ac.kr, jkkim@ee.kaist.ac.kr)

\*\* 현대전자산업주식회사 (jwchung98@hei.co.kr)

\*\*\* 세종대학교 (jhmoon@kunjasejong.ac.kr)

논문번호 : 00087-0306, 접수일자 : 2000년 3월 6일

coding)가 용이하다<sup>[11]</sup>. 또한 선택된 정점(vertex) 들이 윤곽선의 높은 곡률에 위치해 있어서 모양정보의 특징 점을 포함하고 있기 때문에 MPEG-7을 비롯한 영상 데이터베이스를 위한 연구에도 사용되고 있다<sup>[12,13]</sup>.

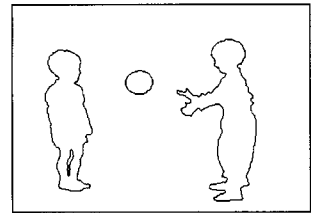
정점 기반 부호화는 선택된 정점들을 직선 또는 곡선(spline)으로 연결하여 원래의 윤곽선이 가장 잘 표현되도록 정점을 선택하는 과정과 선택된 정점들의 위치를 부호화 하는 과정으로 나누어지며, 이 두 과정이 대부분의 부호화 효율을 결정짓게 된다. 그림 1은 정점기반 윤곽선 근사화를 나타낸다. 원래의 윤곽선(a)을 근사화 하기위한 정점들(b)이 선택되고, 선택된 정점들을 직선으로 연결하여 근사화된 윤곽선(c)을 얻는다.

정점 선택(vertex selection) 방법은 윤곽선에서 높은 곡률을 갖는 위치를 선택하는 방법<sup>[12]</sup>과 정점들을 연결한 근사 윤곽선이 허용오차 범위 이내가 될 때까지 오차가 큰 위치에 반복적으로 정점을 선택하는 방법<sup>[3]</sup>들이 연구되었다. 최근에는 재현 오차와 발생 비트량을 모두 고려한 최적의 정점 선택 방법들이 연구되고 있다<sup>[10]</sup>.

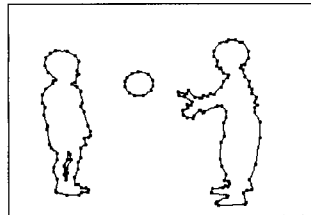
정점 부호화(vertex coding)는 선택된 정점의 위치 정보를 부호화 하는 과정으로 표준화에 필수적으로 포함되는 부분이며, MPEG-4 표준화 과정에서 많은 연구가 이루어졌다<sup>[4,9]</sup>. 정점 부호화는 정점들 간의 상대 거리를 가변길이 부호화 하는 방법들이 많이 연구 되었지만, 정점간 상대 거리에 대한 확률 분포가 영상의 종류와 허용오차의 범위에 따라서 크게 달라지기 때문에 하나의 허프만 부호표를 사용하는 것은 영상의 특성에 따라서 선택적으로 고정길이 부호화 하는 방법 보다 부호화 효율이 좋지 않다<sup>[4]</sup>. 가변길이 부호화 방법으로 부호화 효율을 얻기 위해서는 영상마다 다른 허프만 부호 표를 사용하거나 적응 산술 부호화(adaptive arithmetic coding) 방법으로 부호화 하는 것이 가능하지만 복잡도가 매우 높다는 단점이 있다. 특히 적응 산술 부호화는 전송 에러에 매우 민감하고, 발생된 정점의 수가 많지 않은 경우 통계 특성에 잘 적응 하지 못하여 부호화 효율이 나빠지는 단점이 있다.

본 논문에서는 복잡도가 낮고 발생되는 정보량이 정보원의 엔트로피에 근접하는 블록간 상대 거리에 의한 정점 부호화 방법을 제안한다. 정점들의 위치 정보를 일차원 화소열에 위치시키고, 일정 개수의 화소들을 단위로 블록을 정하고, 정점들이 포함되어 있는 블록간의 상대 거리를 가변길이 부호화한다.

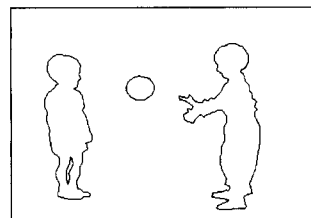
제안 방식에서는 블록의 길이를 가변 시킴으로써 블록간의 상대 거리에 대한 확률 분포가 영상의 종류나 근사화 정도에 관계없이 언제나 거의 같도록 한다. 이러한 특징은 영상의 종류에 관계없이 모든 영상에서 동일한 허프만 부호 표를 사용하여 가변 길이 부호화할 수 있다는 장점을 가져온다. 각 정점에 대해서 두 가지 정보를 부호화 하는데, 하나는 바로 이전 정점이 포함된 블록과의 상대 거리를 가변길이 부호화하는 것이고, 다른 하나는 블록 내에서 그 정점의 위치를 고정길이 부호화 하는 것이다.



(a) 원 윤곽선



(b) 선택된 정점들



(c) 정점들을 직선으로 연결하여 재생된 윤곽선

그림 1. 정점기반 부호화에 의한 윤곽선 근사화. (Children-kids 영상의  $D_0$ 의 경우).

본 논문에서는 정점들이 포함되어 있는 블록간의 상대 거리에 대한 확률 분포를 분석하고, 그 확률 분포가 정점의 수와 블록수의 비율에만 의존한다는 것을 증명한다. 그리고 부호화 효율과 구현상의 복잡도를 고려하여 그 비율을 결정하고, 그 비율의 확률 분포에 맞는 간단한 허프만 부호 표를 구성한다.

제 2장에서는 제안된 정점의 위치정보 부호화 방법에 대해서 설명하고, 제 3장에서는 블록간 상대

거리에 대한 확률분포를 분석한다. 제 4장에서는 부호화 효율과 구현상의 복잡도를 고려한 허프만 부호 표의 구성 방법을 설명한다. 제 5장에서는 모의 실험을 통해 성능을 분석하고 제6장에서 결론을 맺는다.

## II. 블록간 상대 거리에 의한 정점의 위치 정보 부호화 방법

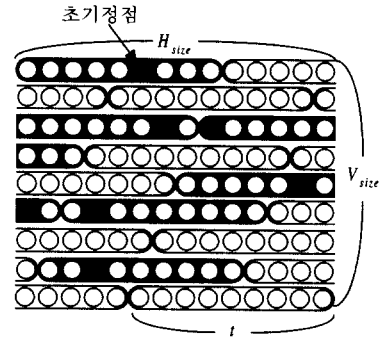
### 1. 초기 정점 부호화

하나의 윤곽선을 표현하기 위한 정점들은 직선 또는 곡선(spline)으로 연결하여 윤곽선을 재구성 하게 되는데 연결되는 순서에 따라서 정점들의 위치 정보를 부호화 하게 된다. 이때 제일 처음 부호화 되는 정점을 초기 정점이라고 정의한다. 모든 윤곽선은 각각 한 개의 초기 정점을 갖고 있으며, 4개의 윤곽선이 있는 그림 1에서는 4개의 초기 정점이 존재한다. 초기 정점을 부호화하는 기존의 방법들은 일반적으로 영상 내에서의 해당 정점의 수평, 수직 위치를 고정길이 부호화(fixed length coding)한다 [4,9].

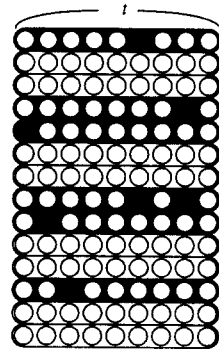
윤곽선의 수가 많지 않은 영상의 경우 초기 정점의 발생 비트 수는 전체 부호화 비트 수에 비하여 크지 않지만 윤곽선이 많은 영상의 경우에는 초기 정점을 무시할 수 없다. 특히 허용 오차가 큰 손실 부호화(lossy coding)의 경우에도 무손실 부호화(lossless coding)나 허용 오차가 작은 경우와 거의 같은 양의 위치 정보를 보내야 한다는 점에서 허용 오차가 큰 경우 초기 정점의 비중이 매우 커지게 된다.

초기 정점들이 속한 블록간의 상대 거리를 구하기 위해서 먼저 다음과 같이 일정한 길이의 블록으로 영상을 나눈다. 영상의 최상단에서 우하단까지 래스터 주사(raster scan) 방향으로 진행하면서 일정한 길이의 연속된 화소들의 단위로 블록을 정한다. 그림 2의 (a)는 7개의 초기정점이 존재하고 수평, 수직 방향으로 각각 14화소, 9화소 크기를 가지는 영상을 9화소 단위로 블록을 나눈 예이다. 블록들을 차례로 정렬시키면 그림 2의 (b)와 같이 14개의 블록들로 재구성된다.

영상의 수평, 수직 방향의 크기, 해당 영상에서의 윤곽선의 수를 각각  $H_{size}$ ,  $V_{size}$ ,  $N_c$ 라고 정의하고,  $i$ 번째 윤곽선을 표현하는 정점들의 수를  $N_i$ 이라고 정의한다. 하나의 영상에서 정점들의 집합  $V$ 는 다음과 같이 정의한다.



(a) 일정한 길이의 블록으로 나누어진 영상



(b) 정렬된 14개의 블록

그림 2. 제안된 초기 정점 부호화를 위한 블록 분할의 예

$$V = \{v_k^i = (x_k^i, y_k^i) | 0 \leq i < N_c, 0 \leq k < N_i\} \quad (1)$$

여기서  $(x_k^i, y_k^i)$ 는 영상 내에서  $i$ 번째 윤곽선의  $k$ 번째 정점에 대한 수평, 수직 방향 위치를 나타낸다. 이때 초기 정점들의 집합  $V_0$ 는 다음과 같다.

$$V_0 = \{v_0^i = (x_0^i, y_0^i) | 0 \leq i < N_c\} \quad (2)$$

$i$ 번째 초기 정점의 위치  $v_0^i$ 를 이차원 위치라고 하면, 래스터 주사 순서에서 영상의 시작점으로부터 그 화소의 위치를 나타내는 일차원 위치  $s_0^i$ 는 다음과 같다.

$$s_0^i = y_0^i \cdot H_{size} + x_0^i \quad (3)$$

초기 정점의 위치를 부호화 하는 기존의 방법은  $\lceil \log_2(H_{size} \times V_{size}) \rceil$ 의 비트를 사용하여  $s_0^i$ 를 고정길이 부호화한다. 여기서  $\lceil c \rceil$ 는  $c$ 보다 큰 정수 중 최소값을 나타낸다. 블록의 길이를  $t$ 로 정의할 때,

일차원 위치  $s_i^0$ 는 다음과 같이 나타낼 수 있다.

$$s_i^0 = b_i^0 \cdot t + a_i^0 \quad (4)$$

여기에서  $b_i^0$ 는  $i$ 번째 초기 정점이 존재하는 블록의 영상 내에서의 위치를,  $a_i^0$ 는  $i$ 번째 초기 정점의 블록 내에서의 위치를 각각 나타낸다.

먼저 영상의 수평, 수직 크기와 윤곽선의 수는 각 영상에 대하여 부가 정보로 수신단에 전송되는 것으로 가정한다. 이러한 가정하에 블록의 길이  $t$ 가 영상의 수평, 수직 크기와 윤곽선의 수에 의해 결정된다면 각 초기 정점은 좌표  $(x_i^0, y_i^0)$  대신  $(a_i^0, b_i^0)$ 를 사용하여 그 위치를 표시할 수 있다. 먼저 블록 내에서 그 화소의 위치를 나타내는  $a_i^0$ 는  $\lceil \log_2 t \rceil$ 의 비트를 사용하여 고정길이 부호화한다. 다음에는 블록의 위치를 나타내는  $b_i^0$ 를 부호화하기 위해서 바로 이전 초기 정점을 포함하고 있는 블록의 위치  $b_{i-1}^0$ 과의 상대 거리  $d_i^0 = b_i^0 - b_{i-1}^0$ 를 부호화 한다.

III장과 IV장에서 자세히 설명되지만, 본 논문에서 제안하는 방식의 가장 중요한 특징은 블록의 길이  $t$ 를 영상의 크기와 윤곽선의 수에 따라서 변화 시킴으로써 블록간의 상대 거리  $d_i^0$ 에 대한 확률 분포를 모든 영상에 대해서 동일하게 한다는 것이다. 이러한 특징은 미리 결정된 확률 분포에 맞는 허프만 부호 표를 구성할 수 있게 하고, 영상의 종류에 관계없이 같은 허프만 부호 표를 사용하여  $t$ 를 가변 길이 부호화할 수 있도록 한다.

## 2. 정점 부호화

초기 정점을 제외한 나머지 정점들에 대한 부호화 방법은 많이 연구되어 왔다. 식 (1)로부터 임의의 한 윤곽선 안에 존재하는 정점들의 집합을  $\{v_k = (x_k, y_k) \mid 0 \leq k < N_v\}$ 으로 정의하면 정점 부호화 방법은 일반적으로 정점들 사이의 상대 거리의 집합  $R$ 을 부호화 한다.

$$R = \{r_k = (r_{k,x}, r_{k,y}) \mid 1 \leq k < N_v\} \quad (5)$$

여기서  $r_k = v_k - v_{k-1}$ 이다.

집합  $R$ 의 확률 분포는 영상의 종류와 근사화 정도에 따라서 크게 다르기 때문에 하나의 허프만 부호표를 사용하여 부호화 하는 것은 부호화 효율을 얻기 어렵다<sup>[4]</sup>. 이를 극복하기 위하여 기존 방법에서는 현재 부호화 되는 윤곽선의 특성에 적응하여

부호화 한다<sup>[4,9]</sup>. 이 방법은  $(r_{k,x}, r_{k,y})$ 를 부호화 하는 대신 방향 정보  $\phi(r_k)$ 와  $(r_{k,major}, r_{k,minor})$ 정보를 부호화 한다.  $\phi(r_k)$ 는 극 좌표계(polar coordinate)에서  $360^\circ$ 를  $0^\circ$ 부터  $45^\circ$ 단위로 8개의 영역을 구분했을 때  $r_k$ 가 어느 영역에 속하는지를 나타내는 값(octant)이고,  $(r_{k,major}, r_{k,minor})$ 는 각각  $r_{k,x}$ 와  $r_{k,y}$ 의 절대값 중 큰 값과 작은 값으로 정의된다.  $\phi(r_k)$ 정보를 이용하여  $r_{k,x}$ 와  $r_{k,y}$ 의 크기를 비교할 수 있으므로  $\phi(r_k)$ ,  $(r_{k,major}, r_{k,minor})$ 는 바로  $r_k$ 를 나타낸다.

$\phi(r_k)$ 는 이전의 방향정보  $\phi(r_{k-1})$ 을 이용하여 효율적으로 가변길이 부호화가 가능하고,  $r_{k,minor}$ 는  $r_{k,major}$ 보다 항상 작다는 점을 이용하여 효율적인 부호화가 가능하다. 그렇지만  $r_{k,major}$ 는 윤곽선마다 확률 분포가 크게 다르기 때문에 하나의 허프만 부호표를 사용하여 가변길이 부호화 하는 것은 효과적이지 못하므로  $r_{k,major}$ 의 최대값을 기준으로 고정길이 부호화 한다. 즉,  $\lceil \log_2 \left( \frac{\text{Max}(r_{k,major})}{v_k} \right) \rceil$ 를 윤곽선마다 부가정보로 전송하고 이 비트 수를 사용하여 고정길이 부호화 한다.

본 논문에서는 가장 발생 비트량이 많은 주요소( $r_{k,major}$ : major component)의 가변 길이 부호화 방법을 제안한다. 먼저 그림 3과 같이 일차원 화소열 상에  $N_v-1$ 개의 주요소 값을 연속적으로 표시한다. 그러면 식 (4)와 같은 방법으로 주요소에 대한 일차원 화소열 상의 위치  $s_k$ 를 다음과 같이 나타낼 수 있다.

$$s_k = \sum_{i=1}^k r_{i,major} = b_k \cdot t + a_k \quad (6)$$

기존 방법에서 윤곽선마다 부가정보로 전송되는 고정길이 부호의 크기 대신 블록의 길이  $t$ 가 부가정보로 전송되는 것을 가정하면, 초기 정점 부호화에서와 같이  $(a_k, b_k)$ 를 부호화 함으로써  $r_{k,major}$ 를 표시할 수 있다. 앞 절에서 설명한 초기 정점 부호화와 같은 방법을 사용하여 블록 내에서의 위치  $a_k$ 는  $\lceil \log_2 t \rceil$ 비트로 고정길이 부호화 하고, 블록의 위치  $b_k$ 는 이전 블록과의 상대거리  $d_k = b_k - b_{k-1}$ 를 부호화 한다.

제안 방법은 블록의 길이  $t$ 를 정점의 수  $N_v$ 와  $\sum_{i=1}^{N_v} r_{i,major}$ 에 따라서 변화 시킴으로써 블록간의 상

대 거리  $d_k$ 에 대한 확률 분포를 동일하게 하여, 하나의 허프만 부호표를 사용하여 효율적으로 부호화가 가능하다.

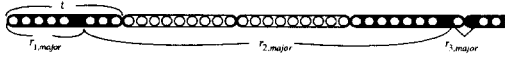


그림 3. 일 차원 직선 상에 위치한 주요소 ( $r_{k,major}$ )

### III. 정점들이 포함되어 있는 블록간의 상대 거리에 대한 확률분포

정점 부호화를 위한 제안 방법은 정점들을 일차원 화소열에 위치시키고 길이  $t$ 인 일정한 블록으로 나누어 정점이 포함된 블록들간의 상대거리를 가변 길이 부호화 함으로써 이득을 얻는다. 본 장에서는 가변길이 부호화를 위한 허프만 부호 표를 만들기 위해서 정점이 포함된 블록들 간의 상대거리에 대한 확률 분포를 분석한다.

일차원 화소열의 전체 화소 수를  $M$ , 그 위에 존재하는 정점의 수를  $n$ 이라고 정의한다. 일차원 화소열을  $t$  길이의 블록으로 나누면 블록 수  $m = \lceil M/t \rceil$  이 된다. 앞 절에서 설명한 초기 정점 부호화의 경우  $M$ 은  $H_{size} \times V_{size}$ 이며,  $n$ 은  $N_c$ 에 해당되고, 윤곽선의 정점 부호화의 경우  $M$ 은  $\sum_{k=1}^{N_v-1} r_{k,major}$ 이며,  $n$ 은  $N_v-1$ 에 해당된다.

정점들이 일차원 화소열에 독립적으로 위치되며 (i.i.d),  $t \rightarrow \infty$ 이고,  $M/t$ 는 정수라고 가정하면, 어떤 정점이 특정한 하나의 블록 안에 나타날 확률은  $1/m$  이므로, 특정 블록 안에  $j$ 개의 정점이 들어있을 확률은 다음과 같다.

$$\Pr\{J = j\} = \binom{n}{j} \left(\frac{1}{m}\right)^j \left(\frac{m-1}{m}\right)^{n-j} \quad (7)$$

여기서  $J$ 는 특정 블록 안에 존재하는 정점의 수를 나타낸다. 블록 안에 한 개 이상의 정점이 들어 있을 확률  $p_{vs}$ 는 식 (7)로부터 알 수 있듯이

$$p_{vs} = \Pr\{J \geq 1\} = 1 - \Pr\{J = 0\} = 1 - \left(\frac{m-1}{m}\right)^n \quad (8)$$

이다. 바로 이전 정점이 같은 블록 안에 들어있을 확률 즉, 정점이 존재하는 블록간 상대 거리를  $d$ 라고 정의할 때  $d$ 가 0일 확률  $p_0$ 는

$$p_0 = \Pr\{d = 0\} = 1 - \frac{p_{vs} \cdot m}{n} \quad (9)$$

이다. 정점이 들어 있는 블록들 역시 독립적으로 (i.i.d) 나타나므로  $m \rightarrow \infty$ 라고 가정하면, 블록 사이의 거리가  $l$ 일 확률  $p_l$ 은 다음과 같다.

$$p_l = \Pr\{d = l\} = (1 - p_0) \cdot p_{vs} \cdot (1 - p_{vs})^{l-1} \quad (10)$$

$l = 1, 2, \dots, m$

그러면  $p_0, p_l$ 의 분포를 살펴보기로 하자. 정점의 수와 블록 수의 비율을  $\gamma = \frac{n}{m}$ 으로 정의하면,  $m \rightarrow \infty, n \rightarrow \infty$ 일 때 포아송 (Poisson) 이론에 따라 식 (8)은 다음과 같이 된다<sup>[5]</sup>.

$$p_{vs} = \Pr\{J \geq 1\} = 1 - e^{-\gamma} \quad (11)$$

따라서 식 (8)의  $m$ 과  $n$ 이 충분히 크면  $p_{vs}$ 는 식 (11)의  $p_{vs}$ 에 근접하게 된다는 것을 알 수 있다. 식 (11)에서 확률  $p_{vs}$ 는  $\gamma$ 의 함수이므로, 식 (9)의  $p_0$ 와 식(10)의  $p_l$  역시  $\gamma$ 의 함수가 된다. 그러므로  $n$ 이 충분히 크다고 가정할 때,  $\gamma$ 가 일정하도록  $m$ 이 정해진다면  $M$ 의 크기나 정점의 수  $n$ 에 관계없이 블록간의 상대 거리에 대한 확률 분포는 동일하게 된다.

상기한 확률 분포의 분석이 정확한지를 보이기 위하여 일차원 화소열 상에 무작위로 발생된 정점의 모의 실험을 수행하고, 분석을 통한 이론치와 실험 결과를 비교한다. 실험에서  $t \rightarrow \infty$ 이고,  $M/t$ 는 정수가 되도록 하였다. 그림 4는  $\gamma=0.5$  일 때 정점의 수에 따른 블록간의 상대 거리에 대한 확률 분포를 나타낸다. 그림 4의 확률 분포는 식 (9)와 식 (10)에서 분석된 확률 분포와 오차가 매우 작고, ( $n \geq 30 (m \geq 60)$ )에서는 오차가 거의 없다. 발생하는 오차는  $m$ 이 작을수록 크게 발생하게 되는데 이것은 식 (10)을 유도할 때  $m$ 이 매우 크다는 가정에서 출발하였기 때문이다. 그림 4에서 알 수 있듯이 확률 분포 결과는  $\gamma$ 가 일정할 때 정점의 수  $n$ 에 따라서 크게 변화하지 않으며,  $n$ 이 커지면 확률 분포는 동일한 결과로 수렴하게 된다. 또한 이것은  $\gamma$ 의 값이 바뀌어도 동일한 경향을 보인다.

위에서 분석을 단순화하기 위해 가정한  $t \rightarrow \infty$ 이고,  $M/t$ 는 정수라는 가정은  $m$ 이 크고,  $t$ 가 1에 근접한 값이 아니면 그에 따른 오차는 실험에서 거의

발생하지 않는다. 또한 정점이 독립적으로 분포하는 것을 가정하였는데, 윤곽선 마다 하나만 존재하는 초기 정점의 경우 일반적으로 윤곽선이 독립적으로 위치한다고 생각할 수 있으므로 가정을 만족하고, 그 외의 정점들은 일반적으로 윤곽선에서 높은 곡률을 갖는 위치에 존재하는데 이것 또한 독립적으로 위치된다고 생각할 수 있다.

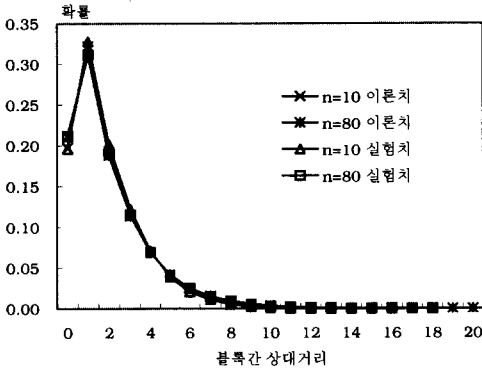


그림 4. 정점의 수  $n$ 에 따라서 블록간 상대 거리에 대한 이론적 확률 분포와 실험에 의한 확률 분포의 비교 여기서 비율  $\gamma=0.5$ 이다.

#### IV. 블록간 상대 거리의 확률 분포에 따른 가변길이 부호화

본 장에서는 부호화 효율과 구현상의 복잡도를 고려한  $\gamma$ 값을 결정하고, 그에 따른 허프만 부호 표를 구성한다. 정점이 존재하는 블록간 상대 거리에 대한 엔트로피  $H(m, n)$ 은 식 (9)와 식(10)의 확률  $p_0, p_1$ 로부터 다음과 같이 얻을 수 있다.

$$H(m, n) = -\sum_{d=0}^m p_d \log_2 p_d \quad (12)$$

고정길이 부호화를 사용할 때  $\log_2 m$ 비트를 사용하는 것으로 가정하면, 고정길이 부호화에 대한 정점 당 이득  $G$ 는 다음과 같다.

$$G = \log_2 m - H(m, n) \quad (13)$$

$n$ 이 일정하다고 가정할 때, 이득이 최대가 되는  $\gamma$ 의 값을 알아보기 위해서  $\gamma$ 의 변화에 따른 이득  $G$ 의 결과를 그림 5에 표시하였다. 그림 5의 엔트로피에 대한 그래프에서 알 수 있듯이  $\gamma$ 가 감소함에 따라서 정점 당 이득은 증가한다. 따라서 작은 값의

$\gamma$ 를 선택할수록 부호화 효율이 좋아지게 된다. 그러나  $\gamma$ 가 어느 일정한 값 이하인 경우에는 이득이 거의 증가하지 않음을 알 수 있다.

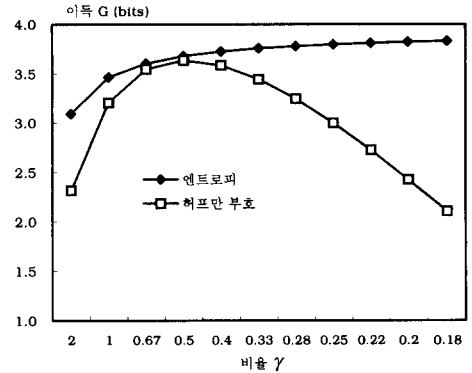


그림 5. 정점의 수  $n$ 이 일정할 때 ( $n=40$ ) 정점의 수와 블록수의 비율  $\gamma$ 의 변화에 따른 정점 당 이득.

실제의 영상에서는  $\gamma$ 가 무한히 작아질 수 없다.  $\gamma$ 가 작다는 것은  $m$ 이 크다는 것을 의미하고,  $m$ 은 영상의 크기보다 커질 수 없다는 점에서  $\gamma$ 가 어느 정도 이상 작아지는 것은 곤란하다. 따라서 구현의 입장을 고려하면 어떤 크기의 영상에도 모두 적용될 수 있도록 이득에서 크게 손해보지 않는 범위 내에서 비교적 큰 값의  $\gamma$ 를 선택해야 하겠다.

구현을 간단히 하기 위해서는 단순한 허프만 부호 표가 구성되도록  $\gamma$ 가 선택되어야 한다. 허프만 부호 표의 부호 수는  $m$ 개로 구성되고,  $m$ 은  $M$ 까지 증가할 수 있으며, 영상에 따라서  $M$ 이 매우 커질 수 있으므로 부호 수는 무한대가 가능해야 한다. 무한대의 부호로 구성되는 허프만 부호 표를 만드는 하나의 방법은 1 큰 거리는 발생 확률이 1/2 작도록 하는 것이다. 이렇게 하면 거리가 1 커질 때 마다 1비트씩 늘어나는 부호가 되어 부호 표의 구성이 간단해진다. 이렇게 되려면 식 (8) 과 식 (11)의  $p_1$ 가 0.5가 되고, 식 (11)에서  $\gamma$ 는 0.69가 된다. 그러나  $p_1$ 은  $0.5 \cdot p_0$ 가 되지 않기 때문에 이 부분에서 손실이 발생하고, 그림 5에서 알 수 있듯이  $\gamma$ 가 크면 이득이 작아지므로 비록 거리가 1 커질 때 마다 1비트씩 늘어나는 부호를 구성한다고 하더라도  $\gamma=0.69$ 가 최적이지 않다.

$\gamma$ 가 0.69보다 작은 값이 최적이라고 가정할 때,  $\gamma$ 가 0.69에 근접하는 값이면  $p_1 > p_0 > p_2$  이고  $2 \cdot p_1 > p_2$  이다. 그러므로 거리에 따라서 1씩 증가하는 부호를 구성하면서 동시에  $\gamma$ 에 따른 상대 거리

의 확률 분포를 고려하여 표 1의 허프만 부호화 표가 얻어진다. 표 1의 부호를 사용할 때 정점 당 비트 수는 다음과 같다.

$$H_{Huffman}(m, n) = \sum_{d=0}^m p_d L_d \quad (14)$$

여기서  $L_d$ 은 블록간 상대 거리가  $d$ 인 부호의 길이를 나타낸다.

그림 5의 허프만 부호에 대한 그래프는 정점의 수  $n=40$ 일 때  $\gamma$ 의 변화에 따른 정점 당 이득을 나타낸다. 제안된 허프만 부호 표를 사용하는 경우  $\gamma=0.5$ 일 때 이득이 최대가 되며, 엔트로피에 의한 이득에 근접함을 알 수 있다. 그림 6은  $\gamma=0.5$ 일 때 정점의 수  $n$ 의 변화에 따른 정점 당 이득을 나타낸다. 표 1의 허프만 부호를 사용한 가변길이 부호화는  $n$ 이 3보다 작은 경우 고정길이 부호보다 손해이며,  $n$ 이 5보다 큰 경우는 엔트로피에 의한 이득에 근접함을 알 수 있다.

표 1. 블록간 상대 거리에 대한 허프만 부호 표

블록간 상대 거리( $d$ )	부호	부호 길이( $L_d$ )
0	10	2
1	11	2
2	01	2
3	001	3
4	0001	4
⋮	⋮	⋮
N	000...0001	N

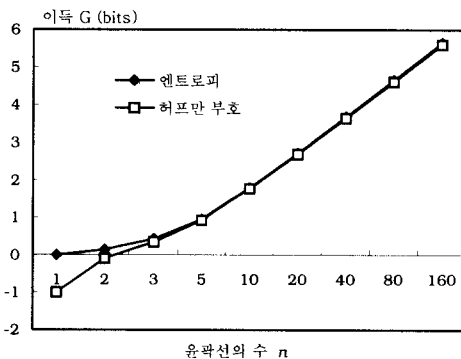
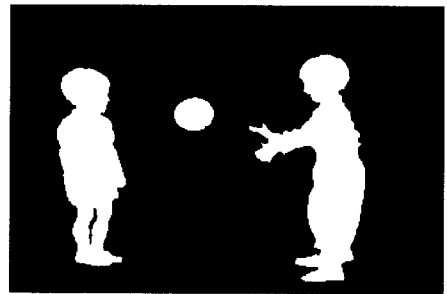


그림 6. 정점의 수와 블록수의 비율  $\gamma$ 가 일정할 때 ( $\gamma=0.5$ ) 정점의 수  $n$ 의 변화에 따른 정점 당 이득.

## V. 모의 실험

제안된 정점 부호화 방법의 성능을 확인하기 위해서 그림7에서 보여주는 MPEG-4의 실험 영상들을 사용한다<sup>[9]</sup>. Robot영상은 44프레임, Children-Kids, 그리고 Cyclamen 영상은 각각 100프레임씩 실험에 사용하며, 실험 영상의 크기는 SIF이다. 그림 7에서 알 수 있듯이, 실험 영상 Children-Kids는 윤곽선의 수가 비교적 적으며, Robot과 Cyclamen은 많은 수의 윤곽선을 포함하고 있다.



(a) Children-kids



(b) Robot



(c) Cyclamen

그림 7. 실험 영상

각 영상은 프레임 단위로 MPEG-4에서 정의된 VOP를 구성하게 되는데, VOP는 모양정보를 포함하는 최소의 직사각형이며, 16화소의 배수로 결정된

다. 프레임 내에서 VOP의 위치와 VOP의 크기는 부가 정보로 전송된다.

초기 정점 부호화에 대한 모의 실험 결과를 표 2에 나타내었다. 표 2에서 제안된 부호화 방법은 기존의 고정길이 부호화 방법과 비교하여 윤곽선의 수가 많은 Robot 과 Cyclamen 영상의 경우 정점당 각각 23.66%, 22.94% 이득이 발생하며, 윤곽선이 적은 Children-Kids 영상의 경우는 정점당 각각 5.3% 이득이 발생한다.

실험 결과에서 제안된 방법은 윤곽선의 수가 많을수록 큰 이득을 얻는다는 것을 알 수 있다. 이것은 고정길이 부호화와 가변길이 부호화를 모두 사용하는 제안된 방법에서 윤곽선의 수가 많은 경우 블록의 길이가 짧아지므로 고정길이 부호가 차지하는 비중이 작아지게 되고, 반대로 윤곽선의 수가 적은 경우는 고정길이 부호화가 차지하는 비중이 커지게 된다. 따라서 가변길이 부호화에 의해 이득을 얻게 되는 제안된 방법은 윤곽선의 수가 많을수록 이득이 커지게 된다.

윤곽선의 수가 많은 경우 전체 윤곽선 부호화 비트량에서 윤곽선의 위치 정보를 부호화하는 비트 수가 차지하는 비중이 커진다. 예를 들어 정점 기반 부호화 방법에서 손실 부호화의 경우, 전체 윤곽선 부호화 비트량에서 윤곽선 위치 정보가 차지하는 비중은 Robot 과 Cyclamen 영상의 경우 6~30% 이며, Children-Kids 영상의 경우는 4~9%이다. 제안된 방법은 윤곽선의 수가 많을수록 이득이 커지게 되므로 윤곽선이 많은 영상에서 부호화 효율이 높게 된다.

표 2. 초기 정점 부호화에 대한 기존 방법과 제안된 부호화 방법의 비교

실험영상	프레임당 윤곽선의 수	엔트로피 (비트)	기존방법		제안방법	
			정점당 비트수	정점당 비트수	이득 (%)	이득 (%)
Children-Kids	3.34	15.09	16.00	15.15	5.3	
Robot	27.30	11.82	15.89	12.13	23.66	
Cyclamen	28.76	12.83	17.00	13.10	22.94	

표 2에서 정점을 전송하기 위한 최소 정보량인 엔트로피를 표시하였다. 영상 내에 독립적으로 위치되는 (i.i.d) 정점의 최소 정보량은 영상 내에 존재하는 정점을 1, 나머지 화소들을 0으로 표시할 때, 0과 1의 이진 정보원에 대한 엔트로피를 정점의 수  $n$ 과 영상 내의 전체 화소의 수  $M$ 를 사용하여 환산함으로써 다음과 같이 얻는다.

$$H_{entropy} = -M(p \log_2 p + (1-p) \log_2 (1-p)) / n \quad (14)$$

여기서  $P=n/M$ 이다. 표 2의 엔트로피는 실험 영상의 정점들이 영상 내에 독립적으로 위치된다고 가정하였다. 표 2로부터 제안된 방법에 의해서 발생되는 정보량은 엔트로피에 근접함을 알 수 있다. 정점 부호화에 대한 기존 방법과 제안방법의 성능을 비교하기 위하여 그림 7의 MPEG-4 실험 영상에서 Children-kids와 Cyclamen영상을 사용하여 실험하였다. 성능 평가를 위한 재현오차는 MPEG-4에서 모양정보 부호화의 성능 평가를 위하여 정의된  $D_n$ 을 사용한다.

$$D_n = \frac{\text{에러화소의 수}}{\text{물체 내부 화소의 수}} \quad (15)$$

그림 8은 VOP당 주요소( $r_{k,major}$ )에 대한 부호화 비트 수를 비교하였다. 제안 방법은 기존 방법과 비교하여 Children-kids 영상에 대해서 0~11.0%, Cyclamen 영상에 대해서 0~10.2% 발생 비트수가 감소한다. 실험 결과에서 제안 방법은 재현오차가 작은 경우 발생 비트가 많이 감소하는 특징을 보인다. 이것은 재현오차가 작은 경우 발생하는 정점들의 수가 많아지므로 그림 6과 같이 상대적으로 개선되는 효과가 크다.

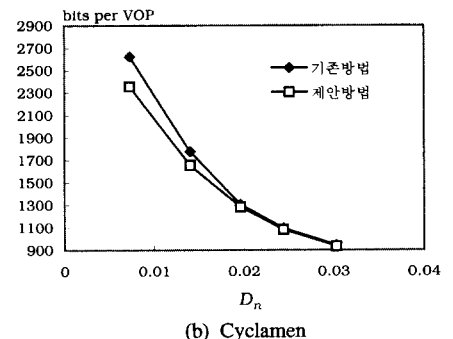
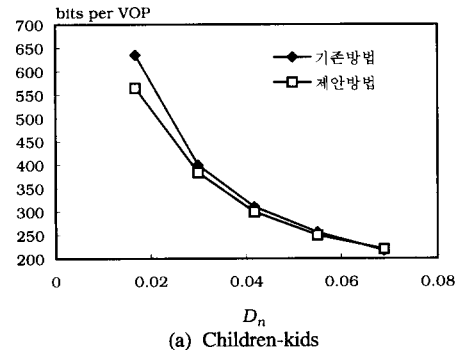


그림 8. VOP당  $r_{k,major}$ 의 부호화 비트수 와  $D_n$



그림 9는 위 두 경우를 동시에 적용하였을 때 기존 방법과 성능을 비교하였다. 기존 방법은 MPEG-4 표준화 과정의 core experiment에서 사용된 방법<sup>9)</sup>이며, 제안된 방법은 초기 정점들과 그 외의 정점들에 대해서 제안된 블록간 상대거리의 가변길이 부호화 방법을 적용하였다. 제안 방법은 기존 방법과 비교하여 Children-kids 영상에 대해서 0~4.8%, Cyclamen 영상에 대해서 3~4.9% 발생 비트수가 감소한다. Cyclamen 영상의 경우 초기 정점의 수가 많아서 재현 오차가 큰 경우 초기 정점 부호화에 의해 3~4% 비트수가 감소한다. 그림 1의 (b)는 Children-kids 영상에 대해서 재현 오차  $D_n$ 이 0.017 인 경우에 얻어진 정점들이며, 이 경우 재생된 윤곽선인 그림 1의 (c)는 원 윤곽선과 거의 같음을 보여 준다.

표를 사용함으로써 블록의 위치를 간단히 부호화할 수 있기 때문에 제안된 부호화 방법은 복잡도가 매우 낮다.

## VI. 결론

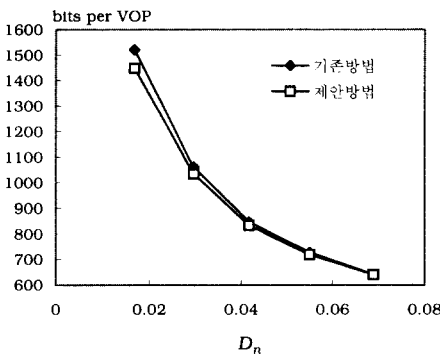
본 논문에서는 정점기반 모양정보 부호화의 부호화 효율 향상을 위해 현재 부호화되는 모양정보의 특성에 적응하는 새로운 정점의 위치정보 부호화 방법을 제안하였다. 정점들을 일차원 화소열에 위치시키고 일정한 블록으로 나누어 정점들이 포함된 블록간의 상대거리를 가변길이 부호화 함으로써 이득을 얻는다.

블록간 상대 거리에 대한 확률 분포를 분석하고, 그 확률 분포가 정점의 수와 블록 수의 비율에만 의존한다는 것을 증명하였다. 부호화 효율과 구현상의 복잡도를 고려하여 비율을 결정하였으며, 결정된 비율의 확률분포에 맞는 간단한 허프만 부호 표를 구성하였다. 모든 영상에 대해 그 비율에 따라 블록의 길이를 정함으로써, 동일한 허프만 부호 표를 사용하여 가변길이 부호화가 가능하게 되었다.

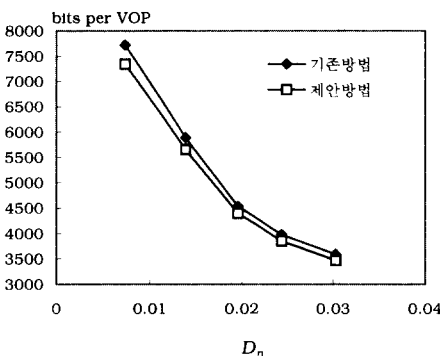
제안된 방법은 초기 정점들에 대해서 고정길이 부호를 사용하는 기존의 방법과 비교하여 5~23%의 부호화 이득이 발생하며, 초기 정점을 제외한 정점들의 상대거리 중 주요소( $r_{k,major}$ )에 대해서 0~11% 부호화 이득을 얻는다. 전체적으로 제안 방법을 적용한 정점기반 모양정보 부호화 방법은 기존 방법에 비하여 0~5% 부호화 이득을 얻는다.

제안 방법은 부호화 과정이 매우 단순하고, 간단한 허프만 부호 표를 사용하기 때문에 복잡도가 매우 낮으며, 위치 정보를 부호화 하는 다른 정점기반 부호화 방법에 대해서도 쉽게 적용이 가능하다. 정점의 위치정보 뿐만 아니라 영상간 상관성을 이용하는 정점기반 모양정보 부호화<sup>9,14)</sup>에서 움직임 보상이 가능한 윤곽선 세그먼트(segment)들의 위치 정보, 윤곽선의 무손실 재현을 위한 시슬 부호화(chain coding)에서 같은 방향 성분을 갖는 윤곽선 세그먼트(segment)들의 위치 정보<sup>14)</sup> 등에 대해서도 적용된다.

본 논문에서 제안하는 부호화 방법은 모양정보 부호화 뿐만 아니라 일차원 또는 다차원 평면에서 특정 위치들을 부호화 하는 응용 등에서 부호화 효율을 얻기 위해 사용될 수 있을 것이다.



(a) Children-kids



(b) Cyclamen

그림 8. VOP당  $r_{k,major}$ 의 부호화 비트수 와  $D_n$

제안 방법에서  $\gamma = 0.5$  일 때 블록의 길이  $t = M / 2n$ 이며,  $M$ 과  $n$ 은 이미 알고있는 값이고, 블록의 위치와 블록 내에서 정점의 위치는 식(4)와 식(6)에서와 같이 정점의 일차원 위치를 블록의 길이  $t$ 로 나누면 알 수 있다. 또한 표 1의 간단한 허프만 부호



