

컴퓨터비전에서 사용되는 모양표시자의 현황

A Survey of Shape Descriptors in Computer Vision

유 헌 우, 장 동 식

(Hun-Woo Yoo, Dong-Sik Jang)

Abstract : Shape descriptors play an important role in systems for object recognition, retrieval, registration, and analysis. Seven well-known descriptors including MPEG-7 visual descriptors are briefly reviewed and a new robust pattern recognition descriptor is proposed. Performance comparison among descriptors are presented. Experiments show that the newly proposed descriptor yields better performance results than Fourier, invariant moment, and edge histogram descriptors.

Keywords : shape descriptor, MPEG-7, Fourier descriptor, invariant moment, edge histogram, Zemike moment, MLEV, curvature scale space, robust pattern recognition descriptor

I. 서론

영상 또는 영상 내에 존재하는 객체(또는 물체)를 추출하여 적절히 표현하고 인식하는 것은 영상처리나 컴퓨터비전에서 오랜 연구분야 중의 하나이다. 보통 객체를 표현할 때 색상(color), 질감(texture) 정보 등을 이용할 수 있으나, 이러한 특징들을 표현하기 위해서는 고가의 획득장비가 필요하고 추출된 특징은 조명변화에 민감하게 작용할 수 있기 때문에 전통적으로 모양정보를 이용하여 표현하는 경우가 많다. 객체의 모양을 적절히 표현하면 문자인식, 공장자동화에 쓰이는 마크인식과 검사, 인간의 세포분석과 분류, 목표 추적, 영상분석등 다방면에 사용될 수 있기 때문에 학술적인 측면이나 산업적인 면에서 유용성이 높고, 최근의 디지털기술의 발달로 수많은 영상데이터들을 효율적으로 검색할 수 있는 내용기반 영상검색(회사로고나 상표검색)등에도 사용될 수 있다.

과거로부터 객체의 모양을 표현할 수 있는 연구와 이론들이 무수히 많이 나왔지만 이들간의 성능 비교나 실제 산업현장에서 적용할 때 적합한 이론이 무엇인지를 알기가 쉽지 않다. 따라서, 본 논문에서는 이진영상이 주어졌다는 가정하에서 영상내에 존재하는 객체나 혹은 영상전체의 모양 정보를 적절히 표현 할 수 있는 표시자들 중 가장 최근까지 우수한 결과를 보이고 자주 사용되는 이론들을 중심으로 소개하고자 한다. 전통적으로 많이 쓰여왔던 푸리에 표시자, 불변모멘트 표시자와 영상의 에지의 각도를 이용한 에지히스토그램 표시자를 소개하며, 저자의 연구내용인 강인한 패턴인식 표시자를 설명할 것이다. 그리고 최근의 MPEG-7 관련해서 제안된 저니크모멘트, MLEV, CSS 등의 표시자도 기술할 것이다.

본 논문에서는 이러한 방법에 대해 실제 프로그래밍에 필요한 구체적인 수식까지도 요약 표현함으로써 알고리즘을 실현하는데 도움을 주고자 한다.

II. 모양표시자

모양 표시자(shape descriptor)는 보통 아래의 사항들이 고려되어야 한다.

- 1) 변환에 대한 강인성
 - 표시자는 위치변화, 각도변화, 크기변화등에 불변(invariant) 해야 하고 외각경계를 정의할 때 최소 시 작점에 대해서도 불변해야 한다.
 - 형태의 굴곡에도(rigid deformation) 어느 정도의 강인성이 보장 되어야 한다
- 2) 잡음에 대한 강인성
- 3) 표시자로 사용되는 특징들의 고속추출
- 4) 매칭시간의 고속화

물체(혹은 영상)의 모양을 표시할 때는 외곽 경계 혹은 물체의 내부에서 표시에 필요한 적절하고 유일한(unique) 특징들을 추출하고 이러한 특징들을 조합하여 벡터(feature vector)화 함으로써 나타내게 되고, 이 후 모양간의 매칭은 이러한 벡터들간에 유클리디안 거리(euclidean distance)등과 같은 유사도를 측정하는 방법을 사용하여 물체 혹은 영상을 인식하게 된다.

모양을 표현하기 위한 표시자로 최근의 MPEG-7 표준화에 사용되는 표시자를 포함하여, 전통적인 푸리에 표시자[1, 2, 3], 불변 모멘트 표시자[4, 5], 4방향과 8방향 에지히스토그램표시자[6, 7], 저니크모멘트[8, 9, 10, 11, 12], 다층고유벡터를 이용한 표시자[13, 14], 커버추어스케일 스페이스표시자[15, 16, 17, 18], 강인한 패턴인식표시자[19]등을 소개한다.

1. 푸리에 표시자(Fourier Descriptor)

물체의 외각선 경계를 따라서 움직이게 되면 경계의 위치좌표는 매 일주마다 주기적인 형태를 띄게 되므로 푸리에 시리즈를 통해 외각 경계의 모양을 근사화 할 수 있다. 물체의 경계선이 $u(n) = [x(n), y(n)]$, $n = 0, 1, 2, \dots, N-1$ 와 같이 N 개의 화소로 이루어 졌으면 연속적인 N 개의 좌표를 아래와 같이 $N(2\pi)$ 주기를 가지는 복소수의 형태로 표현할 수 있다(그림1).

논문접수 : 2002. 4. 18., 채택확정 : 2003. 1. 18.

유헌우 : 고려대학교 산업시스템정보공학과(paulyhw@korea.ac.kr)

장동식 : 고려대학교 산업시스템정보공학과(jang@korea.ac.kr)

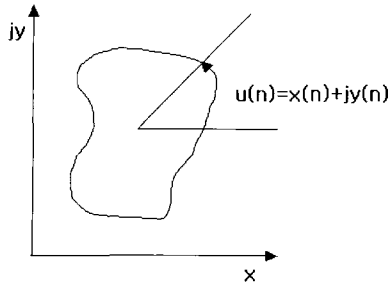


그림 1. 복소수의 형태로 표시된 물체의 외곽경계선.
Fig. 1. Object contour described by complex form.

$$u(n) = x(n) + jy(n), \quad n = 0, 1, 2, \dots, N-1 \quad (1)$$

이차원 좌표 시퀀스(sequence)를 일차원 시퀀스로 바꾼 후에는 이산형 푸리에 변환(discrete Fourier transform)을 통해 외곽경계를 아래와 같이 표현할 수 있다.

$$u(n) = \sum_{k=0}^{N-1} a(k) e^{-\frac{j2\pi kn}{N}}, \quad 0 \leq n \leq N-1 \quad (2)$$

$$a(k) = \frac{1}{N} \sum_{n=0}^{N-1} u(n) e^{-\frac{j2\pi kn}{N}}, \quad 0 \leq n \leq N-1 \quad (3)$$

여기서 $a(k)$ 를 푸리에 표시자(Fourier descriptor)라 하며 물체의 모양 정보를 표시하는데 유용하게 사용할 수 있다. 보통은 N 개의 데이터를 모두 사용하지 않고 중요하다고 생각되는 처음 몇 개의 데이터 M 개만을 사용하여 근사화 한다.

$$\widehat{u}(n) = \sum_{k=0}^{M-1} a(k) e^{-\frac{j2\pi kn}{N}}, \quad 0 \leq n \leq N-1 \quad (4)$$

여기서 주목할 것은 경계 $u(n)$ 을 구하는데 비록 M 개의 계수만을 이용하지만 실제 복원된 시퀀스는 N 개라는 것이다. 즉 근사후에 경계포인트를 복원하는데는 M 개의 데이터가 사용되지만 복원후의 시퀀스는 N 개의 데이터가 존재한다는 것이다. 상기의 푸리에 변환이 위치, 회전, 크기, 최초시작점 등에 불변(invariant)하기 위해서는 아래 표 1의 변환관계를 고려해야 한다.

표 1. 위치, 회전, 크기, 최초시작점 변경에 따른 푸리에 표시자변환관계

Table 1. Transform relation with Fourier descriptor according to the change of translation, rotation, scale, and initial point.

변환	경계	푸리에 표시자
Identity	$u(n)$	$a(k)$
Rotation	$u_r(n) = u(n)e^{j\theta}$	$a_r(k) = a(k)e^{j\theta}$
Translation	$u_t(n) = u(n) + \Delta_y$	$a_t(k) = a(k) + \Delta_y \delta(k)$
Scaling	$u_s(n) = \alpha u(n)$	$a_s(k) = \alpha a(k)$
Starting point	$u_p(n) = u(n - n_0)$	$a_p(k) = a(k)e^{-j2\pi n_0 k / N}$

여기서, $\Delta_y = \Delta_x + j\Delta_y$

2. 불변 모멘트(Invariant Moment)

모멘트는 일반적으로 어떤 기준점 혹은 기준축에서 어느 정도 떨어진 곳에서의 회전력을 표시하는데 사용되는 것으로 원래 기계공학에서 사용되는 개념이지만 패턴인식 분야의 모양 특징을 표시하는 데에도 사용되어 왔다. 영상이 2차원 공간에서 화소값들의 분포로 이루어졌다는 것을 고려하면 왜 모멘트가 영상분석등에 사용되어 왔는지를 알 수 있다. 따라서, 이 표시자는 물체의 전역적인 정보를 표현할 수 있고 외곽경계가 폐곡선을 이루지 않아도 된다는 장점을 가지고 있다(푸리에 표시자는 폐곡선을 이루어야 한다).

먼저, 2차원 연속함수 $f(x, y)$ 에 대해서, $(p+q)$ 차원 모멘트와 무게중심을 고려한 중심 모멘트(central moment)는 (5), (6)과 같이 정의되고,

$$m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q f(x, y) dx dy \quad (5)$$

$$\mu_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - \bar{x})^p (y - \bar{y})^q f(x, y) dx dy \quad (6)$$

여기서, $\bar{x} = \frac{m_{10}}{m_{00}}$ 와 $\bar{y} = \frac{m_{01}}{m_{00}}$ 는 무게중심의 좌표(centroid)이다

이것을 디지털 영상으로 아래 (7)과 같이 표현한 후,

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \quad (7)$$

3차원까지의 중심모멘트를 구하면 (8)-(12)와 같다[4, 5].

$$\mu_{00} = m_{00} \quad \mu_{11} = m_{11} - \bar{x} m_{10} \quad (8)$$

$$\mu_{10} = 0 \quad \mu_{30} = m_{30} - 3\bar{x} m_{20} + 2m_{10} \bar{x}^2 \quad (9)$$

$$\mu_{01} = 0 \quad \mu_{12} = m_{12} - 2\bar{y} m_{11} - \bar{x} m_{02} + 2\bar{y}^2 m_{10} \quad (10)$$

$$\mu_{20} = m_{20} - \bar{x} m_{10}, \quad \mu_{21} = m_{21} - 2\bar{x} m_{11} - \bar{y} m_{20} + 2\bar{x}^2 m_{01} \quad (11)$$

$$\mu_{02} = m_{02} - \bar{y} m_{01} \quad \mu_{03} = m_{03} - 3\bar{y} m_{02} + 2\bar{y}^2 m_{01} \quad (12)$$

정규화된 중심모멘트 η_{pq} 를 아래와 같이 정의하면,

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{r0}^r} \quad (13)$$

여기서 $r = \frac{p+q}{2} + 1, \quad p+q = 2, 3, \dots$

위치변화, 회전변화, 그리고 크기변화에 불변한 7개의 불변 모멘트는 다음의 식으로 계산된다[4,5]:

$$\psi_1 = \eta_{20} + \eta_{02} \quad (14)$$

$$\psi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (15)$$

$$\psi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (16)$$

$$\psi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (17)$$

$$\psi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (18)$$

$$\psi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (19)$$

$$\psi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03}) [3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (20)$$

3. 에지 히스토그램(Edge Histogram)

3.1 4방향 에지 히스토그램 (4-directional edge histogram)

자연영상에서 모양정보를 표시하기 위해서는 일반적인 객체의 경계정보를 추출하는 것이 불가능하므로 영상전체의 에지정보를 추출하여 모양에 대한 표시방법으로 삼는 것이다. 4방향 에지히스토그램 표시지는 수평방향에지, 수직방향에지, 두 개의 대각방향에지, 에지가 없는 것 등의 5가지 타입을 히스토그램화하여 특징으로 표시한다. 먼저 영상을 그림 2와 같이 부영상(Sub-image)이라 불리는 16개의 겹치지 않는 동일한 영역으로 분리한다. 각 부영상에서 5개 타입의 에지정보를 추출하고 이 방법을 모든 부영상에 대해서 구하면 총 16×5=80개의 히스토그램 빈(bin)을 얻을 수 있다.

에지 타입을 추출하기 위해서 그림 2에서와 같이 각 부영상을 영상블럭이라고 불리는 겹치지 않는 동일한 크기의 작은 블럭들로 나눈다. 일반적으로 블럭의 크기는 아래의 식 (21)-(22)로 구하는데 보통 2의 배수가 되도록 하며 그렇지 않은 경우는 배수로 나누어 떨어지지 않은 나머지 화소들은 무시하게 된다. 디폴트 블럭의 크기는 1100이다.

$$x = \sqrt{\frac{\text{image_width} \times \text{image_height}}{\text{desired_num_block}}} \quad (21)$$

$$\text{block_size} = \lfloor \frac{x}{2} \rfloor \times 2 \quad (22)$$

다음에는 각각의 블럭을 아래의 방법을 통해 5개의 에지타입 중 하나로 할당한다. 그림 3과 같이 블럭을 4개로 다시 분할하고 각 분할영역에서 (23)-(26)을 통하여 그레이값의 평균을 구한다. (i, j)th 영상블럭의 kth (k=0, 1, 2, 3)부분력에 대해 평균그레이값 A_k(i, j)는

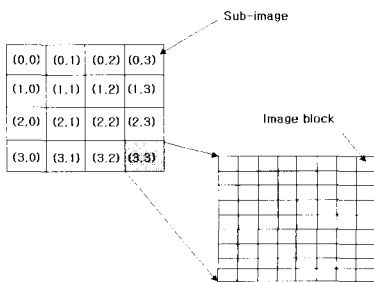


그림 2. 부영상(sub-image)과 영상블럭. Fig. 2. Sub-image and image block.

$$A_0(i, j) = \frac{4}{\text{block_size} \times \text{block_size}} \sum_{m=0}^{(\text{block_size}/2)-1} \sum_{n=0}^{(\text{block_size}/2)-1} I_{ij}(m, n) \quad (23)$$

$$A_1(i, j) = \frac{4}{\text{block_size} \times \text{block_size}} \sum_{m=\text{block_size}/2}^{\text{block_size}-1} \sum_{n=0}^{\text{block_size}-1} I_{ij}(m, n) \quad (24)$$

$$A_2(i, j) = \frac{4}{\text{block_size} \times \text{block_size}} \sum_{m=0}^{(\text{block_size}/2)-1} \sum_{n=\text{block_size}/2}^{\text{block_size}-1} I_{ij}(m, n) \quad (25)$$

$$A_3(i, j) = \frac{4}{\text{block_size} \times \text{block_size}} \sum_{m=\text{block_size}/2}^{\text{block_size}-1} \sum_{n=\text{block_size}/2}^{\text{block_size}-1} I_{ij}(m, n) \quad (26)$$

이고, 여기서, I_{ij}(m, n)은 (i, j)th영상블럭의 (m, n)th그레이값, A_k(i, j)는 (i, j)블럭의 kth부분력의 평균그레이값이다.

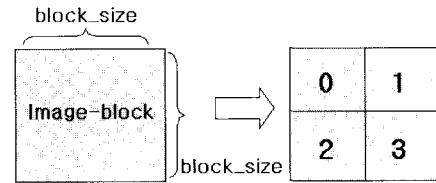


그림 3. 부분력들(sub-blocks)과 라벨링(labeling). Fig. 3. Sub-blocks and labeling.

이후 그림 4의 각방향의 에지 필터를 가지고 (27)-(31)을 통해 각방향별로의 에지의 강도를 구한다. 여기서 라벨링은 수직 에지를 예로 들면 ver_edge_filter(0)=1, ver_edge_filter(1)=-1, ver_edge_filter(2)=1, ver_edge_filter(3)=-1이다. 다른 필터의 라벨링도 동일한 방향으로 얻는다.

1 -1	1 1	√2 0	0 √2	2 -2
1 -1	-1 -1	0 -√2	-√2 0	-2 2
a) ver_edge_filter()	b) hor_edge_filter()	c) dia45_edge_filter()	d) dia135_edge_filter()	e) nond_edge_filter()

그림 4. 에지검출을 위한 필터계수.

Fig. 4. Filter coefficient for edge detection.

$$\text{ver_edge_stg}(i, j) = \left| \sum_{k=0}^3 A_k(i, j) \times \text{ver_edge_filter}(k) \right| \quad (27)$$

$$\text{hor_edge_stg}(i, j) = \left| \sum_{k=0}^3 A_k(i, j) \times \text{hor_edge_filter}(k) \right| \quad (28)$$

$$\text{dia45_edge_stg}(i, j) = \left| \sum_{k=0}^3 A_k(i, j) \times \text{dia45_edge_filter}(k) \right| \quad (29)$$

$$\text{dia135_edge_stg}(i, j) = \left| \sum_{k=0}^3 A_k(i, j) \times \text{dia135_edge_filter}(k) \right| \quad (30)$$

$$\text{nond_edge_stg}(i, j) = \left| \sum_{k=0}^3 A_k(i, j) \times \text{nond_edge_filter}(k) \right| \quad (31)$$

이후 (27)-(31)에서 구한 에지강도중에서 가장 큰 것이 미리 정한 임계값보다 크면 해당블럭을 해당에지타입으로 할당하고 작으면 에지가 없는 것(nond_edge_stg(i,j))으로 할당한다. 디폴트 임계값은 11이다.

에지히스토그램을 가지고 영상간의 유사도를 계산할 때는

위에서 구한 지역적 히스토그램(80개의 빈)과 지역적인 빈을 모두 에지타입별로 합산한 전역적 히스토그램(5개의 빈), 그리고 반전역적히스토그램(65개의 빈)을 통해서 수행한다(32참조).

$$\begin{aligned}
 dis(A, B) = & \sum_{i=0}^{16} |Local_Edge_A[i] - Local_Edge_B[i]| \\
 & + 5 \times \sum_{i=0}^4 |Global_Edge_A[i] - Global_Edge_B[i]| \quad (32) \\
 & + \sum_{i=0}^{64} |Semi_Global_Edge_A[i] - Semi_Global_Edge_B[i]|
 \end{aligned}$$

여기서, A, B 는 2개의 비교영상을 나타내고, $Local_Edge_A$ 는 A 영상의 지역적 에지히스토그램, $Local_Edge_B$ 는 B 영상의 지역적 에지히스토그램, $Global_Edge_A$ 는 A 영상의 전역적 에지히스토그램, $Global_Edge_B$ 는 B 영상의 전역적 에지히스토그램, $Semi_Global_Edge_A$ 는 A 영상의 반전역적에지 히스토그램, $Semi_Global_Edge_B$ 는 B 영상의 반전역적에지 히스토그램을 의미한다.

(32)에서 전역적 히스토그램의 차이에 가중치 5를 준 것은 해당 히스토그램의 빈이 5개밖에 존재하지 않기 때문에 다른 에지히스토그램과 균형을 맞추기 위해서이다. 여기서 반전역적히스토그램은 그림 5와 같이 영상을 13개의 군집화 영역으로 나누어 각 영역당 5개의 에지타입을 추출해서 나타낸다.

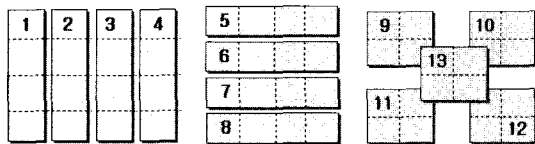


그림 5. 반전역적히스토그램의 13개의 군집화 영역.
Fig. 5. 13 clustering areas of semi-global histogram.

4방향 에지 히스토그램 표시자는 영상전반에 걸쳐서 에지의 방향을 5개 타입으로 히스토그램화 함으로써 객체분할이 되지 않는 일반적인 자연영상간의 유사도 계산에 유용하게 사용될 수 있다. 그러나 객체의 경계에 해당하는 모양정보를 효과적으로 표현하지 못한다는 단점이 있다. 이 표시자는 공간적인 에지분포의 정보를 가지므로 원래 영상의 질감정보를 표현하는데에 사용하도록 제안되었다(MPEG-7의 질감정보를 브라우징하는데에 사용되는 실험모델).

3.2. 8방향 에지 히스토그램(8-directional edge histogram)

4방향 에지 히스토그램이 에지타입을 4방향에지와 에지가 아닌 것으로 표현된 5가지타입을 빈으로 삼았지만 여기서 제안하는 방법은 8방향 ($-63^\circ, -45^\circ, -26^\circ, 0^\circ, 26^\circ, 45^\circ, 63^\circ$)의 에지를 구한 후 그 값을 합산하여 히스토그램화 하는 것이다. 이 방법은 에지의 각도를 보다 세분화했기 때문에 보다 효과적인 형태표현이 가능하다. 먼저 최초의 영상에 캐니에지검출자를 통하여 에지영상을 구한 후 3×3 마스크에 대해서 8근방의 화소를 주목하고 8근방중 2개의 화소가 에지로 판명되면 (33)을 통하여 에지의 각도를 구한다(그림 6). 만약 에지의 개수가 2개가 아니면 해당되는 마

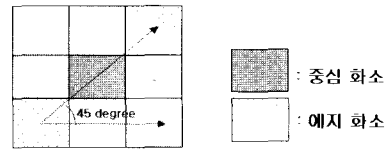


그림 6. 3×3 마스크와 에지각도.
Fig. 6. 3×3 mask and edge angle.

스크에는 방향성분을 가지는 에지가 없는 것으로 판단한다. 이와 같은 방법으로 에지방향을 구하면 총 8개의 방향을 얻을 수 있다.

$$Edge_direction = \tan^{-1} \left(\frac{y_0 - y_1}{x_0 - x_1} \right) \quad (33)$$

여기서, $(x_0, y_0), (x_1, y_1)$ 은 마스크 중심화소의 8근방화소 중 에지를 가지는 2개의 화소의 좌표. 상기의 마스크의 중심을 모든 화소에 대해 실시하고 8방향의 히스토그램의 빈을 누적한 후, 정규화과정을 거쳐 영상의 에지히스토그램으로 삼는다.

$$Edge_histogram(h), h = 0, 1, 2, \dots, 7 \quad (34)$$

4. 저니크모멘트(Zernike Moment)

MPEG-7에서 영역정보표시자의 하나로 실험중인 모델이다. 이 표시자는 물체의 형태를 표현할 때 외곽계정보뿐만 아니라 물체내의 정보등도 이용할 수 있다는 장점이 있고, 회전에 불변하고 노이즈에 강하며, 모양의 왜곡에도 다소 강한 특징을 가지고 있어서 상표검색에 효과적으로 적용되었다[10,11]. 저니크모멘트를 가지고 영역을 표시하기 위해서는 그림 7과 같은 과정을 먼저 거쳐야 한다.

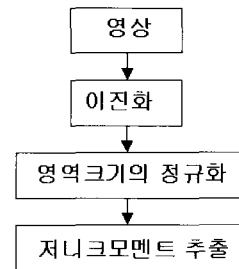


그림 7. 저니크모멘트 추출 블록다이어그램.
Fig. 7. Block diagram of Zernike moment extraction.

저니크모멘트를 구하는 방법은 다음과 같다. 먼저 (x, y) 좌표를 극좌표 (ρ, θ) 으로 바꾼 후 저니크 다항식(polynomial) $R_{nm}(\rho)$ 과 (n, m) 차 저니크 기지함수 $V_{nm}(\rho, \theta)$ 를 반경이 1인 원에서 (35), (36)과 같이 정의하면,

$$R_{nm}(\rho) = \sum_{s=0}^{(n-|m|)/2} (-1)^s \times \frac{(n-s)!}{s!((n+|m|)/2-s)!((n-|m|)/2-s)!} \rho^{n-2s} \quad (35)$$

$$V_{nm}(\rho, \theta) = R_{nm}(\rho) \exp(jm\theta), \rho \leq 1 \quad (36)$$

여기서 n 은 음이 아닌 정수이고, m 은 $n-|m|$ 이 짝수이고 $|m| \leq n$ 을 만족하는 영이 아닌 정수이다. 저니크모멘트 Z_{nm} 는 연속과 이산형으로 각각 (37)과 (38)과 같다.

$$Z_{nm} = \frac{n+1}{\pi} \int \int_{unit\ disk} V_{nm}^*(\rho, \theta) f(\rho, \theta), s.t. \rho \leq 1 \quad (37)$$

$$Z_{nm} = \frac{n+1}{\pi} \sum_{\rho} \sum_{\theta} V_{nm}^*(\rho, \theta) F(\rho, \theta), s.t. \rho \leq 1 \quad (38)$$

여기서 V_{nm}^* 은 V_{nm} 의 공액 복소수(complex conjugate)이다.

영상내에 존재하는 객체의 영역이나 외각 경계를 표시하기 위해서는 보통 8차까지의 저니크모멘트를 사용하고(표 2), 두 객체간의 유사도는 모멘트의 각 요소간 비교를 통해서 계산하게 된다(39)

$$similarity_distance = \sum_{i=0}^{L-1} W_i \times ||M_q[i] - M_d[i]|| \quad (39)$$

여기서, L 은 사용된 전체 모멘트의 수, $M_q[i]$ 와 $M_d[i]$ 는 각각 객체 q 와 객체 d 의 i 번째 모멘트이고, W_i 는 가중치이다.

저니크모멘트는 반경이 1인 원에 대해서만 정의되므로, 먼저 객체영역의 최외각점을 포함하는 반경 R 을 정하고 영역을 가로세로가 $2R \times 2R$ 로 정규화 하기 위하여 다시 샘플링을 한 후 계산을 하면 크기에 불변한 모멘트값을 얻을 수 있다. 보통 $R=50$ 을 사용한다. 저니크모멘트의 기저함수는 오프라인(off-line)에서 계산하여 메모리에 저장하고 모멘트를 구할 때는 룩업표(LookUp Table)[9]를 이용하면 빠르게 계산이 가능하다.

표 2. 8차까지의 저니크모멘트.

Table 2. Eight degree Zernike moments.

N	Moments	No. of moments
0	$Z_{0,0}$	1
1	$Z_{1,1}$	1
2	$Z_{2,0}, Z_{2,2}$	2
3	$Z_{3,1}, Z_{3,3}$	2
4	$Z_{4,0}, Z_{4,2}, Z_{4,4}$	3
5	$Z_{5,1}, Z_{5,3}, Z_{5,5}$	3
6	$Z_{6,0}, Z_{6,2}, Z_{6,4}, Z_{6,6}$	4
7	$Z_{7,1}, Z_{7,3}, Z_{7,5}, Z_{7,7}$	4
8	$Z_{8,0}, Z_{8,2}, Z_{8,4}, Z_{8,6}, Z_{8,8}$	5
9	$Z_{9,1}, Z_{9,3}, Z_{9,5}, Z_{9,7}, Z_{9,9}$	5
10	$Z_{10,0}, Z_{10,2}, Z_{10,4}, Z_{10,6}, Z_{10,8}, Z_{10,10}$	6

5. 다층고유벡터를 이용한 모양표시자(Multi-layer eigenvector shape descriptor(MLEV))

이 방법은 전통적으로 물체를 표현할 때 물체의 면적(area), 둥근 정도(circularity), 주축(major axis) 혹은 부축(minor axis)등을 보다 세분화한 영역에 이용한 것이다. 물체의 외곽경계의 N 개 화소들의 집합을 $l_i = [x_i \ y_i]^T$, $i=1,2,\dots,N$ 라고 하면 물체의 모양을 아래의 과정을 통해 단계적으로 부영역으로 나눈다.

Step 1 : 평균위치 m_L 와 상호분산 행렬 C_L 을 식 (40)-(41)로부터 얻고 $C_L e_j = \lambda_j e_j$, $j=1,2$ 로부터 고유벡터 e_j 와 고유치 λ_j 를 구한다.

$$m_L = \frac{1}{N} \sum_{i=1}^N l_i \quad (40)$$

$$C_L = \frac{1}{N} l_i l_i^T - m_L m_L^T \quad (41)$$

Step 2 : Step 1에서 구한 e_1, e_2 를 가지고 2개의 주축을 중심으로 물체의 모양을 4개의 영역으로 나눈다(그림 8).

Step 3 : 각각의 영역에 대해서 고유벡터를 구하고 Step 1 - Step 2의 과정을 반복해서 수행한다 (그림 9).

Step 4 : 그림 10과 같은 쿼드트리구조로 표시된 각각의 영역에 대해 위치, 크기, 각도변화에 불변한 4개의 특징들을 이후의 방법을 통해 수행한다.

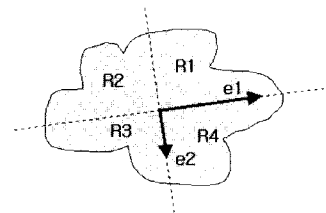


그림 8. 최초의 영역분할.
Fig. 8. Initial region segmentation.

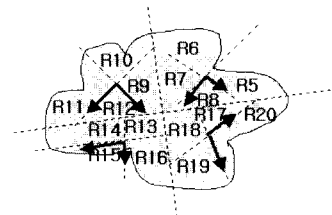


그림 9. 두 번째 영역분할.
Fig. 9. Second region segmentation.

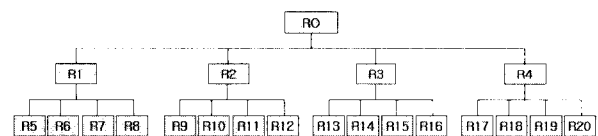


그림 10. 영역분할후에 쿼드트리 표현.
Fig. 10. Quad-tree representation after region segmentation.

영역 i 에 대해 주고유치와 부고유치, 주고유벡터와 부고유벡터를 각각 $\lambda_{b,i}, \lambda_{s,i}$ 와 $e_{b,i}, e_{s,i}$ 라 하면

1. *eigenratio* : 영역 i 의 *eigen_ratio*는

$$eigen_ratio(i) = \frac{\lambda_{s,i}}{\lambda_{b,i}} \quad (42)$$

2. *normal_angle* : 영역 i 의 주고유벡터 $e_{b,i}$ 와 최초 객체

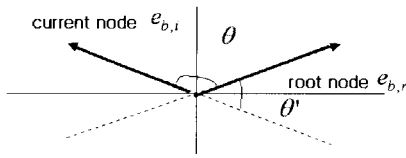


그림 11. normal_angle의 예.
Fig. 11. Example of normal_angle.



그림 12. 영역과 외접하는 최소사각형의 예.
Fig. 12. Minimum outer bounding rectangle.

영역 R_0 의 주 교유벡터 $e_{b,r}$ 사이의 각도를 나타내며 2개의 각도중 작은것을 선택한다(그림 11 의 θ 참조)
3. *center* : 영역 i 의 중심위치 c_i 와 최초 객체영역 R_0 의 중심위치 c_r 과의 거리를 영역 R_0 의 주교유벡터 $x_{b,r}$ 로 나눈 것이다.

$$center(i) = \frac{|c_i - c_r|}{\lambda_{b,r}} \quad (43)$$

4. *compactness* : 영역 i 의 크기 R_i 와 영역 i 를 중심축을 중심으로 외접하는 최소사각형의 영역 T_i 의 비율을 나타낸다 (그림 12).

$$compactness(i) = \frac{R_i}{T_i} \quad (44)$$

각 영역단위의 특징을 구한 후 두 개의 비교영상(또는 객체) q, s 간의 유사도는 아래의 식을 통해 계산한다.

$$d(q, s) = \sum_{i=0}^{K-1} |q_i - s_i| \quad (45)$$

여기서 K 는 특징벡터의 크기이다.
이 표시자는 MPEG-7의 외곽경계(contour)를 표시하는 표시자로 제안되었다.

6. 커버추어 스케일스페이스 표시자 (Curvature Scale Space Descriptor)

이 표시자는 물체의 외곽경계가 오목면(concavity)과 볼록면(convexity)들의 집합으로 이루어졌다는 점에 착안하여 경계의 커버추어(curvature)를 이용하여 물체를 표시하는 방법이다. 먼저, 물체의 외곽경계를 얻었다고 가정하면 경계의 길이를 정규화하기 위해 경계화소간의 길이가 같도록 N 개의 화소로 재샘플링한후 0-1로 정규화 한다. 그림 13과의 물고기 모양의 영상 데이터에 대해 표 3은 45개의 동일한 길이로 샘플링 후 정규화 한 값을 표시한다.

표 3에 정의된 외곽경계의 화소 $\Gamma = \{(x(u), y(u)) | u \in [0, 1]\}$ 에 대해 커버추어 함수는 아래의 (46)와 같다.

$$k(u) = \frac{\dot{x}(u) \ddot{y}(u) - \ddot{x}(u) \dot{y}(u)}{(\dot{x}^2(u) + \dot{y}^2(u))^{3/2}} \quad (46)$$

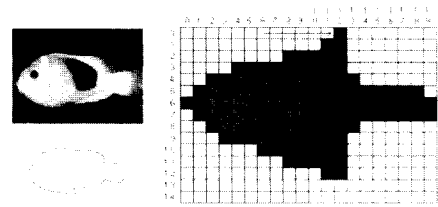


그림 13. 물고기 객체 영상과 외곽경계데이터.
Fig. 13. Fish image and its contour.

표 3. 원래의 외곽좌표와 동일한 길이로 샘플링과 정규화를 거친 후의 외곽좌표

Table 3. Initial outer boundary coordinates and coordinates after sampling in same interval and normalization

	x_i	y_i	u	u_{norm}
0	12	0	0.00000	0.00000
1	12	1	1.00000	0.01922
2	12	2	2.00000	0.03843
3	12	3	3.00000	0.05765
4	13	4	4.41421	0.08482
5	14	5	5.82843	0.11200
6	15	5	6.82843	0.13121
7	16	5	7.82843	0.15043
*	*	*	*	*
44	11	1	50.62742	0.97283
45	12	0	52.04163	1.00000

이제 이 커버추어 함수에 대해 다양한 표준편차를 가지고 가우시안 필터링을 하여 외곽경계를 보다 단순한 모양으로 변화시킨다(curve evolution). 표준편차가 σ 인 1-D 가우시안 함수를 $g(u, \sigma)$ 라 하면, 필터링 후의 커브는 (47)-(48)이 되고,

$$X(u, \sigma) = x(u) * g(u, \sigma) \quad (47)$$

$$Y(u, \sigma) = y(u) * g(u, \sigma) \quad (48)$$

이것을 x 축 방향으로 미분하면 (49)-(50)이 된다(y 축 $Y_u(u, \sigma)$ 와 $Y_{uu}(u, \sigma)$ 도 비슷한 형태로 미분).

$$X_u(u, \sigma) = x(u) * g_u(u, \sigma) \quad (49)$$

$$X_{uu}(u, \sigma) = x(u) * g_{uu}(u, \sigma) \quad (50)$$

따라서 표준편차가 σ 인 가우시안 필터를 적용한 후의 커버추어는 (51)이 된다.

$$k(u, \sigma) = \frac{X_u(u, \sigma) Y_{uu}(u, \sigma) - X_{uu}(u, \sigma) Y_u(u, \sigma)}{(X_u(u, \sigma)^2 + Y_u(u, \sigma)^2)^{3/2}} \quad (51)$$

이제 (u, σ) 평면에 대해 커버추어 스케일 스페이스(CSS) 영상을 얻는다. 먼저 $\sigma=1$ 로 하여 (51)을 통해 커버추어의 제로크로싱(curvature zero crossings)되는 값을 구한다. 다음 번에는 σ 를 1.1로 0.1 증가해서 동일한 방법으로 값을 구

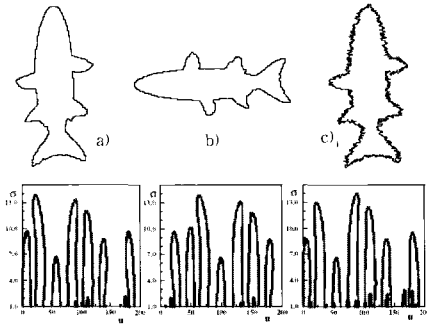


그림 14. 객체의 외곽경계와 CSS 영상.
Fig. 14. Object contour and CSS image.

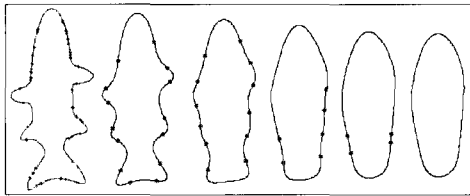


그림 15. 표준편차가 $\sigma=1,4,7,10,12,14$ 에 따른 외곽선의 수축과 부드러워짐(제로크로싱되는 점들이 적어진다).
Fig. 15. Contour contraction and softening according to standard deviation $\sigma=1,4,7,10,12,14$ (zero crossing points are being diminished).

하고 이와 같은 과정을 계속 반복하면 그림 14의 경우처럼 첫 번째 줄의 객체에 대해 두 번째 줄의 CSS 영상을 얻을 수 있다. 보통 σ 의 증가는 커버추어 제로크로싱값이 더 이상 없을 때까지 반복한다.

일반적으로 물체의 외곽경계를 CSS 로 표현할 때 σ 가 증가함에 따라 r_0 는 수축되고 부드러워 지며 커버추어 제로 크로싱되는 수가 적어짐을 알 수 있다(그림 15).

객체간의 유사도를 비교할 때는 CSS 영상에서 주요한 최고점(maxima)를 찾아서 이 최고점간의 거리차를 가지고 유사도를 계산하게 된다. 그림 14의 a)에서 주요한 최고점은 (22 13.8), (91 13.3), (109 12.0), (8 9.7), (181 9.6), (139 8.8), (57 6.7) 이다.

이와 같은 CSS 를 이용한 모양 표시자는 노이즈, 크기변화, 방향변화들에 불변한 특징을 가지고 있으며 MPEG-7의 모양(contour)에 대한 표준모델로 선정되어 있다.

7. 강인한 패턴인식 표시자 (Robust Pattern Recognition Descriptor)

이 표시자는 물체의 위치이동, 회전, 크기변화와 잡음에 강인하도록 설계되었다. 먼저 물체의 불변특징을 추출하고 그 특징벡터를 입력으로 하는 신경망을 이용하여 물체를 구별해 내는 방법이다.

먼저 물체의 외곽선이 주어지면 이 외곽선의 점들(N_0 개)과 물체의 중심(x_c, y_c)과의 거리(\vec{d})를 구하여 위치 변화에 불변한 특징을 얻는다.

$$\vec{d} = \{d_0, d_1, \dots, d_{N_0}\} \quad (52)$$

$$d_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}, \quad x_c = \frac{1}{N_0} \sum_{i=0}^{N_0-1} x_i, \quad y_c = \frac{1}{N_0} \sum_{i=0}^{N_0-1} y_i$$

다음에는 N_0 개의 점들에 대해 N 개의 일정크기를 샘플링하여 FFT(Fast Fourier Transform)을 통해 회전에 불변한 특징을 얻는다. 거리 벡터를 주파수 성분 $F(k)$ 로 분해한 후,

$$F(k) = \frac{1}{N} \sum_{i=0}^{N-1} d_i \exp(-j(\frac{2\pi ki}{N})), \quad k=0, 1, \dots, N-1 \quad (53)$$

여기서, $F(k) = P(k) + jI(k)$, $k=0, 1, \dots, N-1$ 각 주파수의 성분이 어느정도 포함되어 있는가를 나타내기 위하여 파워스펙트럼 $P(k)$ 를 구한다

$$P(k) = |F(k)|^2 = R(k)^2 + I(k)^2, \quad k=0, 1, \dots, N-1 \quad (54)$$

다음에 크기에 무관한 특징벡터를 얻기 위하여 각 하모닉대 별로 정규화된 파워스펙트럼의 크기를 얻는다.

$$p' = \frac{p}{|p|} = \frac{p_k}{|p|}, \frac{p_{k+1}}{|p|}, \dots, \frac{p_m}{|p|} \quad (55)$$

여기서, $p = p_k, p_{k+1}, \dots, p_m$, $|p| = \sqrt{p_k^2 + p_{k+1}^2 + \dots + p_m^2}$, k =하모닉의 시작 인덱스, m =하모닉의 끝 인덱스, $p_k = k$ 하모닉의 파워 스펙트럼이다.

최종적으로 위에서 특징값을 신경망(Fuzzy ART)의 입력으로 사용함으로써 물체를 인식하게 된다.

III. 성능비교

본문에서 소개한 8가지 알고리즘 중에서 푸리에 표시자나, 불변모멘트 표시자, 에지히스토그램 표시자는 나머지 5가지 알고리즘에 비해 크기변화와 각도변화의 민감성평가와 유사도 검색의 정확성 면에서 뒤떨어 것으로 나타났다[7, 14].

표 4는 푸리에 표시자, 불변모멘트 표시자, 에지히스토그램 표시자와 강인한 패턴인식 표시자간의 유사성 검색 성능 비교 결과이다. 임의의 400개의 물체에 대해 각 표시자별로 유사성을 검색을 실시한 결과 처음 10개의 검색결과를 가지고 정확도(precision)와 회수율(recall)을 비교한 것이다. 표의 결과로는 강인한 패턴인식표시자가 모든 면에서 우수한 성능을 나타냄을 알 수 있다. 그 이유는 이 표시자가 위치불변, 회전불변, 크기불변, 잡음에 강인성등을 극복할 수 있는 특징을 가지고 있기 때문이다.

MPEG-7 표시자들(저니크모멘트, MLEV, CSS)간에 대한 성능 비교[12]는 실험데이터 CE-1(그림 16-17과 같이 폐곡선으로 이루어진 단순 객체의 모양)에 대해서 CE-1-A(각도, 크기변화에 대해서 동일 영상을 매칭(exact matching)해내는 정도), CE-1-B(유사한 영상을 검색해 내는 정도), CE-1-C(어느 정도의 모양의 왜곡에도 검색해 낼 수 있는 강인성 정도)를 평가와, 실험데이터 CE-2(그림 18과 같이 복잡한 형태의 모양)에 대해서 CE-2-A(각도, 크기변화에 대해 동일 영상을 매칭(exact matching)해내는 정도), CE-2-B(유사한 영상을

검색해 내는 정도)를 평가결과, 전체적인 성능평가는 표 5-6에서와 같이 커버추어 스케일 스페이스 표시자가 CE-1에서 가장 좋은 성능을 보였고 저니크 모멘트, MLEV순이었다. 그러나 스케일 스페이스 표시자는 물체의 모양이 단순 폐곡선이 아닌 CE-2에는 적용할 수 없다는 단점이 있다.

표 4. 모양표시자 (푸리에 표시자, 불변모멘트 표시자, 에지히스토그램 표시자, 강인한 패턴인식 표시자) 간의 유사성 성능검색 비교.

Table 4. Shape descriptors(Fourier descriptor, invariant moment descriptor, edge histogram descriptor, robust pattern recognition descriptor).

	정확도	회수율
푸리에 표시자	79%	53%
불변모멘트표시자	85%	54%
에지히스토그램표시자	80%	55%
강인한 패턴인식표시자	95%	60%

표 5. 모양표시자의 성능비교(실험 타입별).

Table 5. Evaluation of performance comparison among shape descriptors (based on type of experiment).

Type of experiment	No. of images	No. of queries	Retrieval accuracy	Part average	Total average	
CE-1	A	Rotation	420	420	93.154%	96.457%
		Scale	420	420	100.00%	87.223%
	B		1400	1400	70.634%	70.634%
	C		1300	1	94.500%	94.500%
CE-2	A	Rotation	2881	100	94.184%	83.350%
		Scale	2921	140	69.800%	75.640%
		Rotation & Scale	3101	330	98.961%	
		Prefective transform	3100	330	70.455%	
	B		2800	682	67.931%	67.931%

표 6. 모양표시자의 성능비교(종합).

Table 6. Evaluation of performance comparison among shape descriptors (overall).

	Zernike moment	MLEV	CSS
CE-1	87.223%	84.848%	88.668%
CE-2	75.640%	62.256%	적용불가

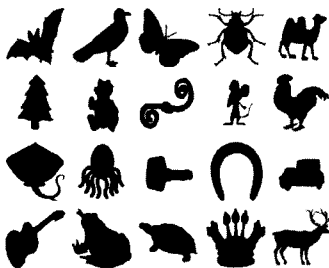


그림 16. CE-1에서 사용되는 몇 개의 영상들의 예.
Fig. 16. Some sample images of CE-1.



그림 17. CE-1영상에서 각 영상 클래스 내에서의 변화의 예.
Fig. 17. Examples of intra-class variations in CE-1.

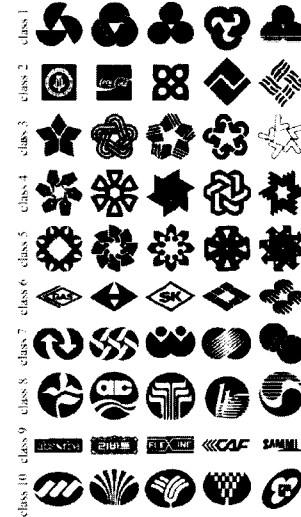


그림 18. CE-2에서 사용되는 몇 개의 영상들의 예.
Fig. 18. Some sample images of CE-2.

IV. 요약과 결론

지난 수십 년간 물체 혹은 영상의 모양을 표현하고자 하는 방법들에 대해 많은 연구가 진행되어 왔고 다양한 분야에서 성공적으로 사용되어 왔다. 본 논문의 목적은 최근까지 자주 사용되는 알고리즘을 소개함으로써 각 산업분야에 효과적으로 이용하도록 하는데 있다. 최근의 MPEG-7표준화에 사용되는 표시자를 포함하여, 전통적인 푸리에 표시자, 불변모멘트 표시자, 4방향과 8방향 에지히스토그램 표시자, 저니크모멘트, 다중고유벡터를 이용한 표시자, 커버추어 스케일 스페이스 표시자, 강인한 패턴인식 표시자등 8개의 모양 표시자를 선별하여 요약하고 성능을 비교하였다.

실세계의 모양을 인식하는 방법은 매우 다양하기 때문에 모양 표시자에 대한 연구는 앞으로도 계속 컴퓨터비전분야의 핵심 문제가 될 것이다. 따라서 인간이 모양을 인식하는 능력을 효율적으로 수치화 하여 보다 많은 분야에 성공적으로 적용하도록 해야 할 것이다.

참고문헌

- [1] C. T. Zahn, R. Z. Roskies, "Fourier descriptors for plane closed curves," *IEEE Transactions on Computer*. Vol. 21, pp. 269-281, 1972.
- [2] E. Persoon and K. S. Fu, "Shape discrimination using fourier descriptors," *IEEE Transactions on Sys. Man. Cyb. SMC-7*, pp. 170-179, 1977.
- [3] Gonzales & Woods, "Digital image processing," Addison Wesley Longman, 1992.

- [4] M. K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory*, Vol. 8, pp. 179-187, 1962.
- [5] S. A. Dudani, K. J. Breeding, and R. B. McGhee, "Aircraft identification by moment invariants," *IEEE Transactions on Computers*, Vol. C-26, pp. 39-46, 1977.
- [6] A. Yamada, M. Pickering, S. Jeannin, L. Cieplinski, J.R. Ohm, and M.C. Kim, "MPEG-7 Visual part of experimentation model version 9.0," ISO/IEC JTC1/SC29/WG11/P40-45, Pisa, January 2001.
- [7] J. L. Shih and L. H. Chen, "A new system for trademark segmentation and retrieval," *Image and Vision Computing*, Vol. 19, pp. 1011-1018, 2001.
- [8] A. Khotanzad and Y.H. Hong, "Invariant image representation by Zernike moments," *IEEE Transactions on Pattern analysis and machine intelligence*, Vol. 12, pp. 489-497, 1990.
- [9] W. Y. Kim and P. Yuan, "A practical pattern recognition system for translation, scale and rotation invariance," in *Proc. IEEE International conference on Computer Vision and Pattern Recognition*, pp. 391-396, June 1994.
- [10] Y. S. Kim and W. Y. Kim, "Content-based trademark retrieval system using a visually salient feature," *Image and Vision Computing*, Vol. 16, pp. 931-939, 1998.
- [11] Y. S. Kim, Y. S. Kim, W.Y. Kim, and M.J. Kim, "Development of content-based trademark retrieval system on the World Wide Web," *ETRI Journal*, Vol. 21, pp.39-53, 1999.
- [12] W. Y Kim and Y. S. Kim, "A region-based shape descriptor using Zernike moments," *Signal Processing: Image Communication* Vol. 16, pp. 95-102, 2000.
- [13] J. D. Kim and H. K. Kim, "Shape descriptor based on multilayer eigen vector," ISO/IEC JTC1/SC29/WG11/P517, Lancaster, February 1999.
- [14] K. K. Kim and J. D. Kim, "Region-based shape descriptor invariant to rotation scale and translation," *Signal Processing: Image Communication*, Vol. 16, pp. 87-93, 2000.
- [15] F. Mokhtarian and A.K. Mackworth, "A theory of multi-scale, curvature-based shape representation for planar curves," *IEEE Transactions on Pattern analysis and machine intelligence*, Vol. 14, pp. 789-805, 1992.
- [16] F. Mokhtarian, S. Abbasi, and J. Kittler, "Efficient and robust retrieval by shape content through curvature scale space," In *Proc. International Workshop on Image Database and Multimedia Search*, pp. 35-42, August 1996.
- [17] F. Mokhtarian, S. Abbasi, and J. Kittler, "Robust and efficient shape indexing through curvature scale space," In *Proc. British Machine Vision Conference*, pp. 53-62, September 1996.
- [18] F. Mokhtarian and S. Abbasi, "Shape similarity retrieval under affine transforms," *Pattern Recognition*, Vol 35. pp. 31-41, 2002
- [19] M. H. Kim, D. S. Jang and Y. G. Yang, "A Robust-Invariant Pattern Recognition Model Using Fuzzy ART," *Pattern Recognition*, Vol 34, pp. 1685-1696. 2001.

유 현 우

제어 · 자동화 · 시스템공학 논문지 제 7 권 제 4 호 참조.

장 동 식

제어 · 자동화 · 시스템공학 논문지 제 7 권 제 4 호 참조.