

# Hilbert 곡선 Scan 방법을 이용한 의학 영상의 압축 방법에 관한 연구

지 영 준 · 박 광 석\*

= Abstract =

## A Study on the Improvement of Compression Method Using Hilbert Curve Scanning for the Medical Images

Young Joon Chee, Kwang Suk Park\*

For efficient storage and transmission of medical images, the requirement of image compression is increasing. Because differences between reconstructed images and original images are related with errors in the diagnosis, lossless compression is generally preferred in medical images. In Run Length Coding which is one of the lossless compression method, we have applied modified scanning direction based on the Hilbert curve, which is a kind of space filling curve. We have substituted the traditional raster scanning by Hilbert curve direction scanning. Using this method, we have studied enhancement of compression efficiency for medical images.

### 1. 서 론

병원에서 의학 영상의 전산화는 여러 분야에서 광범위하게 진행되고 있다. 따라서 디지털 영상의 저장이나 전송이 점차 필수적으로 되어가고 있으며 효과적인 영상 전송과 저장을 위해서는 영상의 압축이 또한 절실히 필요하다. 의학 영상의 압축은 진단에 관계되는 것이기 때문에 다른 영상과는 달리 진단에서의 오류가 발생하는 것을 막기 위하여 정보를 완전히 복원해 내야 한다. 따라서 영상의

화소 값을 완전히 원래의 것과 같이 복원해내기 위해서는 DPCM(Difference Pulses Code Modulation)이나 RLC(Run Length Coding)같은 무손실 압축 방법이 비록 압축비는 2:1 내지 3:1 정도밖에 되지 않지만 현실적으로 널리 사용되고 있다. 또 완전히 같은 값으로 복원하지는 못하지만 진단이 가능한 정도의 압축은 DCT(Discrete Cosine Transform)등의 방법을 사용하여 압축비 10:1 정도를 달성할 수 있다. 벡터 양자화기나 운동 검출을 이용하는등의 최근 압축방법들은 고 화질 보다는 압축 효율을 높이는 것에 관심이 있기 때문에 의학 영상에 직접적인 적용에는 문제가 있다. 특히 이와 같은 고압축 효율의 방법들은 의학 진단의 유용성이 보고되었음에도 불구하고 원천적으로 포함되어 있는 복원된 영상과 원래 영상과의 오차때문에 의학 영상의 미세한 부분의 검출을 못하는

(접수 : 1992년 11월 4일)

서울대학교 대학원 핵동과정 의용생체공학

\* 서울대학교 의과대학 의공학교실

(본 연구는 1990년도 서울 대학교 병원 특진 연구비 지원으로 이루어졌음)

것을 우려하여 현실적으로 기피되고 있다.[1]

본 논문에서는 RLC 방법을 이용한 무손실 영상 압축법을 변형하여 그 효율을 높인 방법을 소개한다. RLC 방법을 적용하기에 앞서, 영상 정보를 2차원적인 성질을 반영하도록 재 배치 시켜놓고 기존의 1차원적인 RLC 방법으로 처리함으로써 압축 효율의 향상을 꾀하였다. 이는 시스템의 성능을 높이기 위한 전처리(Pre-Processing)의 의미를 가진다. 어떠한 스캐닝 방향을 선택하여 화소를 재배치하느냐 하는 것은 사용하는 영상의 종류에 따라 그 효율이 달라진다. 본 논문에서는 최근 관심의 대상이 되고 있는 fractal 이론과 관련하여 연구되는 Hilbert 곡선을 선택하여 이를 RLC 방법으로의 확영상에 적용하여 보았다.

RLC는 팩시밀리, 일기도, 설계도등의 1비트 영상을 압축하는데 주로 사용된다. 이러한 경우는 영상 전체 화소의 값이 0 또는 1이므로 0에서 1로, 1에서 0으로 변하는 확률로서 성능을 계산할 수 있다[2],[6] 또 여러 비트 영상도 주변의 화소값이 같은 경우가 많으므로 RLC를 이용하여 영상을 압축할 수 있다. 본 논문의 관심은 RLC를 할때 다음 화소를 선택하는 방법에 관한 것이다. 기존의 TV 주사선 방향으로 진행하는(Raster Scan)방법이 아니라, 일정한 규칙을 갖고 2차원 평면을 채워가는 "공간 채우기 곡선"중 Hilbert 곡선을 이용하여 스캔하는 방법으로 RLC하여 그 실험 결과를 정리하였다. Hilbert 곡선은 II 절에서 설명하겠지만 기존의 스캔 방법이 영상의 가로 방향의 중복성(redundancy)만을 이용한 것에 반해 가로 세로 방향으로 인접한 화소를 모아서 RLC하므로 1차원적으로 처리를 하면서도 2차원적인 화소의 중복성을 더욱 잘 이용한다고 볼 수 있다. II 절에서는 RLC와 Hilbert 곡선에 대한 설명을 하고 III 절에서는 실제로 변형된 RLC로 1비트 영상과 의학 영상에 적용한 실험 결과를 보였다.

## II. RLC 와 Hilbert 곡선

### 1. RLC 분석

RLC(Run Length Coding)는 같은 화소값이 일정한 길이로 지속될 때, 일일이 화소값을 나타내는 대신 같이 지속되는 위치와 지속되는 길이로 표현함으로써 영상의 데이터 양을 줄이는 방법이다. 여

러 비트 영상에 대해서는 같은 값이 지속되는 경우가 만족할 만한 압축율을 달성할 정도로 많지 않기 때문에 자주 사용되지는 않는다. 하지만 원래의 영상을 완전히 복원해 낼 수 있다는 장점을 갖고 있다. 1 비트 영상에 대해서는 같은 값이 반복되는 경우가 많으므로 자주 사용되고 상황에 따라서 몇가지 개선된 방법도 사용된다. 1 비트 영상인 경우의 RLC는 1차 Markov 모델로 Black 과 White 의 변화하는 확률 분석을 통하여 압축 효율을 계산할 수 있다. 즉,

$P(B | W) = q_0$  : White 에서 Black 으로 변화(Transition)하는 확률.

$P(B | W) = q_1$  : Black 에서 White로 변화(Transition)하는 확률.

이라 하면,

$P(W | W) = 1 - q_0$  : White 에서 White로 지속되는 확률.

$P(B | B) = 1 - q_1$  : Black 에서 Black으로 지속되는 확률.

이 된다. 이때  $q_0$  와  $q_1$  으로부터,

$$V_w = \frac{1}{q_0} : \text{Average White Run Length}$$

$$V_b = \frac{1}{q_1} : \text{Average Black Run Length}$$

이 된다. 즉 변화하는 확률이 작을 수록 같은 값이 지속되는 평균 길이가 길어지고 따라서 압축 효율은 높아진다. 이렇게 만들어진 Black 의 길이와 White 길이의 테이블을 각각 Huffman coding 함으로서 압축을 끝마치는데 이때의 평균 화소당 소요되는 비트 수인 Entropy 는 다음과 같이 구해진다 [2].

$$H_w = -q_0 \cdot \log q_0 - (1 - q_0) \cdot \log(1 - q_0)$$

$$H_b = -q_1 \cdot \log q_1 - (1 - q_1) \cdot \log(1 - q_1)$$

$$H_{wb} = \frac{V_w}{V_w + V_b} H_w + \frac{V_b}{V_w + V_b} H_b$$

여러 비트로 된 영상에 대해서는 위와 같이 간단하게 분석할 수는 없다. 주어진 영상의 gray level 값이 여러가지이므로 변화하는 경우의 수가 여러가지이고, 또 이에 따른 각각의 같은 값이 여러가지이므로 복잡한 식이 된다. 경우에 따라 같은 값이 지속되는 확률이 작을 경우는 오히려 값을 표시하고 길이를 표시하는 RLC 방법이 원래 영상의 데이터 양보다 많아질 수도 있다.

본 논문에서는 기본적인 RLC 방법에 약간의 변

화를 주었다. 다음 화소 값에서 이번 화소 값을 빼면 그 차는 0부터 255의 범위에 있던 값이 -255부터 +255사이의 값으로 되지만 실제의 경우는 대부분 -32부터 +32사이에 분포하게 된다. 이러한 처리는 복원과정에서 완전히 복원할 수 있다. 이와 같이 인접된 화소값의 차이를 구하면 원래의 화소값이 같은 값이 지속되는 동안에는 0이 된다. 그래서 0값이 3개 이상 지속될 때에만 <표식, 길이>로 코딩하고 그렇지 않은 경우에는 원래의 화소 값(Gray Level)을 그대로 쓰는 것이다. 이러한 방법으로 코딩하면 RLC로 인하여 손해보는 영역은 없게 된다. 예를 들어 {11,13,14,14,8,8,8,8,7,7,7,6,1}의 16개의 숫자를 인접한 다음 값에서 빼면 {11,2,1,0,-6,0,0,0,0,-1,0,0,-1,-6,}이 된다. (첫 번째수는 초기값으로 보존해야 한다.) 이 숫자열을 코딩할 때, 표식을 15로 사용하면, {11,2,1,0,-6,15,-1,15,3,-1,-6}으로 코딩하게 된다. (Run Zero Coding) [1].

## 2. Hilbert 곡선

RLC를 함에 있어서 2차원 평면상에 있는 화소들을 어떠한 방향으로 모두 경유하여 진행해 가는 가 하는 것이 이 논문의 관심사이다. 보통의 경우라면 <그림1>과 같이 Raster 스캔이라고 불리는 TV 주사선 방향으로 진행해 나가는 것이다. 이는 가로 방향으로 인접한 화소끼리 같은 값을 갖을 확률이 높으므로 어느 정도의 효율을 달성할 수 있다. 하지만 세로 방향으로도 인접한 화소끼리 비슷한 값을 갖는 성질은 이용하지 못하는 문제가 있다. 2차원 평면상의 정보를 인코딩할 때 Scan하는 방법에 따라 압축효율을 올릴 수 있다. <그림2>와 같은 방향으로 진행한다면 2차원적인 상관도(Correlation)를 이용한 것이 되어 더 나은 결과를 낼 수 있으며, 그 향상의 정도는 각 영상의 특성에 따라 결정된다. [4]

<그림2>는 Hilbert 곡선으로 불리는데 일정한 규칙으로 그 모양이 반복됨을 알 수 있다. Hilbert 곡선은 일정한 규칙으로 모양을 키워나가며 2차원 평면을 빠짐 없이 채워나가는 “공간 채우기 곡선(Space Filling Curve)”의 일종이다.[3] 공간 채우기 곡선은  $f: [0,1] \rightarrow A, A \subset R^2$ 의 형태를 갖는 함수로, 연속 공간의 경우 이를 무한히 반복하면 2차원 상의 어떤 도형의 내부를 완전히 채우게 된다.[5]

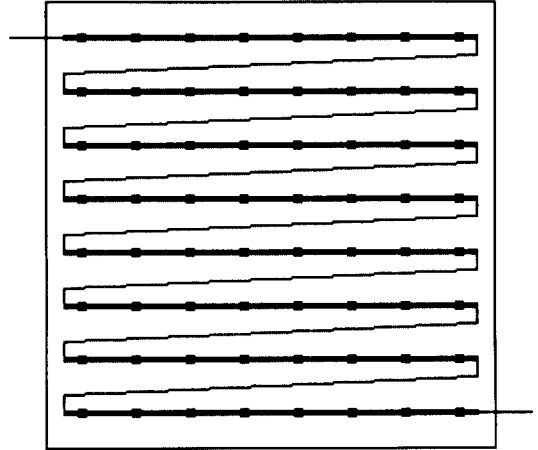


그림 1 Raster 스캐닝

Fig. 1 Raser Scanning

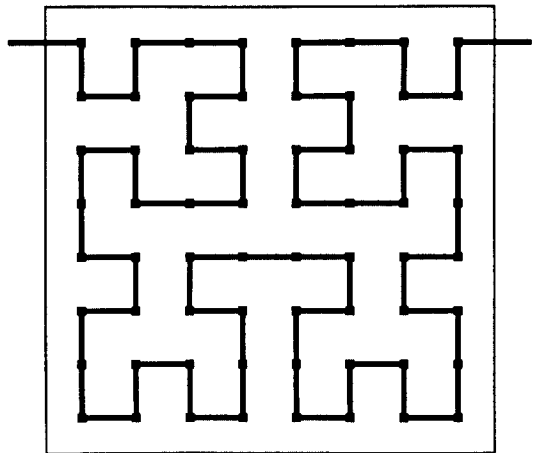
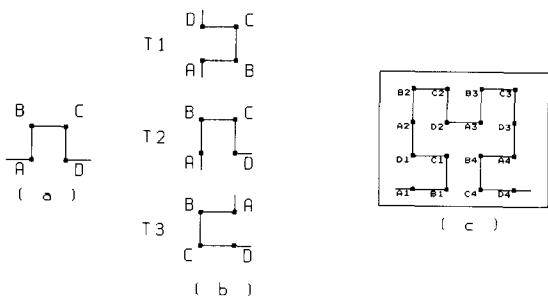


그림 2 힐버트 곡선 스캐닝

Fig. 2 Hilber Curve Scanning

이를 디지털 영상 신호같은 이산 공간에 대해 적용시켜 Hilbert 곡선을 만들 수 있다.<그림 3(a)>와 같은  $2 \times 2$ 의 원시 무늬(primitive pattern)를 일정한 규칙에 따라 가로 방향과 세로 방향으로 각각 두 배씩 확장시킬 수 있다.  $2 \times 2$ 의 원시 무늬를 <그림 3(b)>에서의 T1의 변환으로 만들어 좌하귀에 배치시키고, T2의 변환으로 만들어 좌상귀와 우상귀에 배치시키고, T3의 변환으로 만들어 우하귀에 배치시킴으로서 이 점들을 이어나가면  $4 \times 4$ 의 모든 점을 경유하는 곡선을 만들 수 있다.

이러한 과정을 반복하면  $2^n \times 2^n$  의 디지털 영상을 Hilbert 곡선에 따라 스캔할 수 있다.



**그림 3** 힐버트 곡선 만들기

- (a) 원시 패턴 (b) 세가지 변환 (c) 두 배 확장
- (a) Primitive Pattern (b) Three Transform
- (c) Double Expansion

### III. 실험과 결과

#### 1. Hilbert 곡선을 이용한 RLC 테스트

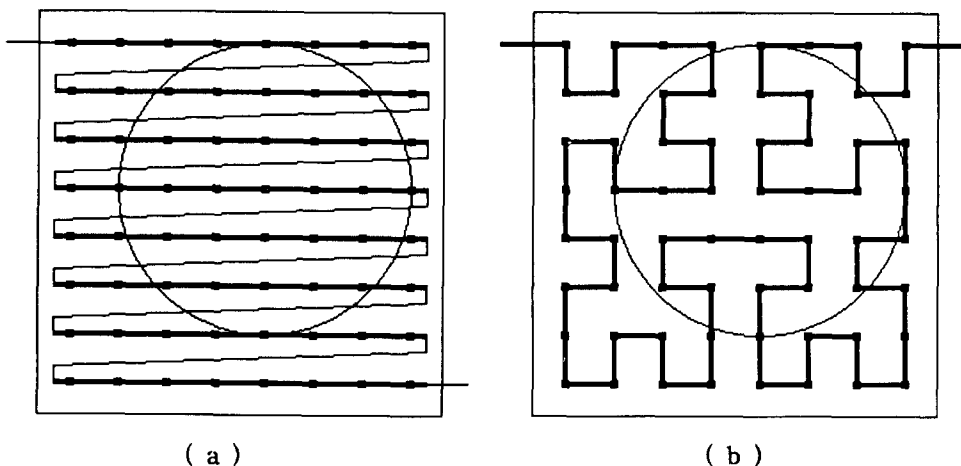
Hilbert 곡선 이외에도 여러 가지 공간 채우기 곡선을 생각할 수 있고 기존의 Raster스캔도 2차원

공간을 채우는 “공간 채우기 곡선”의 일종으로 생각할 수 있는데, 문제는 어떠한 곡선이 영상의 대상의 기하학적인 성질을 잘 반영하여 앞절에서 말한 평균 지속 길이 ( $V_w, V_b$ )를 길게 하느냐하는 것이다. 이는 영상 따라 다를 수도 있으나 보통의 경우를 생각해보면 가로 세로 방향의 성질을 다 이용한다는 면에서 Raster 스캔 보다는 Hilbert 곡선으로의 스캔이 나음을 볼 수 있다. <그림4>는  $8 \times 8$ 의 그리드(grid)에 반지름이 3인 원을 위치시켰을 때, 0과1이변하는 횟수를 세어보면(a)의 Raster Scan의 경우는 14번의 변화가 생기지만 (b)의 Hilbert Scan의 경우는 10번의 변화가 생김을 알 수 있다. 이로써 원에 대해서는 RLC 할 때 Hilbert 스캔이 Raster 스캔보다 더 나은 효율을 나타냄을 알 수 있다.

이를 더 체계적으로 검사해 보기 위해서 간단한 시험용 영상을 만들어 실험해 보았다. 모든 값이 0인 영상위에 원의 중심으로 쓰일 ( $X_{ci}, Y_{ci}$ )를 임의로 여러개 선정하여, 반지름이 R인 원을 그리고 그 내부는 1로 채운다.

$$f(X,Y) = \begin{cases} 1, & (X-X_{ci})^2 + (Y-Y_{ci})^2 < R^2 \\ 0, & \text{elsewhere} \end{cases}$$

( $X_{ci}, Y_{ci}$ ) : i번째 원의 중심, R : 원의 반지름.



**그림 4** 래스터 스캐닝과 힐버트 스캐닝의 비교

- (a) 래스터 스캐닝 : 14번의 변화
- (b) 힐버트 스캐닝 : 10번의 변화

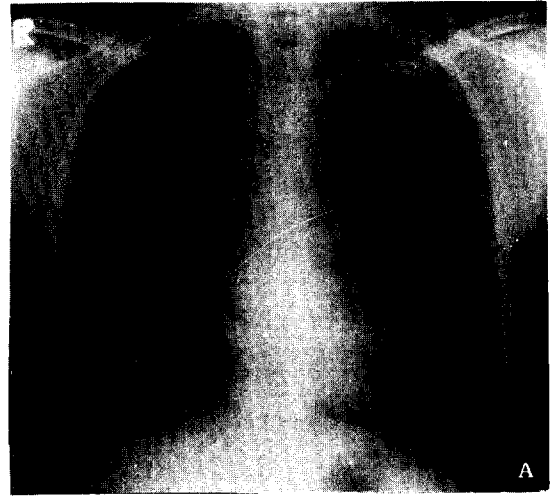
**Fig. 4** Raster Scanning vs, Hilbert Scanning

- (a) Raster Scanning : 14 changes
- (b) Hilbert Scanning : 10 changes

이와 같은 영상을 Raster-Scan 방향과 Hilbert 방향에 대하여 각각,  $q_0, q_1$  을 구하여 한 값이 지속되는 길이의 평균을 다음 표 1에 나타내었다. 표에서 보듯이 Hilbert 스캔이 Raster 스캔보다  $q_0, q_1$ 이 더 작아, 0이나 1의 한 값이 지속되는 평균 길이가 더 길어짐을 알 수 있다.

**표 1** 반지름에 따른 한 값이 지속되는 평균 길이  
**Table 1** The Average Length for circles with various radius

	Raster Scanning	Hilbert Scanning
3	1356.36	1627.63
5	530.56	548.86
8	240.32	248.33
10	117.08	123.83
13	173.42	175.68
15	277.61	329.26
18	193.37	259.04



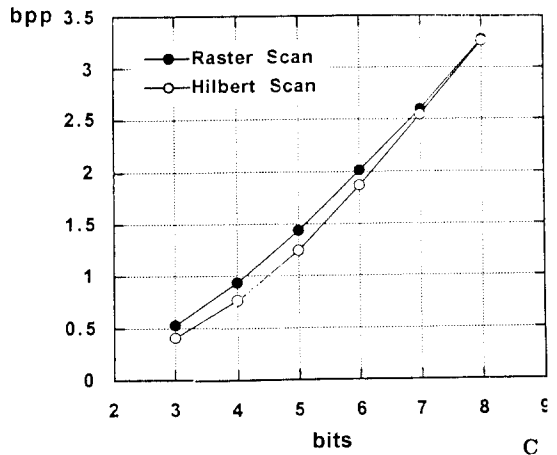
	Raster Scan	Hilbert Scan
8 bit	3.2756	3.2617
7 bit	2.5963	2.5449
6 bit	2.0167	1.8713
5 bit	1.4387	1.2453
4 bit	0.9335	0.7665
3 bit	0.5276	0.4071

**2. RLC 의 1 비트 영상의 적용**

이 방법의 일반적 영상의 압축 효과를 실험하기 위하여, 신문에서 글씨와 그림이 섞여 있는 광고 부분을  $512 \times 512 \times 1$  비트의 영상으로 만들어 앞 절에서 분석한 것과 같이 분석한 결과를 <표2>에 보였다. 두 번째 줄은 기존의 방법인 Raster 스캔으로 RLC 한 결과이고 세 번째 줄은 Hilbert 곡선으로 경로를 잡아 RLC 한 것이다. 표에서 보듯이 Hilbert 곡선의 경우가  $q_0, q_1$  의 값이 더 작아 결과적으로 더 나은 압축 효율을 나타냄을 알 수 있다.

**표 2** 1 비트 영상의 통계적 분석과 압축 결과  
**Table 2** The Statistical Analysis and Compression Results for 1 bit image

	Raster Scanning	Hilbert Scanning
$q_0$	0.05629	0.04576
$q_1$	0.09400	0.08497
Vw	19.75168	21.85441
Vb	10.6383	11.76953
Hwb	0.34529	0.32107



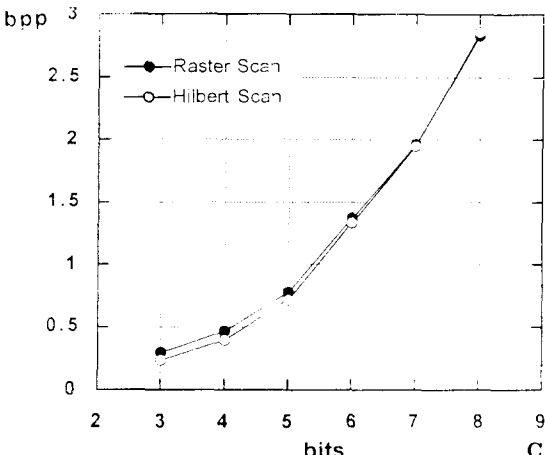
**그림 5** Raster Scan 과 Hilbert Scan 의 RLC 성능 비교 1.

- (a) 사용된 영상 (b) 결과 Entropy
- (c) (b) 의 그래프

**Fig. 5** RLC Performance Comparison of Chest Image between Raster Scanning and Hilbert Curve Direction Scanning  
(a) Chest Image (b) Result Table  
(c) Graph of (b)



	Raster Scan	Hilbert Scan
8 bit	2.8316	2.8568
7 bit	1.9596	1.9465
6 bit	1.3720	1.3363
5 bit	0.7779	0.7170
4 bit	0.4659	0.3971
3 bit	0.2984	0.2369



**그림 6** Raster Scan 과 Hilbert Scan 의 RLC 성능 비교 2.  
 (a) 사용된 영상. (b) 결과 Entropy  
 (c) (b) 의 그래프

**Fig. 6** RLC Performance Comparison of CT Image between Raster Scanning and Hilbert Curve Direction Scanning  
 (a) CT Image (b) Result Table  
 (c) Graph of (b)

**3. 의학 영상에의 적용**

<그림 5 (a)>는 512×512×8 비트 영상이고 이 영상을 여러 비트인 경우에 대하여 II.1 절에서 설명한 방법으로 RLC 한 결과의 entropy(bpp)를 <그림 5 (b),(c)>에 보였다.<그림 6>도 같은 방법으로 다른 영상에 대해서 적용시킨 예이다.<그림 5(c)>에서 원으로 이어진 곡선이 Raster 스캔으로 RLC 한 경우의 Entropy이고, 사각형으로 이어진 곡선이 Hilbert 곡선으로 RLC 한 경우의 Entropy 이다. 결과에서 보듯이 8비트를 다 쓴 경우는 사용되는 영상의 종류에 따라서 약간씩 다른 결과를 보이지만 7비트 이하의 영상에서는 점점 낮은 비트로 갈 수록 Hilbert 곡선으로의 RLC가 우수해짐을 알 수 있다, bit 수가 높은 영상에 대해서는 같은 값이 지속될 확률이 매우작아지므로 RLC의 효율이 자체적으로 높지 않게 나타난다. 이러한 영상에 대해서는 Hilbert 곡선 스캐닝의 방법은 뚜렷한 개선 효과를 나타내지 못했다.

**III. 검토 및 결론**

본 논문에서는 Hilbert 곡선 Scanning을 RLC 에 적용하여 압축효율을 향상 시킬 수 있음을 보였다. 영상의 종류에 따라서 약간씩의 차이는 있지만 Hilbert 곡선으로 진행하는 RLC가 기존의 Raster 스캔 방법보다 화소간의 중복성(Redundancy)을 더욱 많이 이용하였기 때문이다. 영상을 RLC 하기에 앞서서 인접한 화소끼리 일정한 규칙에 따라 재배치함으로써 같은 값이 지속되는 평균 길이가 길어지기 때문이다.

Hilbert 곡선을 이용한 RLC 방법에 있어서 전반적인 압축 효율 향상이 가능하였다. 특히 6 bit 이하의 낮은 bit per pixel 의 영상일 수록 효과적임을 알 수 있었다. 원래의 영상의 bit 수가 높은 영상에 대해서는 확실한 효과를 보지 못했다. 이러한 영상에 대해서는 RLC 자체가 한계일 수도 있다. 하지만 다른 스캔 방법을 연구하여 영상의 종류에 적합한 경로를 찾아낸다면 압축효율을 높일 수 있을 것이다. 이는 영상의 피사체의 기하학적인 모양과 패턴등 영상의 구조적 특성을 잘 반영하도록 곡선을 만들어야할 것이다. RLC 이외에도 Hilbert 곡선으로 재배치시킨 영상을 다른 영상처리에도 응용할 수 있을 것으로 생각된다. 이러한 방법은 추가

적인 H/W 부담 없이 의학 영상의 압축 효율을 개선시킬 수 있어 실용적으로 유용하게 사용될 수 있을 것이다.

#### IV. 참고 문헌

- 1) H.K. Huang, Prentice-Hall, 1987, 「Elements of Digital Radiology」
- 2) T.S. Huang, IEEE Trans. on Comm, VOL.COM-25(11) Nov. 1977, "Coding of Two Tone Images"
- 3) W.A. McWorter Jr. et al. BYTE Aug.1987, "Cr-eating Fractals"
- 4) A. Lempel, J. Ziv, IEEE Trans. on Information Theory, VOL.IT-32(1) Jan. 1986, "Compression of Two Dimensional Data"
- 5) M.F. Barnsley, Academic Press, 1988, 「Fractals Everywhere」
- 6) Jayant & Noll, Prentice-Hall, 1984, 「Digital Coding of Waveform」