

논문97-2-2-02

1차원 정점과 정점 재배열을 이용한 효율적 정점기반 모양정보 부호화

정재원*, 문주희*, 김재균*

Efficient Vertex-based Shape Coding using One-dimensional Vertex and Vertex Reordering

Jae-Won Chung*, Joo-Hee Moon*, and Jae-Kyoon Kim**

요 약

본 논문에서는 1차원 정점의 선택 및 부호화와 정점 재배열을 이용하는 정점기반 이진 모양정보 부호화기를 제안한다. 기존의 물체 적응형 정점 부호화 방식과는 달리, 제안 방식에서는 추출된 정점들을 특성이 서로 다른 1차원 정점과 2차원 정점으로 분리한다. 1차원 정점은 제안된 부호화 방식을, 2차원 정점은 정점 재배열과 물체 적응형 정점 부호화 방식을 사용하여 수신단에 전송한다. 모의 실험 결과는 제안된 방식이 기존의 방식에 비해, 재현 오차의 변화 없이 부호화 비트 수를 최대 12 % 감소시키며, 부호화 이득이 모양정보 특성에 의존함을 보여준다.

Abstract

This paper presents a new vertex-based binary shape coding scheme using one-dimensional vertex selection/encoding and vertex reordering. In compared with the conventional object-adaptive vertex encoding(OAVE), the extracted vertices are, firstly, classified into one-dimensional(1D) vertices and two-dimensional(2D) vertices in the proposed method. For 1D vertices, new coding method proposed in this paper is performed. For 2D vertices, the vertex reordering and OAVE are carried out.

Experimental results show that the proposed vertex-based coding scheme reduces coding bits up to 12 % compared with the conventional one and its coding gain depends on the characteristics of contour.

I. 서 론

윤곽선 화소(contour pixel) 중에서 높은 2차원 곡률(curvature)을 가지는 정점(vertex)들은 모양정보(shape information)의 특성을 잘 표현하기 때문에, 영상 해석(image analysis)과 패턴 인식(pattern recognition) 분야에

서 정점 추출 방식(vertex selection method)에 관한 많은 연구가 수행되어 왔다[1,2,3,4]. 한편, 적은 수의 정점들을 연결하는 다각 근사화(polygonal approximation)가 많은 수의 윤곽선 화소들을 표현할 수 있으므로, 이러한 정점들을 기반으로 하는 부호화 방식(vertex-based coding)은 모양정보 압축(compression) 측면에서도 유리하다. 정점기반 모양정보 부호화 사용의 대표적 예로 임의의 모양 정보의 전송이 요구되는 물체별 동영상 부호화(object-based video coding)[5,6,7]가 있다.

최근에는 영상 내용을 단위로 부호화(content-based video coding)하는 등의 새로운 기능들을 포함하는 부호화 방식의 표준화 작업 MPEG-4에서, 모양정보의 (초)저속 부호화를 위한 정점기반 부호화의 연구가 수행되어 왔다

* 현대전자산업주식회사 멀티미디어 연구소
Multimedia R & D Center, Hyundai Electronics Industries Co., Ltd.

** 한국과학기술원 전기 및 전자공학과
Dept. of Electrical Engineering, KAIST.

[8,9,10,11]. 정점기반 부호화 방식의 성능을 결정하는 가장 중요한 부분은 정점 추출 방식과 정점 부호화(vertex encoding) 방식이다. MPEG-4에서는 정점 추출 방식으로 순환 정점 추출 방식(iterated refinement method, 이하 IRM)[5,6]이 널리 사용되어왔다. IRM은 근사 정도 또는 정점의 갯수에 대한 제어가 용이하고 수행이 간단해서 기존의 정점기반 부호화 방식에서 널리 이용되어 왔다. 추출된 정점의 부호화 방식으로 물체 적응형 정점 부호화(object-adaptive vertex encoding, 이하 OAVE)[10,12] 방식이 대표적이다. 이 방식은 정점의 방향을 8진(octant) 표현으로 나타내고, 각 정점의 상대 거리를 전송하기 위한 비트 할당을 정점의 특성에 의존하여 해당 율곽선 단위로 조정하는 방식이다.

이러한 정점 추출 방식과 정점 부호화 방식 이외에도, 전처리 과정(preprocessing), 근사 재현(approximate reconstruction), 재현 오차 부호화(reconstruction error encoding) 등이 정점기반 부호화의 구성 요소[12]이다. 입력 모양정보가 물체와 배경을 나타내는 이진 마스크(binary mask)일 때, 율곽선 추출을 위한 전처리 과정을 수행하며, 본 논문에서는 참고문헌 [10]의 방법을 사용한다. 근사 재현 방식으로는 본 논문에서 사용되는 다각 근사화와 함께 주관적 화질(subjective quality) 측면에서 유리한 스플라인(spline) 근사화[5,6] 등이 있다. 재현 오차 부호화에서는 원 율곽선(original contour)과 재현 율곽선(reconstructed contour) 사이의 재현 오차를 부호화하여 수신단에 전송하며, 부호화 방법의 대표적인 예로 1차원 DST[13]가 있다. 본 논문에서는 재현 오차 부호화를 고려하지 않는다.

이밖에도 MPEG-4 표준화 작업 과정에서 정점기반 부호화와 관련하여 많은 연구들이 진행되었다. 시간축상의 모양정보 중복성을 줄이는 방식의 연구[5,6,7], 신축형(scalable) 정점기반 부호화 방식에 관한 연구[14], error-resilient 부호화를 위한 정점기반 부호화 방식에 관한 연구[15], 고속 알고리즘을 위한 연구[16] 등이 있다.

본 논문에서는, MPEG-4에서의 이진 모양정보 부호화를 위한 정점기반 부호화 방식의 부호화 효율(coding efficiency)을 향상시키기 위하여, 1차원 정점의 선택 및 부호화(one-dimensional vertex selection/encoding), 그리고 정점 재배열(vertex reordering) 등을 제안한다. 이러한 제안 방식들은 원 율곽선과 재현 율곽선 사이의 재현 오차에는 영향을 미치지 않고, 부호화 비트 수만을 감소시킴으로써 높은 부호화 효율을 얻는다. IRM을 사용하여 추출된 정점들 중에서 서로 다른 특성을 가지고 있는 1차원 정점과 2차원 정점을 분리해낸다. 선택된 1차원 정점은 제안된 방식에 의하여 부호화 전송되고, 2차원 정점은 기존의 OAVE 방식에 의하여 부호화 전송된다. 이때, 정점의 상대 거리를 전송하기 위한 비트 수를 감소시키기 위하여 부호화 이전 단계에서 정점 재배열을 수행한다.

본 논문의 구성은 다음과 같다. 먼저 II에서 기존의

IRM과 OAVE를 간략히 설명한다. III에서는 1차원 정점의 선택 및 부호화 방식, 그리고 정점 재배열 방식을 자세히 설명한다. IV에서는 모의 실험 결과 및 검토를, 그리고 마지막 V에서는 결론 및 추후 연구 과제에 대하여 기술한다.

II. 순환 정점 추출 방식 및 물체 적응형 정점 부호화 방식

그림 1은 기존 정점기반 부호화 방식의 구성도를 보여준다. 기존의 정점기반 부호화 방식은 정점 추출 방식으로 IRM을 사용한다. IRM의 입력은 율곽선 화소 리스트(contour pixel list), $C = \{c_0, c_1, c_2, \dots, c_{N_c-1}\}$ 이고 출력은 정점 리스트(vertex list), $V = \{v_0, v_1, v_2, \dots, v_{N_v-1}\}$ 이다. N_c 와 N_v 는 율곽선 화소의 갯수와 정점 갯수를 각각 나타낸다. 추출된 정점은 OAVE 방식으로 부호화되어 수신단에 전송된다.

무손실 부호화(lossless coding)의 경우에는 모든 율곽선 화소들을 정점으로 추출한다. 그러므로 IRM과 같은 정점 추출 방식을 사용하지 않고, OAVE만 사용하면 된다. 본장에서 IRM과 OAVE 방식에 대하여 간략히 설명한다.

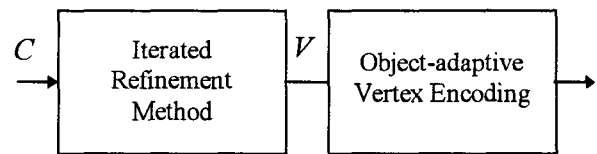


그림 1. 기존의 정점기반 모양정보 부호화 방식의 구성도. C와 V는 율곽선 화소 리스트와 정점 리스트.

Fig. 1. Block diagram of the conventional vertex-based shape coding where C and V are the contour pixel list and the vertex list, respectively.

1. 순환 정점 추출 방식[5,6]

일반적으로 다각형 근사화를 위한 정점 추출 방식은 두 가지로 분류 가능하다. 첫번째로는 높은 2차원의 곡률을 가지는 율곽선 화소를 corner detection 방법을 사용하여 찾고, 이를 정점으로 하는 방법이다[1,2]. 다른 하나는 주어진 정확도(accuracy criterion)를 만족시키도록 다각형 근사화를 수행하며 정점을 선택하는 방법이다[3,5,6]. 기존의 IRM은 정확도가 최대 절대 거리(maximum absolute distance), d_{max} 인 후자의 대표적인 예이다. 그러나 IRM에 의해 선택된 정점들도 높은 곡률을 가진다[3].

IRM의 과정을 보여주는 예가 그림2에 있으며, 자세한 설명은 다음과 같다:

- (제1단계) 입력된 윤곽선 화소들 중에서 가장 상대 거리가 큰 한 쌍의 화소들을 찾는다. 이 두 개의 윤곽선 화소들을 초기 정점으로 한다. (그림 2(a))
- (제2단계) 모든 윤곽선 화소에서 두 개의 초기 정점을 연결하는 직선과의 최대 거리가 d_{max} 보다 큰 경우, 해당 윤곽선 화소를 새로운 정점으로 한다. 그리고 새 정점을 포함하는 다각 근사화를 수행한다. (그림 2(b))
- (제3단계) 모든 윤곽선 화소에서 자신을 근사화하는 다각 직선과의 거리가 d_{max} 보다 작거나 같을 때까지, 최대 거리의 윤곽선 화소를 정점으로 하고 이를 포함하는 다각 근사화를 반복 수행한다. (그림 2(c))

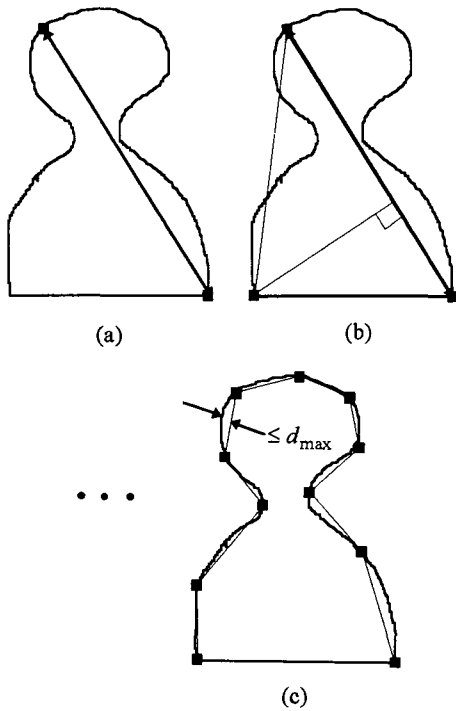


그림 2. 순환 정점 추출 방식의 과정 예.
Fig. 2. Procedure example of iterated refinement method.

제2단계와 제3단계에서 윤곽선 화소와 직선과의 거리 계산은 참고문헌 [3]의 방법을 사용한다.

2. 물체 적응형 정점 부호화 방식[10,12]

OAVE에서의 정점들의 부호화는 초기 정점과 비초기 정점들의 부호화로 나누어진다. 초기 정점은 절대 위치 (absolute address) 부호화를 수행한다. MPEG-4에서는 영상 신호를 Video Object Plane(VOP) 단위로 부호화한다.

초기 정점은 VOP 내부에 위치하며, 초기 정점의 절대 위치를 위한 비트 할당은 VOP 크기를 고려하여 결정된다 [10]. 비초기 정점의 위치 전송을 위한 기존의 모든 방식들에서는 이전 정점과의 상대 거리를 부호화한다 [5,6,7,10]. 이 중에서 OAVE 방식은 무손실 부호화($d_{max} = 0$)에서 d_{max} 가 큰 손실 부호화에 이르기까지, 넓은 범위의 d_{max} 에서 부호화 효율 측면의 높은 성능을 보인다 [16]. OAVE 방식은 정점들의 상대 거리를 8진(octant) 표현, 주요소(major component), 그리고 부요소(minor component)로 표현하고 이 순서대로 부호화한다.

8진 표현은 정점의 방향을 나타낸다. 인접한 정점들의 방향 사이에는 중복성이 존재하므로, 현재 정점의 8진 표현은 이전 정점의 8진 표현과의 차이를 Huffman 부호화하여 전송한다.

주요소와 부요소는 절대 크기 값을 전송하는데 이에 필요한 비트 할당은 해당 윤곽선에서 추출되는 모든 정점들의 상대 거리들 중에서 x-, y-방향 각각의 최대 값을 표현할 수 있도록 결정한다. 부요소는 주요소보다 항상 작거나 같기 때문에 부요소를 위한 비트 할당은 주요소 값도 고려되어 결정된다.

이러한 OAVE 방식은 윤곽선(혹은 정점)의 특성에 비트 할당을 의존하게 하는 장점을 가진다. 그러나 추출된 정점들 중에서 한 개의 정점이라도 상대 거리가 큰 경우에는 다른 모든 정점에도 큰 비트 할당이 되는 문제점이 있다.

무손실 부호화의 경우에는 정점(윤곽선 화소)들이 바로 이웃한 화소에 위치하므로, 8진 표현만을 전송하고 주요소와 부요소는 전송하지 않는다. 이러한 경우, OAVE는 부가 정보를 제외하면 기존의 differential chain coding과 같다.

III. 1차원 정점과 정점 재배열

본 논문에서 제안하는 정점기반 부호화 방식의 구성도가 그림 3이다. II에서 설명한 IRM으로 정점을 추출한 다음, 비초기 정점들 중에서 1차원(one-dimensional: 이하 1D) 정점을 제안된 방식을 사용하여 선택한다. 나머지 비초기 정점들은 2차원(two-dimensional: 이하 2D) 정점으로 한다. 1D 정점의 이전 정점까지의 상대 거리는 VOP내의 모든 물체 영역을 포함하는 최소 직사각형의 경계(이하 VOP 경계) 상에서의 1차원 거리이고, 2D 정점에서의 상대 거리는 VOP 내부에서의 2차원으로 표현된다. 다음 단계에서는 2D 정점들을 기준으로 정점 재배열(vertex reordering)을 수행한다. 마지막으로 1D 정점을 위한 제안된 부호화 방식과 2D 정점을 위한 OAVE 방식을 사용하여 정점들을 부호화한다. 이러한 1D 정점의 도입이나 정점 재배열은 재현 오차에는 영향을 미치지 않으며, 부호화 비트 수의 감소만을 가져온다.

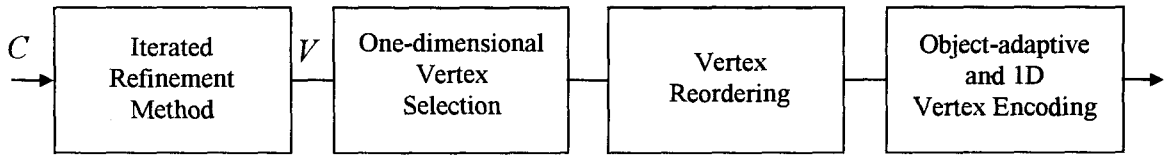


그림 3. 제안된 정점기반 모양정보 부호화 방식의 구성도. C 와 V 는 윤곽선 화소 리스트와 정점 리스트.
Fig. 3. Block diagram of the proposed vertex-based coding where C and V are the contour pixel list and the vertex list, respectively.

무손실 부호화의 경우에는 기존 정점기반 부호화 방식과 마찬가지로 IRM을 수행하지 않는다. 그러나 추출된 정점(윤곽선 화소) 중에서 1D 정점의 선택은 수행한다. 2절에서 설명되듯이 정점 재배열은 무의미하므로 수행하지 않는다. 본 장에서는 1D 정점의 선택 및 부호화 방식, 그리고 정점 재배열 방식에 관하여 차례로 기술한다.

1. 1차원 정점의 선택 및 부호화 방식

정점기반 부호화 방식은 2차원의 높은 꼭를을 가지는 몇 개의 정점으로 윤곽선을 표현하는 방식이다. 그러나 VOP 경계에 위치하는 정점은 윤곽선의 특성이 아닌 외적인 측면(예를 들면, 카메라 범위(camera scope)의 한계)에 기인한다. 이러한 정점들은 물체의 윤곽선 특성에 의해 추출되는 정점들과 구분하여 부호화를 수행할 시, 부호화 이득을 꾀할 수 있다. 이러한 동기에 의해 본 논문에서는 1D 정점의 선택과 부호화 방식을 제안한다.

가. 1차원 정점의 선택

제안된 1D 정점 선택 방식은 다음과 같다: IRM을 수행한 후 정점 리스트 V 가 얻어진다. 이 때 v_k (또는 v_0)와 v_{k-1} (또는 v_{N_v-1})이 VOP 경계에 위치하고 이 두 정점 사이를 연결하는 직선 구간이 VOP 경계에 위치하고 있으면, v_k (또는 v_0)를 1D 정점으로 선택한다. 더 나아가서, v_k (또는 v_{N_v-1})와 v_{k+1} (또는 v_0)의 두 정점들이 모두 1D 정점이면, v_k (또는 v_{N_v-1})는 V 에서 삭제되고 N_v 는 한 개 감소한다. 결과적으로 1D 정점이 연속해서 발생하지 않는다. 이 과정을 모든 정점에 대해서 순차적으로 수행한다. 1D 정점이 아닌 나머지 정점들은 2D 정점으로 정의한다. 그림 4(a)와 (b)에 1D 정점 선택의 예가 있다. IRM으로 얻어진 12개의 정점 중에서 VOP 경계에 위치하는 비초기 정점들은 v_4, v_5, v_6, v_7 등 4개이다. 위의 방법에 의해 선택된 1D 정점들은 이 중에서 v_5, v_6, v_7 등 3개이며, v_5, v_6 등 두 개의 ID 정점은 삭제된다. 이의 결과가 그림 4(b)이다. N_v 는 12개에서 10개로 줄어든다. 그림 4(b)에서 화살표는 1D 정점, v_5 의 이전 정점과의 상대 거리를 표시한다. 이와 같이 1D 정점과 2D 정점의 차이는 상대 거리가 각각 1차원과 2차원으로 정의된다는 점이다.

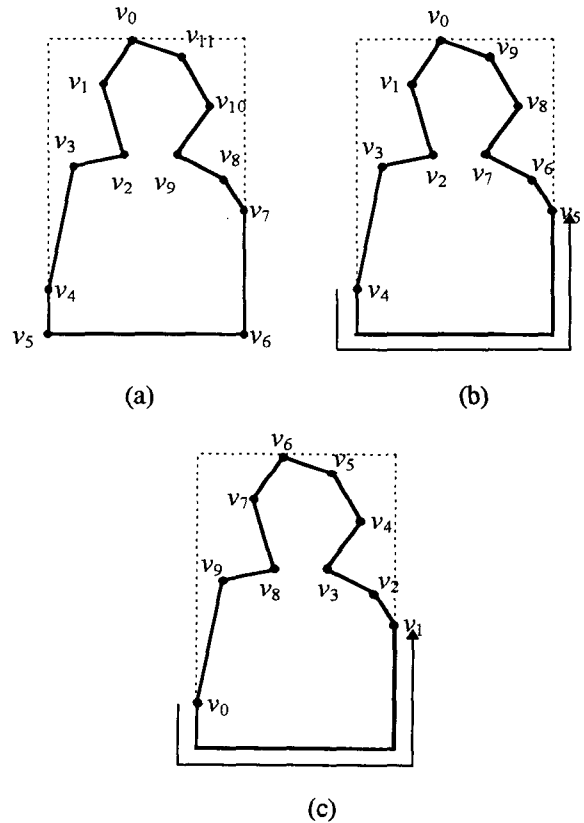


그림 4. 1D 정점 추출과 정점 재배열의 예. 점선 사각형과 화살표는 VOP 경계와 1D 정점의 상대 위치 거리를 각각 나타냄.
(a) IRM에 의해 추출된 정점들, $N_v=12$, (b) 1D 정점 (v_5), $N_v=10$, (c) (b)에서 v_4 가 가장 큰 상대 위치 거리를 보여줄 때의 정점 재배열의 결과.
Fig. 4. Example of 1D vertex selection and the vertex reordering where the dashed rectangle and the arrow are VOP boundary and the relative distance of 1D vertex, respectively, (a) firstly extracted vertices by IRM, $N_v=12$, (b) 1D vertex selection (v_5), $N_v=10$, (c) after vertex reordering when distance of v_4 in (b) is the largest.

나. 1차원 정점의 부호화

II장에서 설명한 OAVE 방식은 이전 정점과의 상대 거리를 전송하는데, 2차원 표현을 사용한다. 방향의 8진 표현, 상대 거리의 x-, y-방향 크기 등이 그것이다. 그러므로 상대 거리가 2차원으로 표현되는 2D 정점의 부호화에 OAVE 방식을 사용한다. 1D 정점도 이전 정점과의 상대 거리를 전송하는데, VOP 경계상의 1차원적인 거리를 전송한다(그림4(b), (c)에서의 화살표).

한편, MPEG-4에서는 모든 신호-이동(motion)정보, 모양정보, 색상(texture)정보-들을 VOP 단위로 부호화하며 [8], 이를 위해 VOP에 대한 부가 정보를 전송한다.

전송되는 VOP 부가 정보들은 VOP 경계의 높이(VOP_height), 너비(VOP_width), 상좌단의 위치 등이 있다[17]. 이미 수신단에서도 알고 있는 VOP 부가 정보를 1D 정점의 부호화에 이용한다.

기존의 OAVE와 제안된 1D 정점의 부호화 방식이 결합된 제안 정점기반 부호화 방식에서의 비초기 정점 부호화의 흐름도가 그림 5이다. 초기 정점을 절대 위치 부호화한 후, 2D 정점을 위한 OAVE 방식에서 비트 할당을 나타내는 두 지시자(indicator) $indicator_x, indicator_y$ 를 전송한다. 만일 초기 정점이 VOP경계에 위치하고 있으면, 다음 정점이 1D 정점인지를 확인하고 이에 관한 플래그(flag)를 1 비트 전송한다. 만일 2D 정점이면 OAVE의 부호화를 수행한다. 다음 정점이 1D 정점인 경우, 1D 정점의 1차원 상대 거리를 전송한다. 한 정점의 부호화가 끝나면 다시 VOP 경계에 위치하는가를 판단하고 같은 과정을 수행한다.

1D 정점의 상대 거리는 VOP 경계 상의 1차원 거리이다. 이 거리 정보는 방향(시계/반시계), 거리의 범위를 나타내는 정보, 그리고 거리 크기 등으로 구성된다. 방향 정보는 1 비트를 사용한다. 표 1에 거리 크기(r_{1d})와 소요되는 비트 수($indicator_{1d}$)의 관계를 나타내었다. 표 1에서의 8가지의 거리 범위는 3 비트로 고정 길이 부호화(fixed length coding; FLC)한다. 표 1에서 알 수 있듯이, r_{1d} 는 최대 255 화소까지 전송이 가능하다. 만일 255화소를 초과하는 경우는 256을 전송하고 나머지 ($r_{1d} - 255$)에 대하여 다시 $indicator_{1d}$ 를 구하고, 이를 이용하여 ($r_{1d} - 255$)의 부호화를 수행한다.

수신단에서는 이미 전송되어 알고 있는 VOP의 부가 정보를 이용하여, 복호화된 정점의 위치가 VOP 경계에 위치하고 있는가를 확인한다. 만일 VOP 경계에 위치한다면, 다음 1 비트를 가지고 다음 비트열로 OAVE의 복호화를 수행할 지, 1차원 정점의 복호화 할지를 결정한다.

제안된 1D 정점의 부호화는 2D 정점의 전송을 위한 OAVE 방식과 비슷한 구조를 가진다. 차이점으로는 먼저, 상대 거리의 표현이 1차원과 2차원인 차이가 있다. 예를 들어 상대 거리의 방향을 나타낼 때, OAVE 경우는 2차원의 8진 표현임에 반해, 1D 정점 부호화의 경우는 시계/반시계 방향으로 나타낸다. 두번째 차이점으로는, OAVE

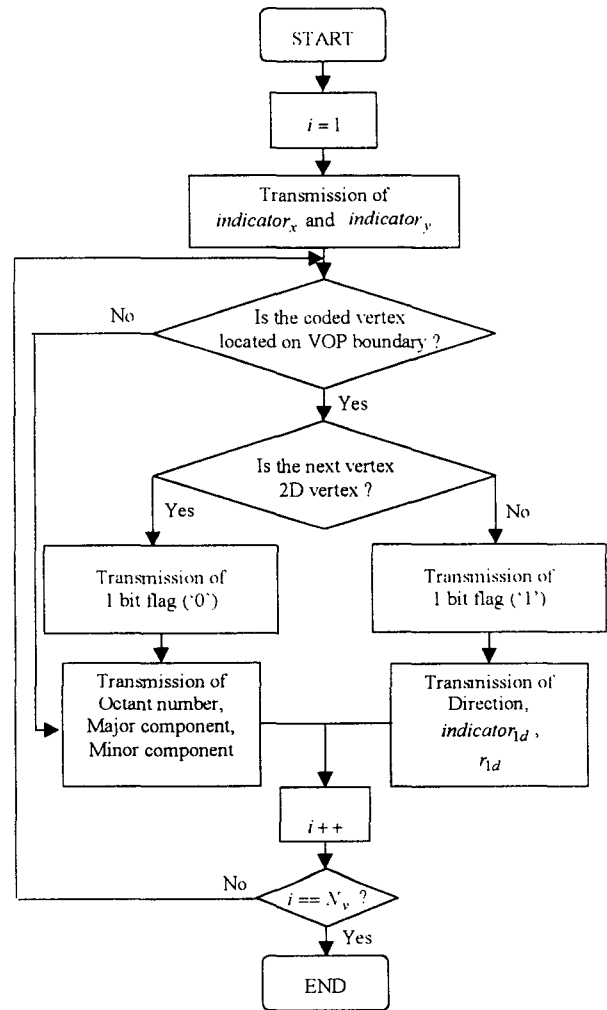


그림 5. 제안 정점기반 부호화 방식에서의 비초기 정점 부호화 방식의 흐름도.

Fig. 5. Flowchart of non-initial vertex coding in the proposed vertex-based coding method

방식에서의 주요소, 부요소에 대한 비트 할당 정보(그림 5의 $indicator_x, indicator_y$)를 윤곽선 당 한 번 전송하는 반면, 1D 정점 부호화에서는 1D 정점이 발생할 때마다 해당 $indicator_{1d}$ 의 정보를 전송한다. 만일 1D 정점이 많이 존재하는 경우, $indicator_{1d}$ 의 전송이 부호화 효율을 저하시킬 위험이 있다.

제안한 1D 정점의 선택 방법에서는 1D 정점과 이전 정점과의 상대 거리는 고려하지 않고 있다. VOP 경계 상의 상대 거리가 작은 경우, 이 거리의 전송을 위해 위의 1D 정점 부호화를 수행하는 것은 $indicator_{1d}$ 등의 부가 정보로 인하여 부호화 효율 측면에서 불리할 수 있다. 1D 정점과 이전 정점간의 거리가 주어진 임계치 이하면 1D 정점

으로 선택하지 않는 임계 수행을 할 경우, 더 높은 부호화 이득을 얻을 수 있지만, 본 논문에서는 고려하지 않는다.

2. 정점 재배열 방식

OAVE 방식에 있어서, 초기 정점의 절대 위치와 비초기 정점 각각의 상대 위치가 부호화되고 수신단에 전송된다. 다시 말해서 초기 정점(v_0)과 마지막 정점(v_{N_v-1}) 사이의 거리의 전송은 불필요하다. OAVE 방식에서는 각 정점 리스트에서의 최대 상대 거리가 클수록 각 정점에 대한 비트 할당이 커질 가능성이 있다. 그러므로 가장 긴 상대 거리가 v_0 와 (v_{N_v-1}) 사이의 거리가 되도록 정점을 재배열하면 각 정점의 비트 할당을 줄이고, 결과적으로 전체 발생 비트량을 줄일 가능성이 있다.

그림 4(b)의 예를 살펴보자. v_4 의 상대 거리가 가장 큰 경우의 재배열된 정점이 그림 4(c)이다. 1D 정점은 2D 정점과는 독립적으로 부호화되므로 정점 재배열은 1D 정점의 1D 상대 거리를 고려하지는 않는다. 그러나 그림 4(c)의 v_1 ($\leftarrow v_5$)처럼 1차원 정점 역시 2차원 정점과 같은 방법으로 재배열된다.

정점 재배열을 정리하면 다음과 같다:

M번째 정점이 2D 정점이고 2D 상대 거리가 가장 큰 경우, k번째($k=0, \dots, N_v-1$) 정점은 $((k+N_v-M) \bmod N_v)$ 번째 정점으로 재배열 된다.

이러한 정점 재배열을 위해서는 정점의 상대 거리의 정의가 필요하며, 다음과 같은 정의가 고려 가능하다.

$$dist^1(k) = (v_{k,x} - v_{k-1,x})^2 + (v_{k,y} - v_{k-1,y})^2, k=1, \dots, N_v-1 \quad (1)$$

$$dist^2(k) = |v_{k,x} - v_{k-1,x}| + |v_{k,y} - v_{k-1,y}|, k=1, \dots, N_v-1 \quad (2)$$

$$dist^3(k) = \max\{|v_{k,x} - v_{k-1,x}|, |v_{k,y} - v_{k-1,y}|\}, k=1, \dots, N_v-1 \quad (3)$$

위 식들에서 $v_{k,x}$ 와 $v_{k,y}$ 는 v_k 의 x-방향 크기와 y-방향 크기를 각각 나타내며, k=0일 경우에, k-1은 N_v-1 로 된다.

정점 재배열 방식은 IRM을 수행하는 경우에만, 다시 말해서 손실 부호화에서만 적용 가능하다. 무손실 부호화인 경우, 모든 정점(윤곽선 화소)에서 상대 거리가 동일하거나($dist^3(\cdot)$), 1 혹은 2($dist^1(\cdot)$ 과 $dist^2(\cdot)$) 값을 가지기 때문에 최대 값의 의미가 없다. 그리고 정점 재배열 방식은 2D 정점만 3개 이상이거나, 1D 정점이 존재하고 2D 정점이 2개 이상인 경우에 적용 가능하다.

IV. 모의 실험 결과 및 검토

실험 영상(test sequence)으로는 “Ad hoc group on MPEG-4 Binary Shape Coding” 이 배포한 Weather (QCIF), Children-Kids(SIF), 그리고 Robot(SIF) 등의 세 가지 MPEG-4 실험 영상들을 사용한다.

표 1. 1D 정점 부호화를 위한 r_{1d} 와 $indicator_{1d}$
Table 1. r_{1d} and $indicator_{1d}$ for 1D vertex encoding

$indicator_{1d}$	r_{1d}
1	$1 \leq r_{1d} \leq 2$
1	$3 \leq r_{1d} \leq 4$
2	$5 \leq r_{1d} \leq 8$
3	$9 \leq r_{1d} \leq 16$
4	$17 \leq r_{1d} \leq 32$
5	$33 \leq r_{1d} \leq 64$
6	$65 \leq r_{1d} \leq 128$
7	$129 \leq r_{1d} \leq 255$

표 2. 정점 재배열에 의한 VOP 당 평균 부호화 비트 수
Table 2. Average coding bits per VOP by the vertex reordering

d_{max}	1	2	3	5
<i>Weather</i>				
no reordering	361.04	245.53	213.14	187.24
$dist^1()$	334.15	230.04	202.24	179.51
$dist^2()$	334.39	230.25	202.41	179.65
$dist^3()$	334.15	230.04	202.24	179.54
<i>Children-Kids</i>				
no reordering	1468.30	982.30	786.89	566.71
$dist^1()$	1431.87	961.84	763.72	556.77
$dist^2()$	1438.55	963.63	766.22	557.28
$dist^3()$	1431.18	961.29	762.05	556.10
<i>Robot</i>				
no reordering	1729.27	1220.93	991.72	769.06
$dist^1()$	1782.72	1256.77	1014.75	780.70
$dist^2()$	1784.93	1257.15	1014.81	780.84
$dist^3()$	1783.75	1258.11	1015.06	780.88

사용된 제어 파라미터 d_{max} 값들은 0(무손실 부호화), 1, 2, 3, 5 화소 등이다. 윤곽선 추출을 위한 전처리 과정은 참고문헌 [10]의 방식이 사용되었다. 그림 6은 각각의 실험 영상들의 첫번째 이진 마스크와 VOP의 윤곽선, 그리고 VOP 경계(오른쪽 그림 내의 점선 직사각형) 등을 보여준다. 그림에서 알 수 있듯이, 실험 영상 Weather와

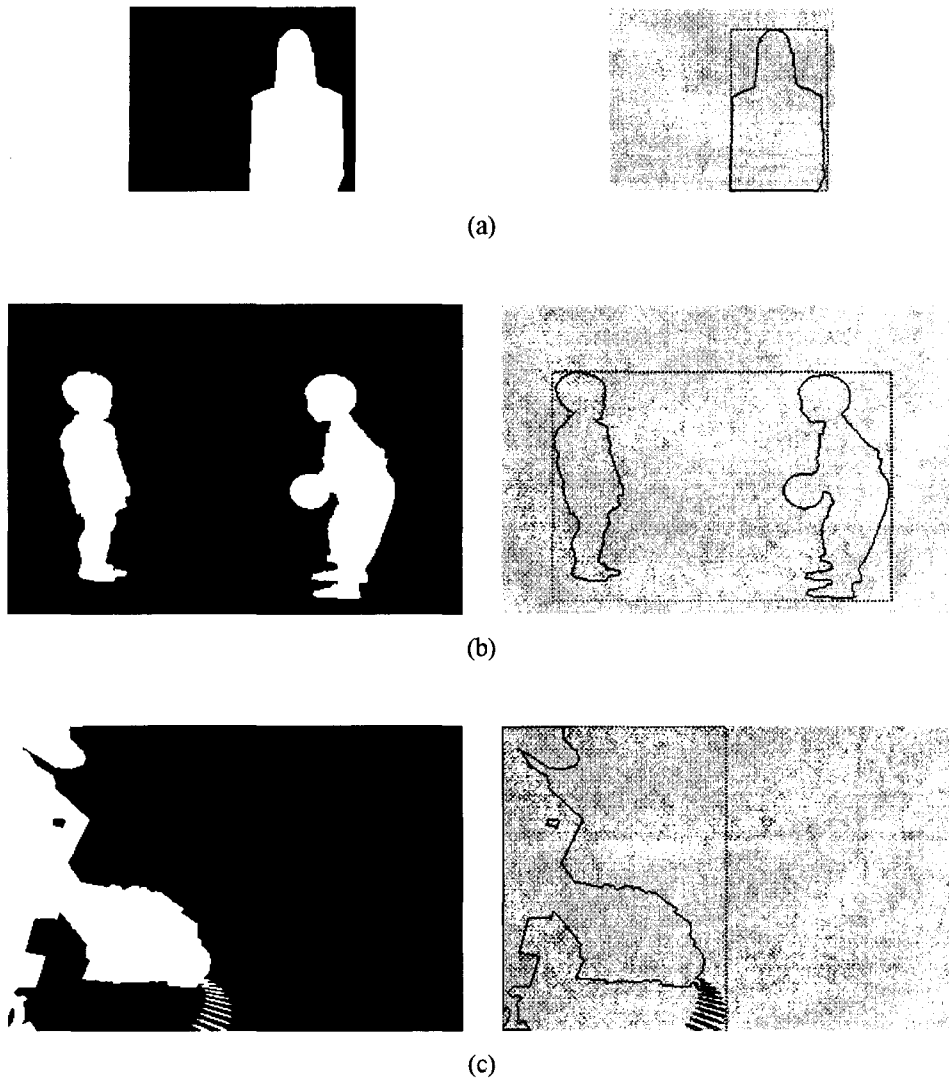


그림 6. 실험 영상들의 첫번째 VOP의 이진 마스크와 윤곽선. (a) *Weather* (b) *Children-Kids* (c) *Robot*
 Fig. 6. Binary mask and contour of the first VOP of the test sequences. (a) *Weather* (b) *Children-Kids* (c) *Robot*

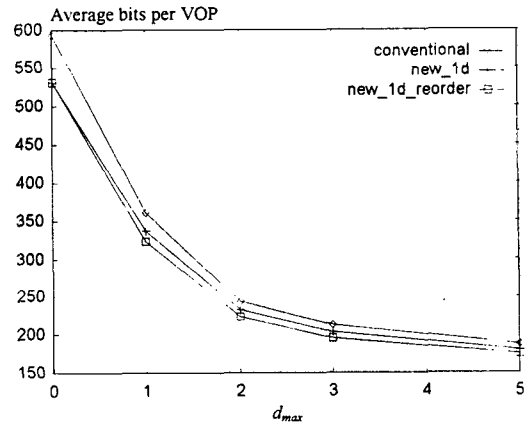
*Robot*의 경우 VOP 경계에 위치하는 윤곽선 화소가 상당히 많은 반면, 실험 영상 *Children-Kids*의 경우는 VOP 경계 상의 윤곽선 화소가 거의 없다.

앞에서 언급했듯이, 1D 정점의 사용과 정점 재배열 방식은 재현 오차에는 영향을 미치지 않으며, 발생하는 부호화 비트 수에만 관련된다. 그러므로 제안 방식들의 성능 분석을 위하여 본 실험에서는 평균 부호화 비트 수만 비교한다. 한편 제안된 방식들은 비초기 정점에 대한 부호화에만 영향을 미치므로, 제안 방식들의 영향을 보다 정확하게 고찰하기 위해서 N_v 등의 부가 정보와 초기 정점 전송을 위한 비트 수를 포함하지 않는다. 표 2는 정점 재배열과 관련된 결과들이다. 이 결과들은 1D 정점을 사용하지 않는 결과들이다. 정점 재배열을 수행하지 않는 기존 정점기

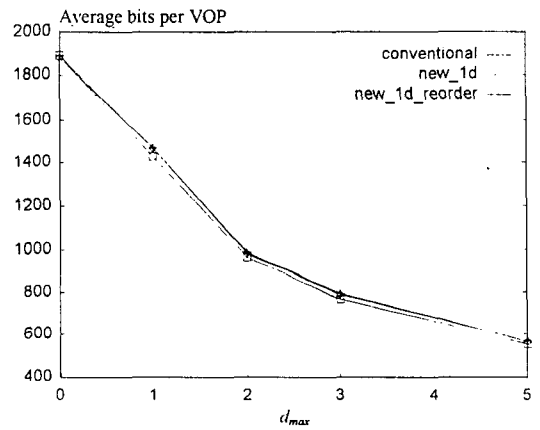
반 부호화 방식의 결과(*no reordering*)와 상대 위치 거리의 정의를 식(1), 식(2), 식(3)을 각각 사용하여 정점 재배열을 수행한 결과들을 비교하였다. $dist^1(\cdot)$ 과 $dist^3(\cdot)$ 의 결과들이 $dist^2(\cdot)$ 의 결과보다 약간 우수하다. 그러나 복잡도 측면에서 $dist^3(\cdot)$ 가 $dist^1(\cdot)$ 보다 수행이 간단하며 OAVE 방식에서도 상대 거리, 즉 $|v_{k,x} - v_{k-1,x}|$ 와 $|v_{k,y} - v_{k-1,y}|$ 를 구해야 한다. 그러므로 $dist^3(\cdot)$ 를 사용하는 정점 재배열에 의한 복잡도 증가는 미미하다.

실험 영상 *Weather*와 *Children-Kids* 영상에서는 약 5%까지 부호화 비트가 감소함을 알 수 있다. 그러나 *Robot* 영상의 경우, 정점 재배열에 의해 부호화 비트 수가 증가함을 볼 수 있다. 이는 OAVE의 특성에 의한 것으로, $dist^3(\cdot)$ 이 작아진다고 해서 주요소와 부요소의 비트 할당이 항상 작아

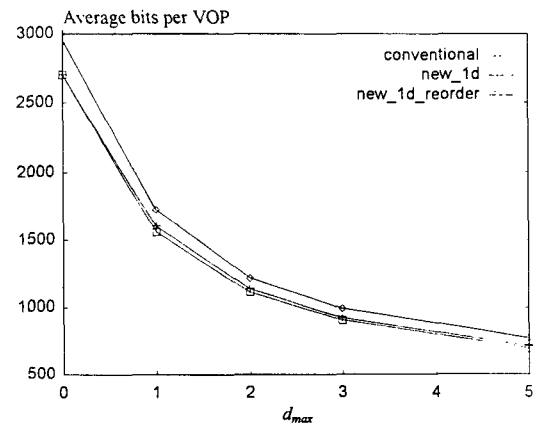
지지 않음을 보인다. 예를 들어 $dist^3(\cdot)$ 가 큰 순서대로 (0, 120), (80, 100), (30, 50) 등의 상대 위치를 가지는 정점들이 추출된 경우를 생각해 보자. 재배열을 수행 안 했을 때 (0, 120), (30, 50) 들을 전송해야 한다면, OAVE 방식에서는 정점 당 12 비트(x-방향, y-방향 각각 5 비트와 7 비트)가 필요하다. $dist^3(\cdot)$ 에 의해 정점 재배열을 수행하면 (80, 100), (30, 50) 등을 전송하는데, 이 때 정점 당 14 비트(x-방향, y-방향 각각 7 비트씩)가 필요하다. 그러므로 정점 재배열을 한 경우에 더 많은 부호화 비트가 발생한다. 이러한 결과들로부터, 정점 재배열의 부호화 이득은 추출된 정점 위치의 특성에 의존함을 알 수 있다. 그림 7에서는 d_{max} 의 변화에 따른 VOP 당 평균 부호화 비트 수를 나타내었다. 비교 대상은 기존 정점기반 부호화 방식(conventional), 정점 재배열을 수행하지 않고 1D 정점만을 이용한 방법(new_1d), 1D 정점과 정점 재배열($dist^3(\cdot)$)을 함께 이용한 방법(new_1d_reorder) 등 세 가지이다. 무손실 부호화에서는 정점 재배열을 수행하지 않기 때문에 new_1d와 new_1d_reorder의 결과가 같다. 실험 영상 *Weather*와 *Robot*의 경우 1D 정점의 사용으로 인한 부호화 이득이 높다. 이는 두 영상의 경우, VOP 경계에 윤곽선 화소가 많이 존재하기 때문이며, 이러한 특성은 그림 6에서 확인할 수 있다. 반면, *Children-Kids*의 경우, 1D 정점에 의한 성능 향상이 거의 없다. new_1d_reorder의 경우, 모든 영상에서 이득이 있음을 알 수 있다. 주목할 점은 실험 영상 *Robot*에서 1D 정점을 사용한 경우, 정점 재배열에 의한 이득이 있다는 점이다. 이는 1D 정점의 상대 거리를 고려하지 않고 정점 재배열을 수행하는 경우, 위에서 언급한 경우가 감소함을 보여준다. 정리하면, 제안된 1D 정점의 선택 및 부호화는 입력 영상의 윤곽선 화소가 VOP 경계 상에 위치하는 정도에 부호화 이득이 의존하며, 많은 윤곽선 화소가 VOP 경계 상에 위치할수록 부호화 이득이 커진다. 또한 정점 재배열에 의한 이득을 크게 한다. 제안된 방식의 사용으로 실험 영상에서는 같은 d_{max} 하에서 최대 12%의 부호화 비트 수 감소를 가져올 수 있다. 이상의 1D 정점의 선택 및 부호화에 관련한 모의 실험 결과는 정점기반 부호화 방식이 저지연 부호화(low delay coding)와 error-resilient 부호화에 적용될 수 있는 가능성을 보인다. 정점기반 부호화 방식에서는 적은 수의 정점들이 물체 윤곽선의 특성을 표현하기 때문에 부호화 효율이 증가한다. 그러나 저지연 부호화나 error-resilient 부호화에서는 VOP 단위 대신 매크로 블럭(macroblock)이나 슬라이스(slice) 단위의 부호화 수행이 고려되므로, 정점기반 부호화는 부호화 효율 측면에서 불리하다. 그러나 각 단위로 입력 이진 마스크를 분할하고 윤곽선을 추출시, 많은 윤곽선 화소들이 매크로 블럭 또는 슬라이스의 직사각형 경계에 많이 존재하게 되며, 수신단에서도 직사각형 경계의 정보를 알고 있다. 그러므로 1D 정점의 사용이 가능하며, 이는 1D 정점을 사용하지 않은 경우에 비해서 부호화 효율을 상당히 높일 것으로 고려된다.



(a)



(b)



(c)

그림 7. 기존 방식과 제안 방식의 VOP 당 평균 부호화 비트 수.

(a) *Weather* (b) *Children-Kids* (c) *Robot*

Fig. 7. Average coding bits per VOP of the conventional vertex-based coder and the proposed one. (a) *Weather* (b) *Children-Kids* (c) *Robot*

V. 결론

본 논문에서는 1차원 정점과 정점 재배열을 사용하는 새로운 정점기반 모양정보 부호화 방식을 제안하였다.

1차원 정점의 선택과 1차원 정점 부호화 방식은 윤곽선 화소가 VOP 경계에 많이 위치할수록 부호화 이득의 증가를 가져온다. 한편 2차원 정점을 위한 정점 재배열 방식은 최대 상대 거리가 초기 정점과 마지막 비초기 정점 사이에 위치하도록 정점의 순서를 재배열함으로써 부호화 이득을 가져온다. 특히 1차원 정점을 사용할 경우, 정점 재배열 방식으로 인한 이득이 크다. 전체 윤곽선 화소를 정점으로 정의하는 무손실 부호화의 경우에는 1차원 정점의 선택 및 부호화만 적용되고, 정점 재배열을 사용하지 않는다.

제안된 정점기반 부호화 방식이, 물체 적응형 정점 부호화 방식만을 사용하는 기존의 방식에 비하여 최대 12%의 부호화 비트 수를 감소시킴을 모의 실험을 통해 확인하였다. 한편 1차원 정점의 선택 및 부호화가, 정점 기반 부호화 방식이 저지연 부호화와 error-resilient 부호화에 적용될 수 있는 가능성을 보임을 설명하였다.

향후 연구 방향으로는 1차원 정점 선택할 때, VOP 사각 경계의 윤곽선 화소 길이를 고려하는 적응형 선택 방식 등이 있다.

참고 문헌

- [1] N. Ansari and E. J. Delp, "On detecting dominant points," *Pattern Recognition*, vol. 24, pp. 441-451, 1991.
- [2] J. G. Dunham, "Optimum uniform piecewise linear approximation of planar curves," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, pp. 67-75, Jan. 1986.
- [3] F. S. Hill Jr, *Computer Graphics*, Macmillan Publishing Company, New York, NY., 1990.
- [4] C. h. Teh and R. T. Chin, "On the detection of dominant points on digital curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, pp. 859-872, Aug. 1989.
- [5] P. Gerken, "Object-based analysis-synthesis coding of image sequences at very low bit rates," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 4, pp. 228-235, Jun. 1994.
- [6] M. Hoetter, "Object-oriented analysis-synthesis coding based on moving two-dimensional objects," *Signal Processing: Image Communication*, vol. 2, no. 4, pp. 409-428, Dec. 1990.
- [7] J. Chung, J. Moon, and J. Kim, "Shape information reduction based on contour prediction and shape coding type," in ISO/IEC JTC1/SC29 /WG11 MPEG95/0461, Dallas, TX., Nov. 1995.
- [8] T. Ebrahimi, "MPEG-4 video verification model: A video encoding/decoding algorithm based on content application," *Signal Processing: Image Communication*, vol. 9, no. 4, pp. 367-384, May. 1997.
- [9] R. Koenen, F. Pereira, and L. Chiariglione, "MPEG-4: Context and objectives," *Signal Processing: Image Communication*, vol. 9, no. 4, pp. 295-304, May. 1997.
- [10] MPEG Video Group, "Description of core experiments on shape coding in MPEG-4 video," in ISO/IEC JTC1/SC29/WG11 N1326, Tampere, Jul. 1996.
- [11] J. Ostermann, "Report on the Ad Hoc Group on evaluation of tools for non tested functionalities of video submissions for MPEG-4," in ISO/IEC JTC1/SC29/WG11 MPEG95/0488, Dallas, TX., Nov. 1995.
- [12] K. O'Connell, "Object-adaptive vertex-based shape coding method," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, pp. 251-255, Feb. 1997.
- [13] J. H. Kim, J. L. Kim, C. S. Park, H. M. Jung, and H.-S. Kim, "Contour coding using DST(discrete sine transform)," in *Proc. International Workshop on Coding Techniques for Very Low Bit-rate Video*, Tokyo, paper no. C-1, Nov. 1995.
- [14] C. Jordan and T. Ebrahimi, "Scalable vertex-based shape coding-S4h results," in ISO/IEC JTC1/SC29/WG11 MPEG97/2034, Bristol, Apr. 1997.
- [15] K. O. Connell, P. Gerken, C. L. Buhan, and J. Kim, "Error resilient vertex-based(S4h) shape coding description," in ISO/IEC JTC1/SC29/WG11 MPEG97 /1963, Bristol, Apr. 1997.
- [16] J. Lee, J. Chung, S. Lee, J. Moon, and J. Kim, "Complexity and robustness of vertex based shape coding," *Proc. MoMuC'97*, Seoul, Sept. 1997.
- [17] MPEG Video Group, "MPEG-4 Video Verification Model Version 7.0," in ISO/IEC JTC1/SC29/WG11 N1642, Bristol, Apr. 1997.

저 자 소 개



정 재 원

1966년 8월 22일생
 1989년 2월 한양대학교 전자통신공학과 졸업(공학사)
 1991년 8월 한국과학기술원 전기및전자공학과 졸업(공학석사)
 1998년 2월 한국과학기술원 전기및전자공학과 졸업(공학박사)
 1995년 6월 ~ 현 재 현대전자산업주식회사



문 주 회

1985년 2월 서강대학교 전자공학과 졸업(공학사)
 1987년 2월 한국과학기술원 전기및전자공학과 졸업(공학석사)
 1992년 8월 한국과학기술원 전기및전자공학과 졸업(공학박사)
 1992년 9월 ~ 1994년 2월 한국과학기술원 정보전자연구소 선임연구원(Post-Doc)
 1994년 3월 ~ 현 재 현대전자산업주식회사 책임연구원
 주관심분야 : 영상 처리, 영상 부호화, 정보 이론

김 재 군

1938년 9월 17일생
 1962년 2월 한국항공대학 응용전자과 졸업(공학사)
 1967년 2월 서울대학교 전자공학과 졸업(공학석사)
 1971년 8월 미국 University of South California 전자공학과 졸업(공학박사)
 1972년 4월 ~ 1973년 3월 NASA GSFC 연구원
 1993년 1월 ~ 1994년 12월 한국통신학회 학회장
 1973년 4월 ~ 현재 한국과학기술원 전기및전자공학과 교수
 1993년 9월 ~ 현재 한국과학기술원 멀티미디어 통신 공동연구 센터장
 주관심분야 : Video Coding, Visual Multimedia Communication