

수리형태론적 스켈리턴 영상을 이용한 형상인식

장 주 석[†] · 손 윤 구^{††}

요 약

본 논문에서는 패턴인식 시스템의 성능 향상을 목적으로 원영상의 데이터 량을 압축하고 난 뒤 형상을 인식하는 개선된 방법을 제안한다. 제안한 방법에서는 수리형태론적 연산을 사용하여 원영상을 미리 스켈리턴변환하여 데이터 량을 줄이고, 변환된 영상에서 이동 및 크기의 정규화와 회전불변의 과정을 수행하여 패턴을 정합하였다. 크기의 정규화는 형상인식에 필요한 픽셀의 수를 최소로 하여 정합을 하기위하여 스켈리턴의 픽셀들에게 가중치를 부여하고 이를 이용하여 크기를 조정하였다. 따라서 원영상에서 수행하는 이러한 과정들을 스켈리턴 영상에서 수행하게 함으로써 데이터 량이 크게 줄어들게 되어 기억장소의 용량이 최소화되고 연산의 량도 줄어들어 계산의 속도를 고속화 할 수 있게 하였다. 실험을 통하여 인식에 필요한 최적의 크기 인수를 조사하였고, 제안한 방법이 실제의 인식시스템 구현시 유용하게 사용할 수 있음을 확인할 수 있었다.

Shape Recognition Using Skeleton Image Based on Mathematical Morphology

Chu Seock Chang[†] · Yoon Koo Son^{††}

ABSTRACT

In this paper, we propose an improved method to recognize the shape for enhancing the quality of the pattern recognition system by compressing the source images. In the proposed method, we reduced the data amount by skeletonizing the source images using mathematical morphology, and then matched patterns after accomplishing the translation and scale normalization, and rotation invariance on the transformed images. Through the scale normalization, it was possible for the shape recognition at minimum amount of the pixel by giving the weight to the skeleton pixel. As the source images was replaced by the skeleton images, it was possible to reduce the amount of data and computational loads dramatically, and so became much faster even with a smaller memory capacity. Through the experiment, we investigated the optimum scale factor and good result was proved when realizing the pattern recognition system.

1. 서 론

물체를 자동으로 인식하기 위한 연구와 이를 위한

시스템의 개발은 꾸준히 이루어져 왔다. 그러나 아날로그 값을 가진 자연 영상을 디지털 시스템에 입력시키는 컴퓨터비전 시스템은 조명과 잡음 및 영상의 왜곡 현상 등으로 인한 문제와 방대한 량의 영상 데이터를 처리하는데 필요한 기억장소 확보와 이에 수반되는 처리 속도 문제 등, 아직도 범용 시스템의 구축에는 여러 가지 어려움이 따르고 있다.

[†] 정 회 원: 대구전문대학 전자계산과 교수

^{††} 정 회 원: 영남대학교 전산공학과 교수

논문접수: 1996년 1월 24일, 심사완료: 1996년 3월 15일

자연 영상의 물체를 인식하기 위하여서는 일반적으로 크게 두 가지의 과정이 있다. 첫 번째 과정은 CCD카메라나 스캐너와 같은 영상 입력 기기를 통하여 얻은 원영상을 여러 가지 필터를 이용하여 잡음을 제거하고, 명암도 조절 및 평활화와 예리화 등의 과정을 거쳐 좀더 개선된 영상을 만들며[1,2], 또한 이들 영상을 분할하거나 분석하여 필요한 정보를 추출하는 것이다. 두 번째는 앞의 과정에서 얻은 결과를 이용하여 인식하는 과정으로 통계적인 방법[2,3,4], 구문론을 이용하는 방법[1,2], 그리고 최근에는 신경망을 이용하는 방법[5,6,7], 수리형태론을 이용하는 방법[8,9,10] 등 여러 가지가 사용되고 있다.

이제까지의 영상 처리에 대한 접근은 신호의 대수적인 구조를 분석하는 데 주력하여 왔다. 그러나 기하학적인 패턴을 중요한 요인으로 보유하고 있는 이진 영상 신호의 경우, 대수적인 구조보다는 기하학적인 구조를 강조하는 표현 방식이 필요하며, 이러한 표현 중의 한가지가 스켈리턴이다. 일반적으로 스켈리턴이란 용어는 형상을 간략화하고, 크기와 회전 및 연결성에 대한 정보를 전달하는 이진 영상의 세선화된 골격을 묘사하기 위하여 사용되어 왔다[1,2]. 이러한 스켈리턴은 생물학적 표현[11,12], 패턴인식[13], 영상코딩[14], 공장 자동화의 검사[15] 등에 응용되고 있으며, Kotelly[16]와 Calabi[17]가 연속 영상에 대하여, Rosenfeld와 Pfaltz[18] 및 Montanari[19]가 이산 영상에 대하여 연구하였다. 수리형태론은 Matheron[20], Serra[21], Haralick[22] 등에 의해 개발, 발전되어 왔으며, 원영상의 기하학적인 특징을 그대로 보존하면서 영상 처리에 필요한 여러 가지 연산을 제공하여 준다는 장점이 있으므로 영상의 분해, 압축 및 복원, 잡음 제거와 형태 인식 등에 광범위하게 사용되고 있다[1,2,20,21,22,31,32,33].

수리형태론을 이용하여 물체를 인식하는 방법으로는 크게 원영상을 형태 분해하는 방법[23,24,32]과 원영상을 이용하여 인식 과정을 좀더 쉽고 간편하게 할 수 있는 여러 가지 특징들을 추출하면서 이 영상을 대변할 수 있는 다른 형태의 표현 방법으로 변환한 뒤 이를 사용하는 방법[25,26,31]이 있다. 전자는 주어진 물체를 여러개의 단순한 원시 형태들로 분해한 뒤 이를 적절한 표현 방법을 이용하여 나타낸 뒤 인식하는 것이고, 후자는 물체의 경계 정보를 이용하지

나, 세선화 혹은 골격화를 실행하여 적은 정보량으로 주어진 물체의 기하학적인 정보를 표현할 수 있는 방법을 이용하여 인식하는 것을 말한다. 전자는 개념적으로는 비교적 단순하지만 수행 시간이 많이 걸리고 물체의 회전에 대하여 취약하며 정확한 인식에는 미흡한 면이 있는 반면, 후자의 방법은 물체의 영상을 적은 데이터 량으로 표현 가능하며, 실행 시간도 비교적 작게 소요되므로 이러한 방법이 많이 사용되고 있으며, 또한 이들을 고속으로 처리하기 위한 여러 가지 하드웨어들도 연구되고 있다[27,28,29]. 수리형태론을 이용한 스켈리턴 변환[2,22]은 기존의 변환 방법에 비해 변환 과정이 형태론적 녹입과 열림연산에 의해 단순화되면서도 스켈리턴 변환된 영상은 불림연산에 의해 원영상으로 정확히 복구될 수 있는 장점이 있다.

따라서 본 논문에서는 수리형태론을 이용하여 2차원 영상내에 포함되어 있는 원영상을 스켈리턴 변환시킨 뒤, 변환된 결과영상에 포함되어 있는 정보를 이용하여 이동, 회전 및 크기에 불변이면서 뛰어난 인식률을 가지고, 또한 고속으로 인식을 실현할 수 있는 정합 방법을 제시하였다. 제시된 방법은 어떤 물체의 형상에서 외부보다는 내부로 들어갈수록 본체에 더 접근된 정보를 가지고 있다는 일반적인 인지 방법에 따라서, 수리형태론을 이용하여 스켈리턴변환을 수행한 뒤, 스켈리턴영상의 각 픽셀에 그들이 형성될 당시의 순서에 준하는 가중치를 부여하고, 최대의 가중치를 가진 픽셀의 위치가 일정한 위치에 오도록 스켈리턴 영상을 배치시킴으로서 이동정규화를 실현하며, 스켈리턴 영상크기의 최대치가 일정한 값이 되도록 전체 스켈리턴영상의 각 좌표와 각 픽셀의 가중치들을 조정하여 크기에 대한 정규화를 실현하였으며, 패턴 정합시 스켈리턴영상을 회전시키면서 최소의 거리값을 찾아내도록하여 회전에 대한 불변인 정합방법을 구현하였다. 제안한 방법으로 임의의 영상에 적용하여 실험을 수행한 결과 우수함을 확인하였다.

본 논문의 2장에서는 수리형태론에 대하여, 3장에서는 수리형태론을 이용한 스켈리턴 변환에 대하여, 4장에서는 본 논문에서 제안하는 정규화 및 정합알고리즘에 대하여, 5장에서는 타당성 검증을 위한 실험 및 결과를 기술하였으며, 6장에서는 결론으로써 끝을

맞았다.

2. 수리형태론

수리형태론은 집합론에 근거를 둔 연산으로서 임의의 영상에서 객체의 형상을 처리하는 분야에 광범위하게 사용되고 있는데, 수리형태론 연산자들을 이용한 필터는 기존의 다른 필터와는 달리 영상의 기하학적인 특징들을 보존해 주면서, 불필요한 요소들을 제거해 줌으로써 영상을 단순화시키고 동시에 유용한 기하학적 정보를 추출하는데 뛰어난 기능을 발휘하므로 영상 처리에 많이 사용되고 있다.

수리형태론 연산자[2,22]에는 기본적으로 녹임(erosion)과 불림(dilation)이 있으며, 이 두연산자를 조합한 열림(opening)과 불림(closing)이 있다. 이들은 형태소(Structuring Element)라 불리는 연산의 탐침역할을 하는 작은 영상을 이용하여 연산을 수행하는데, 녹임은 밝은 부분을 축소하는 기능을 가지고 있으며 식(1)과 같고, 불림은 밝은 부분을 확장시키며 식(2)와 같다. 또한 열림은 형태소의 크기보다 작고 밝은 부분을 제거하여 영상에서 객체의 형상에 대하여 작은 협곡부분을 제거하고 경계부분을 부드럽게하며 식(3)과 같고, 불림은 열림과 반대로 형태소 크기보다 작고 어두운 부분을 영상에서 제거하여 길고 좁은 만이나 구멍 또는 틈을 메우며 식(4)와 같다. 또한 이들을 응용하여 영상 처리에 유용하게 사용할 수 있는 여러가지 수리형태론적 변환방법들이 개발되어 있다 [1, 2, 21, 22].

E^N 은 모든 점, $p = \{x_1, x_2, \dots, x_N\}$ 의 집합이며, A와 B는 E^N 의 부분집합일 때 형태소에 의한 연산은 다음과 같다.

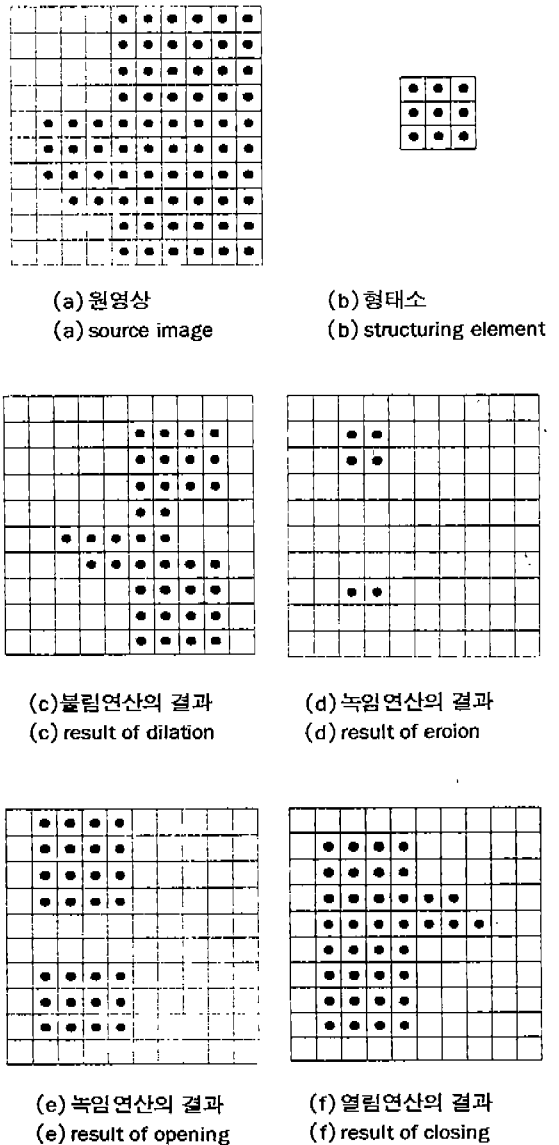
$$A \ominus B = \{x \in E^N \mid x + b \in A \text{ for every } b \in B\} \quad (1)$$

$$A \oplus B = \{c \in E^N \mid c = a + b \text{ for some } a \in A \text{ and } b \in B\} \quad (2)$$

$$A \circ B = (A \ominus B) \oplus B \quad (3)$$

$$A \bullet B = (A \oplus B) \ominus B \quad (4)$$

(그림 1)은 수리형태론연산의 수행 예를 보이고 있다.



(그림 1) 수리형태론 연산
(Fig. 1) Morphological operation

3. 스켈리턴 변환

영상의 스켈리턴 변환은 1967년 BLUM[11, 12]에 의하여 중심축 변환이란 개념으로 제시되었으며 원영상의 기하학적 정보를 그대로 가지고 있으면서 작은 량의 데이터로 영상을 표현하므로 패턴인식과 영

상압축 및 복원 등에 아주 많이 사용되는 변환 기법 중의 하나이다.

수리형태론을 이용한 스켈리턴 변환은 Lantue'joul [30], Serra[21], Maragos[14] 등에 의하여 제시되었는데, 원영상에 대하여 연속적으로 녹임과 열림 연산을 수행함으로써 변환이 이루어지며, 원영상 A의 스켈리턴 영상 S(A)는 다음과같이 표현될 수 있다.

$$S(A) = \bigcup_{k=0}^K S_k(A) \tag{5}$$

$$S_k(A) = (A \ominus kB) - [(A \ominus kB) \circ B] \tag{6}$$

여기서 K는 A가 공집합이 되기 이전까지의 녹임 횟수, B는 형태소, $(A \ominus kB)$ 는 A를 k회 연속 녹임함을 의미한다. 즉

$$K = \max \{k \mid (A \ominus kB) \neq \emptyset\} \tag{7}$$

$$(A \ominus kB) = (\dots(A \ominus B) \ominus B) \ominus \dots \ominus B \tag{8}$$

이며, 위 식에서 A의 스켈리턴 S(A)는 스켈리턴 부분 집합 $S_k(A)$ 의 합으로 구할 수 있다. 또한 스켈리턴 변환된 영상 S(A)는 다음과 같이 각 픽셀에 대한 불림연산에 의해 원영상으로 복원될 수 있다.

$$A = \bigcup_{k=0}^K (S_k(A) \oplus kB) \tag{9}$$

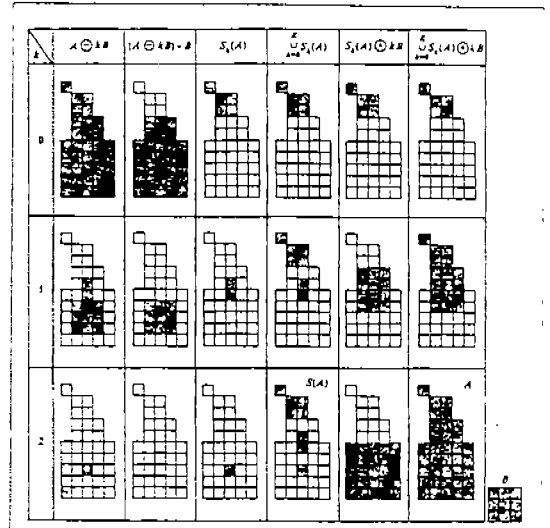
여기서 $(S_k(A) \oplus kB)$ 는 $S_k(A)$ 를 형태소 B로 k번 연속 불림 연산함을 의미한다.

$$(S_k(A) \oplus kB) = (\dots(S_k(A) \oplus B) \oplus B \oplus \dots) \oplus B, \tag{10}$$

k회 반복

(그림2)는 식(5)에서 식(10)까지에 근거한 스켈리턴 변환 및 복원의 예를 보이고 있는데[2], 좌측 상단은 원 영상을 나타내고 있으며 형태론적 스켈리턴 영상 S(A)는 4번째 칸의 하단에 나타나 있다. 그리고 이를 다시 복원한 영상은 최 우측 하단에 나타나 있다.

Maragos와 Schafer[14]는 2차원적인 Serra의 복잡한 알고리즘을 1차원으로 감축시키는 고속화된 스켈리턴 변환 및 복원알고리즘을 제시하였다. 변환알고리즘은 영상 A를 kB로 녹임하는 것은 기본형태소 B로 k번 연속녹임하는 것과 같다는 식(8)의 원리를 이용



(그림 2) 스켈리턴변환 및 복원
(Fig. 2) Skeleton transformation and restoration

하였으며(그림 3)에 나타내었다. 복원알고리즘은 스켈리턴 부분집합으로부터 원영상으로 복원하는 것으로 모든 부분집합에 대하여 그들이 만들어질 당시의 형태소로써 불림연산을 한 뒤 이를 합하면 된다. 즉,

$$A = (S_K(A) \oplus KB) \cup (S_{K-1}(A) \oplus (K-1)B) \cup \dots \cup S_0(A) \tag{11}$$

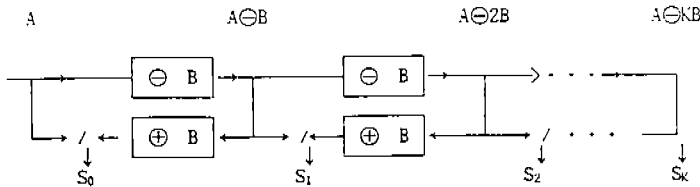
이며, 여기서

$$\begin{aligned} S_K(A) \oplus KB &= S_K(A) \oplus B \oplus B \oplus \dots \oplus B, K \\ S_{K-1}(A) \oplus (K-1)B &= S_{K-1}(A) \oplus B \oplus B \oplus \dots \oplus B, K-1 \\ &\dots \\ S_0(A) &= S_0(A) \end{aligned}$$

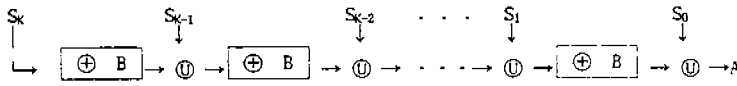
이므로 다시 표현하면

$$A = [S_K(A) \oplus B] \cup [S_{K-1}(A) \oplus B] \cup [S_{K-2}(A) \oplus B] \dots \cup S_0(A) \tag{12}$$

이며, (그림 3)에 나타내었다.



(a) 스켈리턴 변환
(a) Skeleton transform



(b) 스켈리턴 부분집합으로부터 복구
(b) Restoration from skeleton subset

(그림 3) 고속 변환 알고리즘
(Fig. 3) Fast transform algorithm

4. 스켈리턴 정규화 및 정합알고리즘

4.1 스켈리턴의 정합

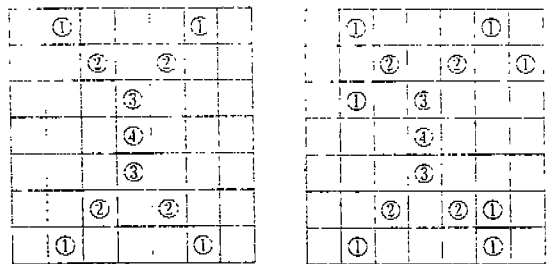
일반적인 형태인식추면에서 볼때 물체의 전체적인 구조 형상을 통한 인식이 미세한 부분을 통한 인식보다 더 정확한 인식결과를 도출할 수 있으므로, 이러한 사실에 기인하여 인식 알고리즘을 구성하였다.

스켈리턴 부분집합 $S_k(A)$ 에서 k 는 스켈리턴 픽셀이 형성된 순서를 나타내므로 가장 큰 k 값을 가지는 픽셀은 가장 나중에 형성되었다. 그러므로 $S_k(A)$ 에 포함되어 있는 k 값을 이용하여 각 픽셀의 무게값을 나타내는 스켈리턴 함수 S_r 를 다음과 같이 표현할 수 있다.

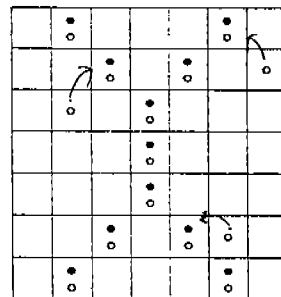
$$[S_r(A)](x, y) = \alpha \cdot (k + 1), \quad (x, y) \in S_k(A), \quad \alpha \text{는 계수}$$

$$\phi, \quad (x, y) \notin S_k(A) \quad (13)$$

여기서 k 값에 비례하게 무게값 S_r 를 각 픽셀에 부여한다면 큰 S_r 값을 가진 스켈리턴 픽셀은 원래의 형상에 대하여 기하학적으로 중요한 정보를 가지고 있는 것을 의미하고 작은 S_r 값을 갖는 픽셀은 형상의 외부의 세부 경계이거나 또는 잡음 등을 나타낸다고 볼 수 있다. 따라서 스켈리턴 픽셀의 좌표를 이용한 두 형



(a) 스켈리턴 패턴 1
(a) Skeleton pattern 1
(b) 스켈리턴 패턴 2
(b) Skeleton pattern 2



(c) 정합과정
(c) matching process

(그림 4) 스켈리턴의 정합과정
(Fig. 4) Skeleton matching process

상간의 정합은 유클리드 거리(Euclidean distance)와 각 픽셀에 부여된 S_r 값을 고려하여 그들간의 유사도를 거리 값으로 나타낼 수 있으므로 최소 거리 분류기로 사용할 수 있다. 즉, 거리 값의 계산 결과 가장 작은 값을 두 형태의 유사도가 높은 것으로 선택한다. (그림 4)는 정합 과정의 예를 보인것인데, 그림에서 보듯이 각 픽셀들은 무게값 S_r 를 가지고 있으며, \circ 와 \bullet 이 동일한 픽셀위치에 표시된 부분은 정합결과값이 0인 지점을 나타내고 있으며, \circ 이나 \bullet 이 픽셀위치에 한개만 표시된 부분은 두 패턴사이의 차이를 나타내는 부분으로 해당 픽셀에서 가장 가까운 거리에있는 상대 패턴과의 거리(\rightarrow 으로 표시됨)와 무게값을 고려한 값을 구하여 이들의 합을 얻으면 결과적으로 두 패턴 사이의 닮은정도를 나타내는 거리값이 된다.

4.2 정규화

영상의 정합에서는 이동 및 크기에 대한 정규화 및 회전에 대한 불변이라는 세 가지 조건이 만족되어야 한다[23]. 스켈리턴 변환은 이동 및 크기에 대한 정규화 와 회전에 대한 불변이 아니기 때문에 정합이 이루어지기 이전에 정규화 과정을 별도로 거치게하여야 한다. 이러한 전제 조건이 만족되지 않은 상태에서의 정합은 큰 오차를 유발할 수 있다.

일반적으로 이동에 대한 정규화는 물체의 영상이 예정된 위치에 오도록 무게중심을 원점에 이동시키며, 크기 정규화는 원영상의 픽셀수가 미리 정해진 일정한 상수가 되도록 영상의 크기를 조정한다. 또한 회전에 대한 정규화는 원 영상의 중심 축과 수평축의 각도를 미리 예정된 값으로 맞추기 위하여 원 영상을 일정한 각도만큼 회전시켜야 한다. 그러나 이러한 정규화 방법은 실제 구현에 있어서 다음과 같은 어려운 문제를 가지고 있다.

첫째, 복잡한 원영상에서 무게의 중심을 계산하기가 어렵다.

둘째, 크기를 조정하는데 많은 시간과 계산 과정이 소요된다.

셋째, 원영상에서 중심 축과 수평축을 찾기 위하여서는 미리 일정한 특징점이 있어야 한다. 그러나 이러한 특징점의 요소를 찾기가 어려운 형상에서는 정규화 과정이 힘들다.

따라서 본 논문에서는 이러한 정규화 과정을 개선

한 정합 방법을 제안하고자 한다. 우선 임의의 원영상에 대하여 정규화가 되지 않은 상태에서 스켈리턴 변환을 수행한 후, 결과 영상을 이용하여 정합 하는 과정에서 이동과 크기의 정규화 및 회전에 대한 불변성을 구현하고자 한다.

스켈리턴 변환 연산식에서 k 값은 스켈리턴 연산이 수행된 순서를 나타내고 있으므로, 스켈리턴 연산의 연속적인 과정은 형태소의 크기를 한 단계씩 키워 가는 녹임연산과 열림연산의 수행이다. 따라서 한 단계의 스켈리턴변환 연산을 수행할 경우 원영상은 기본 형태소에 의한 축소가 이루어진다. 또한 원영상에 대하여 연속적인 녹임연산을 수행할 경우 공집합이 되기 직전의 연산 횟수를 나타내는 K 는 원 영상에서 최대 내접원의 크기를 나타내며, 이때 남아 있는 픽셀의 위치는 최대 내접원의 중심점이 된다. 따라서 영상에서 최대 내접원의 중심점 위치는 형상의 전체적인 구조의 핵으로써 형태의 시작점으로 볼 수 있으므로 전체 형상의 원점으로 간주하여 K 의 위치를 형상 비교 과정에서 두 영상의 비교 중심점으로 사용하면 이동에 대한 정규화를 실현할 수 있다.

또한 최대 내접원의 크기인 K 값은 전체 형상의 크기에 근사한 의미의 값을 가지고 있으므로 K 값을 미리 정한 일정한 상수 β 가 되도록 전체 스켈리턴 영상의 크기를 조정하여 크기에 대한 정규화를 실현한다. 즉 모든 스켈리턴 픽셀 $f(x, y)$ 에 대하여 계수 γ 를 이용하여 식(14)와 같이 변경된 픽셀 $f_i(x, y)$ 로 좌표 이동하고, 변경된 픽셀의 무게값 S_n 는 S_r 를 이용하여 식(15)와 같이 계산한다[25].

$$f_i(x, y) = f(x/\gamma, y/\gamma) \quad (14)$$

$$\text{여기서 } \gamma = \sqrt{K/\beta}$$

$$S_n = S_r \cdot \frac{\beta}{K} \quad (15)$$

이 과정이 원영상에 대하여 크기 변화를 수행한 것과 동일한 효과를 가진다고 보면, 처리속도면에서는 고속이되며 상당한 기억 장소의 절감을 가져올 수 있다.

일반적인 형상표현 및 정합[23, 24, 25]에서는 원영상의 특징점을 이용하여 회전의 정도를 찾아 일정한 각도로 회전시키거나 또는 형태소를 회전시킨 뒤 연

산을 수행한다. 따라서 이 과정에서는 상당한 량의 계산시간을 필요로 한다. 따라서 크기 및 위치의 정규화가 된 스켈리턴 영상을 이용하여 회전된 영상의 문제를 해결하고자 한다. 즉 스켈리턴 영상을 정합하기 위하여 원점(K값의 위치)을 기준으로 회전한후 최소거리값을 구하여 표준패턴과 비교하는 방법을 택하였다. 이 과정은 스켈리턴 영상이 원 영상에 비하여 아주 적은 픽셀을 가지고 있어 일반적인 방법에 비하여 훨씬 고속 처리가 가능하므로 효율적이라 할 수 있다.

영상에서 대상물체를 회전시키기 위한 회전 중심점의 좌표를 (x_0, y_0) 라고 할 때 회전 대상에 포함된 임의의 점 (x, y) 가 회전각 θ_1 로 회전되었을 때의 최종점 (x', y') 에 대하여 각점들을 복소 평면상에서 계산하여 보면 다음과 같다.

회전대상에서 임의의 점 (x, y) 와 회전중심점 (x_0, y_0) 를 잇는 선과 실수축이 이루는 각을 θ_2 라고 한다면 임의의 점 $B(x, y)$ 는 아래와 같이 표현될 수 있다.

$$B = x + jy = (x_0 + jy_0) + r(\cos \theta_2 + j\sin \theta_2) \tag{16}$$

여기서 $r = \sqrt{(x-x_0)^2 + (y-y_0)^2}$ 이며, 회전 중심과 회전대상의 임의의 점간의 거리이다.

회전 후의 최종점 $C(x', y')$ 는 아래와 같이 표현될 수 있다.

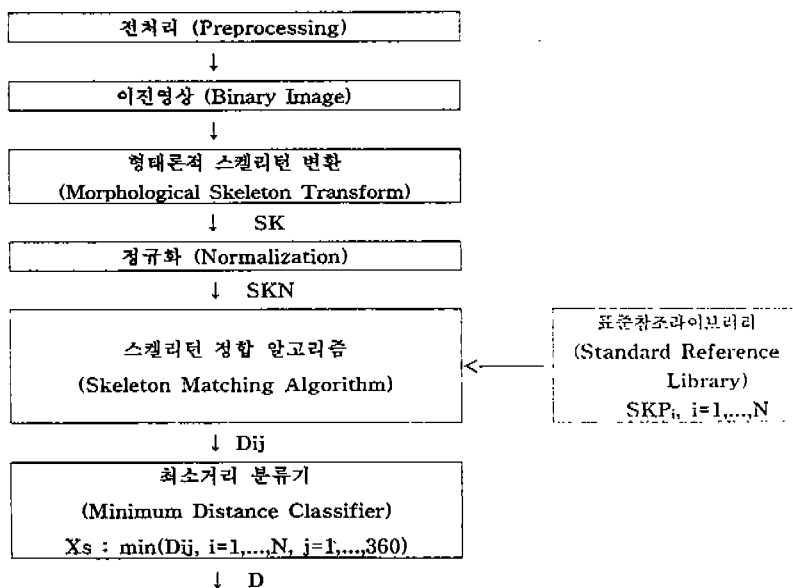
$$\begin{aligned} C &= x' + jy' \\ &= x_0 + jy_0 + r[(\cos(\theta_1 + \theta_2) + j\sin(\theta_1 + \theta_2))] \\ &= x_0 + jy_0 + r[(\cos \theta_1 \cos \theta_2 - \sin \theta_1 \sin \theta_2) \\ &\quad + j(\sin \theta_1 \cos \theta_2 + \sin \theta_2 \cos \theta_1)] \\ &= x_0 + jy_0 + r(\cos \theta_2 + j\sin \theta_2)(\cos \theta_1 + j\sin \theta_1) \end{aligned}$$

r 과 θ_2 를 소거하면

$$\begin{aligned} &= x_0 + jy_0 + [(x-x_0) + j(y-y_0)](\cos \theta_1 + j\sin \theta_1) \\ &= [x_0 + (x-x_0)\cos \theta_1 - (y-y_0)\sin \theta_1] \\ &\quad + j[y_0 + (x-x_0)\sin \theta_1 + (y-y_0)\cos \theta_1] \end{aligned} \tag{17}$$

로 된다. 따라서 회전후 최종점 (x', y') 는

$$\begin{aligned} x' &= x_0 + (x-x_0)\cos \theta_1 - (y-y_0)\sin \theta_1 \\ y' &= y_0 + (x-x_0)\sin \theta_1 + (y-y_0)\cos \theta_1 \end{aligned}$$

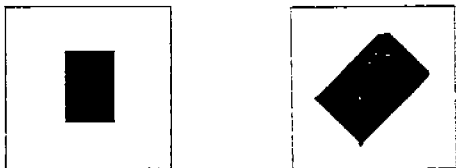


(그림 5) 제시한 정합 알고리즘의 흐름도
(Fig. 5) Flowchart for proposed matching algorithm

4.3 정합 알고리즘

일반적으로 정합을 위하여 원영상에서 수행하는 이동 및 크기의 정규화와 회전은 원영상의 많은 데이터 량으로 인하여 연산의 량이 많아 시간이 많이 소요되고, 무게 중심 및 특징점의 추출이 어려운 형상에서는 정규화와 회전이 어렵다. 따라서 본 논문에서는 원영상을 정규화 되지 않은 상태에서 스켈리턴 변환하여 데이터 량이 <표5>와 같이 크게 감소된 스켈리턴영상에서 정합을 수행함으로써 기억장소의 용량과 연산의 량을 대폭 감소하였고 또한 스켈리턴 변환 연산식의 최대 k값을 이용하여 스켈리턴영상의 중심점을 추출함으로써 크기의 정규화와 회전을 용이하게 할 수 있도록 하였다. 스켈리턴 영상의 회전에서는 일정한 각도로 회전하면서 표준라이브러리내의 스켈리턴 패턴과 무게값 S_k 를 고려한 거리값을 계산하여 최소거리값을 찾아 정합하였다. 본 논문에서 제시하고자 하는 알고리즘의 흐름도는 (그림 5)와 같으며 상세한 스켈리턴 정합 과정은 다음과 같다.

1단계 입력된 원영상에 대하여 기본적인 전처리후 입계치화에 의한 이진영상생성



(a) 정상적인 영상 (a) Normal image
(b) 변형 및 회전된 영상 (b) Distorted and Rotated image

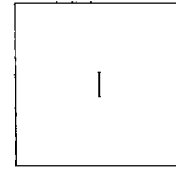
(그림 6) 이진화된 영상
(Fig. 6) binary images

(그림6)은 이진화된 두 개의 사각형 형상을 나타내고 있는데 (그림6-a)는 표준패턴을 생성하는 형상을 나타내었으며 (그림6-b)는 변형되고 회전된 형상을 나타내고 있다.

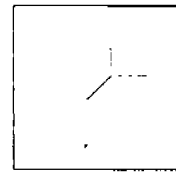
2단계 이진영상의 스켈리턴변환

3단계 제시된 방법 즉 K를 이용한 스켈리턴영상의 이동 정규화

(그림8)은 (그림7)의 영상에 대한 이동정규화된 결과로 사각형의 중심 즉, 좌표의 중심에 K값을 가진 픽셀이 위치하게 된다.

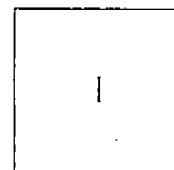


(a) (그림6-a)의 스켈리턴 영상
(a) Skeleton image of (Fig.6-a)

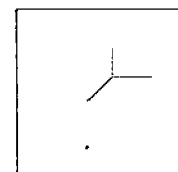


(b) (그림6-b)의 스켈리턴 영상
(b) Skeleton image of (Fig.6-b)

(그림 7) 스켈리턴 영상
(Fig. 7) Skeleton images



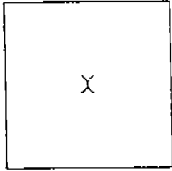
(a) (그림7-a)의 이동 정규화된 영상
(a) Translation normalized image of (Fig.7-a)



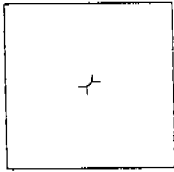
(b) (그림7-b)의 이동 정규화된 영상
(b) Translation normalized image of (Fig.7-b)

(그림 8) 이동 정규화된 영상
(Fig. 8) Translation normalized images

4단계 제시된 방법 즉 식(14) 및 식(15)에 의한 스켈리턴영상의 크기정규화



(a) (그림8-a)의 크기 정규화된 영상
(a) Size normalized image of (Fig.8-a)



(b) (그림8-b)의 크기 정규화된 영상
(b) Size normalized image of (Fig.8-b)

(그림 9) 크기 정규화된 영상
(Fig. 9) Size normalized image

(그림8)의 영상을 $\beta=10$ 에서 크기 정규화시킨 영상을 그림9에 나타내었는데 (그림8)의 영상에서 최대 S_r 의 값은 (a), (b) 각각 38과 39이었다.

5단계 1단계에서 4단계까지를 수행한 영상 SKN과 표준 라이브러리 내의 각 스켈리턴 패턴 SKP에 대하여 무게값을 S_r 를 고려하여 다음과 같이 거리값 계산 및 정합과정을 수행

```
for (i=1;i<=N;i++) /*표준라이브러리 내의 SKP의 수 N*/
for (j=1;j<=360;j++) /*SKN 회전각도*/
while (for each (x, y) ∈ SKN, find its nearest
point (x', y') ∈ SKPij)
W[(x, y), (x', y')] = |Sr(x, y) - Sr(x', y')| + 1;
R[(x, y), (x', y')] = √((x-x0)2 + (y-y0)2 ;
Dij(SKN, SKPij) = Dij(SKN, SKPij)
+ W[(x, y), (x', y')] · R[(x, y), (x', y')];
```

```
mark (x', y') as visited;
while (for each not visited point(x'', y'') ∈ SKPij,
find its nearest point (x, y) ∈ SKP)
Dij(SKN, SKPij) = Dij(SKN, SKPij)
+ W[(x, y), (x'', y'')] · R[(x, y), (x'', y'')];
D = min { Dij(SKN, SKPij), I=1, 2, ..., N,
j=1, 2, ..., 360};
```

여기서 $W[(x, y), (x', y')]$ 는 두픽셀 (x, y) 와 (x', y') 의 무게값 인수의 차이를 나타내며 $R[(x, y), (x', y')]$ 는 두픽셀 (x, y) 와 (x', y') 의 유클리드 거리 값을 나타낸다. 그리고 $D_{ij}(SKN, SKP_{ij})$ 는 입력 영상을 스켈리턴 변환시킨 뒤 정규화 과정을 거친 패턴을 j 로 회전시킨 것과의 정합 거리 값을 나타낸다. D 값은 라이브러리의 모든 패턴들을 회전시키면서 거리비교시 최소거리값을 나타내며, 이때의 패턴이 입력된 영상과 가장 잘 정합되는 패턴이 된다.

알고리즘에서 두 개의 스켈리턴 영상 비교에서 각 픽셀간 거리 차이에도 무게값의 차이 +1을 곱하기함으로써 D_{ij} 의 값은 형상의 지역적 차이보다는 형상의 전체적인 구조상 차이를 고려한 것이라 볼 수 있다. 여기서 1을 더한 것은 동일한 무게값을 가지면서 위치가 다른 두픽셀의 정합 거리 값이 0이 되는 것을 방지하기 위함이다.

(표1)은 (그림9)에 나타낸 두영상에 대한 정합 결과 거리값으로 이때 표준라이브러리의 패턴은 (그림6)의 영상을 스켈리턴화 하고 $\beta=10$ 에서 정규화시킨 것을 사용하였다.

(표1-a)에서는 (그림9-a) 즉 S06와 표준영상들간의 정합 결과 거리값으로 최소거리 결과값이 0이되는 것은 (그림9-a)가 표준영상들 중의 하나이기 때문이다.

(표1-b)는 (그림9-b) 즉 D06의 형상이 S06과 유사한 형태이면서 변형과 회전된 상태에서, 정규화된 후 정합과정에서 1도 단위로 회전되면서 계산한 거리값 중 최소거리 값을 산출한 결과이다. 따라서 표에 의하면 라이브러리 내에 다른 표준패턴 보다는 S06과의 정합에서 가장 작은 거리값을 나타냄을 알 수 있다.

(표2)는 (그림9-a)와 (그림9-b)의 영상이 회전되면서 정합하는 과정에서 산출한 거리값을 나타내고 있는데 (그림9-b)의 형상이 315도 회전되었을 경우 최소거리값을 나타냄을 알 수 있다.

〈표 1〉 S06과 D06의 정합 결과값
 〈Table 1〉 Matching distance of S06 and D06

(a) S06의 정합결과 값

(a) Matching distance of S06

거리값 $\times 10^{-2}$

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
S06	6.29	12.35	6.69	8.47	5.77	0	6.00	46.19	6.47	4.92	27.25	16.91	1.92	6.89	84.93	1.29	9.00	8.33	12.42	9.53

(b) D06의 정합결과 값

(b) Matching distance of D06

거리값 $\times 10^{-2}$

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
D06	6.51	11.38	6.83	8.90	5.42	1.24	5.69	48.51	6.95	4.64	28.18	16.94	2.22	6.50	88.70	1.51	9.36	6.84	14.34	10.49

〈표 2〉 (그림9-a)와 (그림9-b)의 정합과정시 생성되는 거리값
 〈Table 2〉 Procedural distance of (Fig.9-a) and (Fig.9-b)

거리값 $\times 10^{-2}$

0°	5°	10°	15°	20°	25°	30°	35°	40°	45°	50°	55°	60°	65°	70°	75°	80°	85°	90°	95°
25.54	26.48	28.38	25.14	17.30	11.26	9.55	8.69	7.84	8.93	8.73	9.99	10.53	12.14	14.87	30.80	29.17	27.61	21.62	21.50
100°	105°	110°	115°	120°	125°	130°	135°	140°	145°	150°	155°	160°	165°	170°	175°	180°	185°	190°	195°
11.94	9.98	8.08	7.34	5.40	4.15	3.31	3.12	3.57	4.14	5.12	6.92	8.30	9.12	10.00	11.65	26.15	26.92	27.77	25.32
200°	205°	210°	215°	220°	225°	230°	235°	240°	245°	250°	255°	260°	265°	270°	275°	280°	285°	290°	295°
20.30	12.98	11.68	11.07	10.35	9.75	10.39	10.59	10.66	11.53	19.34	28.56	26.63	25.84	24.94	9.61	8.75	6.89	5.69	4.86
300°	305°	310°	315°	320°	325°	330°	335°	340°	345°	350°	355°								
3.31	2.45	1.52	1.24	1.63	2.71	3.73	5.11	6.75	7.14	10.12	11.33								

5. 실험 및 결과

본 논문에서 제안된 방법의 타당성을 검증하기 위하여 (그림 11)과 같이 256×256 크기를 가진 임의로 선택한 20개의 정상적인 이진형상과 (그림 12)에 나타난 것과 같은 왜곡 및 회전된 20개의 형상을 만들어 실험하였다. 이들 형상들을 보면 일반적인 물체인 식이나 검사공정에서 처리대상이 될 수 있는 형상들을 취하고 있는데, (그림 11)에서 보듯이 사각형, 삼각형 및 원형 등과 이들과 유사한 변형된 형상들로 구성되어 있다. 이와 같은 형상들은 인식분야에서 여러 가지 모델에 자주 사용되는 것으로써[23], 사각형은 차량번호판 검출에서, 삼각형과 원형은 도로표시판 검출 등에서 그리고 변형된 형상은 겹쳐진 영상 입자 분리 분야 등에서 이용할 수 있다. 형상들을 스켈리

턴 변환시 여러 가지 형태의 형태소를 사용할 수 있고 또한 이에 따라 각기 다른 결과를 얻을 수 있으나, 스켈리턴의 기본 개념에 가장 적합한 모양의 형태소는 원(disk)이다. 디지털 시스템에서 원의 모양을 구현하기 위하여서는 최소한 5×5의 공간이 필요한데, 본 논문에서는 3×3 Rhombus와 3×3 정사각형 구성소를 연속으로 사용하여 원의 형태소를 사용하는 효과를 얻을 수 있게 하였다. 정합시 회전 각도는 1도 단위로 하였으며, α 값은 1로 하였다.

식(14)과 식(15)에서 최대 가중치 β 를 1에서부터 30까지 변화시키면서 (그림 11)의 표준 영상의 각 패턴 즉 라이브러리 내의 패턴과, (그림 12)의 회전과 왜곡이 가해지고 크기와 영상 내에서 위치가 다른 형상에 대하여 제안된 방법으로 정규화와 정합 과정을 거친 결과를 인식율 곡선으로 (그림 15)에 나타내었다. 그

림에서 β 값이 8인 지점 이후부터는 인식율이 100%라는 것을 알 수 있으며, <표 4>에서는 이때의 정합거리값을 나타내고 있다. 따라서 본 논문에서 임의로 채택한 실험 영상들을 인식하기 위하여서는 β 값이 8 이상만 되면 충분한 인식 결과를 나타내는 시스템의 구현이 가능하다. 또한 이때 인식하고자 하는 원영상의 픽셀수와 이를 스�কে리턴 변환한 후의 결과영상 픽셀수, 그리고 $\beta=8$ 로 정규화한 경우의 픽셀수를 <표 5>에 나타내었다. <표 5>에서 보듯이 원영상보다 스�কে리턴 변환한 경우의 픽셀수가 전체적으로 97.8% 줄었다는 것을 알 수 있으며, 완벽한 인식에 필요한 수준의 최소 픽셀수($\beta=8$ 인 경우)와의 대비에서는 원영상과는 99%, 그리고 스�কে리턴영상과는 67.1% 정도 줄었다는 것을 알 수 있다. 따라서 인식에 필요한 픽셀

수의 감소는 상당한 량의 기억 장소의 감소와 K값을 이용한 크기의 정규화 및 회전에서 사용 연산의 량이 줄어들게 되므로 계산상의 고속화를 이룰 수 있다.

<표 6>은 라이브러리 내에 있는 원래의 패턴과 회전 및 왜곡된 형상간의 정합거리를 β 값에 따라 나타내고 있다. 거리 값의 크기는 픽셀수에 따르고, 픽셀수는 형상의 면적이므로 식(14)에서 β 는 거리 값의 제곱근과 비례하게 된다. 따라서 표에서 보면, β 가 증가하게 되면 픽셀수가 증가하여 평균거리값도 증가하고 있으며, 그 제곱근의 값은 β 의 증가에 따라 일정한 비율로 증가하고 있으므로 크기의 정규화 과정이 원활히 수행되었음을 알 수 있다.

<표 3>에 나타난 $\beta=7$ 일 경우의 정합거리 값을 살펴보면 (그림10-a)에서 보는 D02의 영상이 표준패턴

<표 3> $\beta = 7$ 일때의 정합거리값
<Table 3> Matching distance at $\beta = 7$

거리값 $\times 10^{-2}$

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
D01	0.15	1.30	2.42	1.17	1.21	2.56	0.89	18.02	1.19	1.44	11.13	7.73	1.27	2.56	21.37	1.44	5.17	3.90	7.72	2.59
D02	0.95	1.40	2.36	1.38	1.48	2.56	0.93	18.41	1.47	2.59	14.35	6.43	1.29	2.82	26.88	1.78	5.82	4.94	9.06	4.38
D03	2.03	1.80	0.34	3.05	0.81	3.01	2.62	15.34	2.93	2.87	11.37	3.11	1.59	0.94	23.05	1.87	7.47	4.03	6.10	3.15
D04	1.09	1.43	2.90	0.50	1.76	3.55	1.26	18.97	0.78	2.94	15.51	7.68	1.73	3.60	33.14	1.79	4.91	5.24	8.22	4.45
D05	0.96	1.36	1.48	1.70	0.63	1.55	1.29	17.87	1.63	1.65	10.76	6.10	1.11	1.88	20.61	0.90	4.09	3.97	6.84	3.67
D06	1.93	3.77	2.84	3.46	2.54	0.36	2.71	16.60	2.91	2.37	10.13	6.35	1.08	3.17	23.99	0.78	2.57	2.50	5.47	4.56
D07	1.06	1.10	2.27	1.56	1.36	2.54	0.41	18.88	1.57	2.33	13.65	6.62	1.45	2.63	24.04	1.37	4.99	4.89	8.17	3.45
D08	20.85	19.08	13.50	19.29	18.57	19.04	19.42	5.09	20.51	15.43	17.39	12.89	16.35	14.57	21.84	18.24	16.63	13.90	10.91	14.45
D09	0.96	2.36	3.14	0.87	1.78	3.19	1.65	18.97	0.79	2.94	14.02	7.79	1.76	3.23	27.25	1.95	2.72	4.93	4.77	4.18
D19	2.17	3.02	3.00	3.73	2.09	2.42	3.37	13.44	3.55	0.77	12.98	6.28	3.52	3.13	20.85	2.83	7.67	1.43	9.09	2.49
D11	15.73	16.72	9.03	15.70	15.45	10.50	17.04	14.74	14.17	14.37	1.17	10.56	10.18	8.51	9.89	11.74	11.14	9.81	8.49	12.64
D12	6.55	5.94	2.82	7.70	5.21	7.19	6.91	13.06	6.64	6.32	11.26	0.36	5.50	3.38	22.23	4.95	10.60	6.44	7.67	6.54
D13	1.18	1.95	2.46	1.63	1.99	1.59	1.62	17.25	1.64	2.41	11.79	6.56	0.51	3.15	29.76	0.70	3.41	2.79	5.69	4.93
D14	3.10	3.04	1.27	3.99	2.37	2.91	3.50	14.04	3.76	3.66	8.94	3.04	2.04	0.66	20.73	2.21	6.61	3.75	6.28	4.26
D15	36.56	31.59	21.34	35.38	31.34	24.19	32.60	21.95	33.91	26.45	9.72	25.32	27.09	21.38	1.25	28.56	22.28	20.77	21.24	19.05
D16	1.57	2.41	2.05	2.68	1.18	0.63	1.85	17.20	2.25	1.51	10.77	6.21	0.91	2.29	23.86	0.56	3.10	2.89	5.45	3.78
D17	3.00	5.20	3.82	4.34	4.27	3.16	4.29	11.89	2.21	4.23	9.58	5.33	2.29	3.23	15.38	2.28	0.89	4.88	2.90	4.25
D18	4.32	5.22	4.70	4.55	4.36	4.27	5.55	10.07	5.39	2.75	10.35	6.81	4.27	5.20	18.79	4.55	7.11	0.71	5.60	3.93
D19	4.51	4.34	3.39	4.74	3.94	3.94	4.97	10.97	3.82	4.97	9.67	4.78	3.21	3.28	20.60	3.05	3.36	5.88	2.52	5.66
D20	2.61	4.09	4.00	4.94	3.15	3.37	3.74	12.46	3.63	2.53	10.41	5.96	2.72	3.59	18.25	3.32	5.46	2.31	7.93	1.59

인 S02의 영상과 최소정합 거리값을 산출하지 못하고 있음을 알 수 있다. 즉 원하는 패턴을 찾는것에 대하여 실패한 경우로써 이는 D02 영상의 스켈리턴 구조가 S01, S04, S07, S13의 스켈리턴의 구조와 많은 부분이 닮았다는 것을 (그림10)을 통하여 확인할 수 있으며, β 값이 8이상의 경우에는 비록 중복된 성분이 많이 있지만 스켈리턴 영상자체가 비교적 크기 때문에 타영상과 뚜렷한 차이의 거리값을 산출할 수 있다. 그러나 β 값을 7로 하였을 경우 이산좌표계에서 크기와 이동 정규화시의 오차와 회전시 정수값을 가진 각도 계산으로 인한 픽셀 좌표의 오차들이 비교대상 픽셀수가 작은 이들 영상들에 대하여 전체적인 정합거리 값에 상대적으로 큰 영향을 미친다는 것을 알 수 있다.

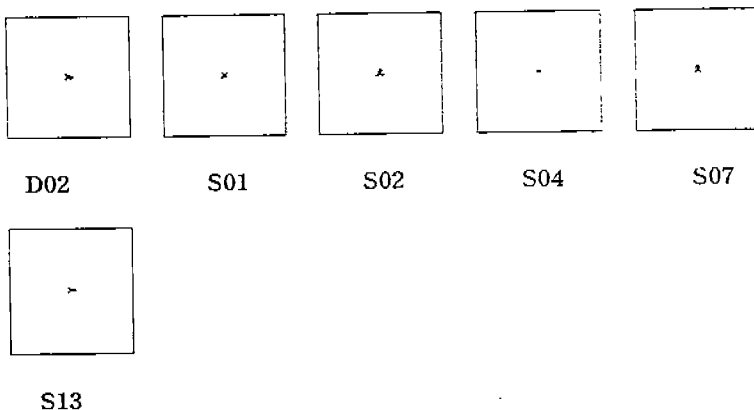
6. 결 론

본 논문에서는 이동, 크기 및 회전 불변이면서 잡음과 변형에 강한 형상 인식을 고속으로 구현하기 위하여 수리형태론적 스켈리턴 영상을 이용하였다. 수리형태론은 병렬구현이 용이하며 영상 처리의 다양한 분야에 적용할 수 있다. 제안된 방법은 원영상에 대하여 수리형태론을 이용하여 스켈리턴으로 변환을 수행하여 픽셀수를 줄이고, 정합을 위하여 크기와 이동에 대한 정규화 과정을 스켈리턴영상에서 구현하였다. 그리고 정규화된 스켈리턴영상을 일정한 각도

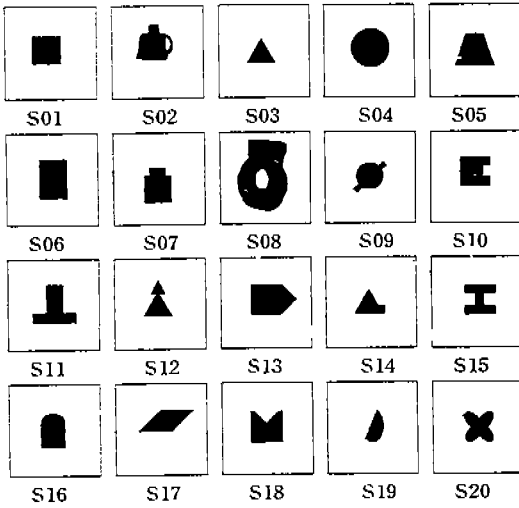
로 회전시켜 최소의 거리 값을 찾게 함으로서 회전에 무관한 정합 알고리즘을 제시하였다. 이러한 정합 방법은 원영상에서 정규화 한후 정합하는 방법과는 달리 원영상에 대한 정보를 가진 스켈리턴 픽셀들을 정합하는 것으로 원영상에 비하여 스켈리턴 변환후 영상의 픽셀수가 97.8%가 줄어들었고 또한 K값을 이용한 크기의 정규화에서 원영상과 대비하여 99%가 축소되었다. 따라서 연산의 양이 줄어들게 되어 인식의 속도가 빨라지고 사용하는 메모리의 용량도 줄일 수가 있었다. 또한 실험을 통하여 유용한 정합결과를 확인하였다.

스켈리턴 변환된 픽셀들을 β 의 값을 이용하여 크기 정규화를 하면서 픽셀수를 축소시킴으로서 원래의 형상에 대한 정보를 상당히 잃어버리게 되나, 원래의 스켈리턴 픽셀수가 원영상에 비하여 대단히 작으므로 소규모의 기억장소에 별도로 보관하여 원활히 복원할 수 있다.

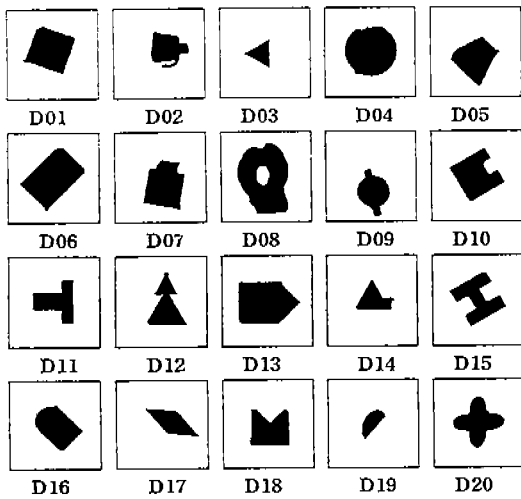
원영상에 있을 수 있는 기본적인 잡음에 대하여서는 전처리과정에서 수리형태론적 열림-붙임, 붙임-열림 등의 연산을 통하여 원영상의 큰 변형없이 이를 제거할 수 있다. 또한 원영상의 경계 부분에 잡음과 변형이 심한 경우에는 식(13)의 가중치 계수 α 를 조정하여 원영상의 중심부에 해당하는 스켈리턴 픽셀들이 상대적으로 높은 가중치를 가지도록 하거나 최대 S_n 값을 적당히 조정함으로써, 형상의 외곽 부에 있는 잡음 및 왜곡 성분에 해당하는 스켈리턴 픽셀들을



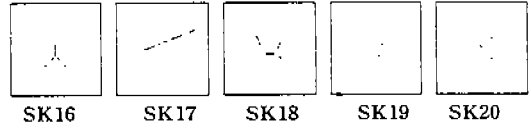
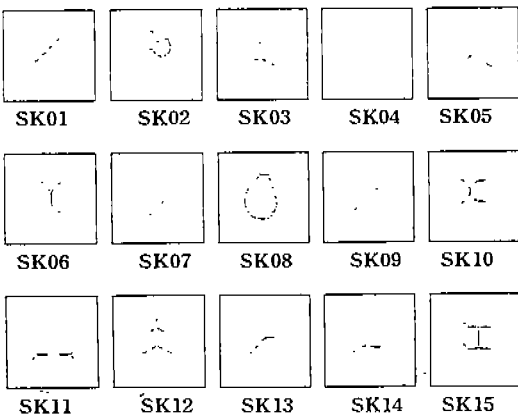
(그림 10) $\beta = 7$ 에서 정규화한 영상
(Fig. 10) Normalized image at $\beta = 7$



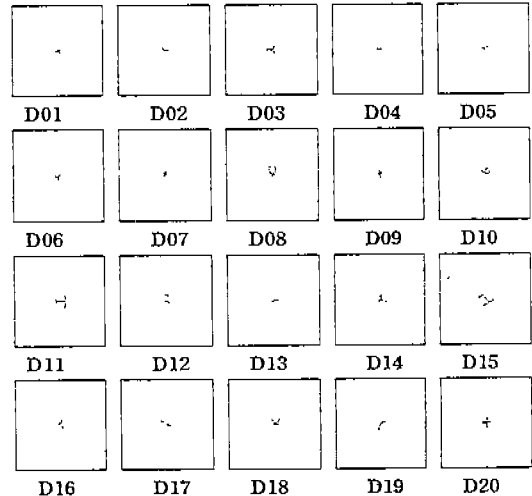
(그림 11) 정상적인 이진 영상
(Fig. 11) Original binary images



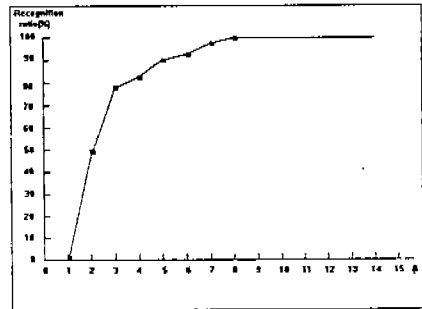
(그림 12) 회전 및 왜곡된 이진 영상
(Fig. 12) Rotated and distorted binary images



(그림 13) (그림 11)의 스켈리턴 영상
(Fig. 13) Skeleton images of (Fig. 11)



(그림 14) (그림 12)을 $\beta = 8$ 로 정규화한 후의 스켈리턴 영상
(Fig. 14) Skeleton images of (Fig. 12) after normalization at $\beta = 8$



(그림 15) β 값에 따른 인식율 그래프
(Fig. 15) Recognition ratio graph at β

제거, 혹은 무시 가능한 상태로하면 이러한 문제를 해결할 수 있다. 디지털화 된 영상처리로 인하여 크기의 정규화에서 이산성으로 인한 오차가 발생하였고 회전하여 정합거리값 계산시 고속으로 처리하기 위하여 최소값에 접근할 수 있는 알고리즘을 부과하면 더좋은 성능을 기대할 수 있다.

앞으로 다양하고 많은 형상들을 대상으로 하여 영상의 픽셀수에 대한 이상적인 최대 S_n 값, 즉 β 의 체계적인 분석 및 실험과 대상 형상의 형태와 왜곡 및 잡음의 정도에 대하여 α 값에 해당하는 함수에 대한 연구가 있어야 하겠다.

<표 4> 회전 및 왜곡된 형상과 표준 라이브러리 내의 패턴과의 정합 거리 값 ($\beta=8$ 에서 정규화됨) 거리값 $> 10^{-3}$

<Table 4> Matching distance between rotated, distorted shapes and patterns in the library (normalized at $\beta=8$)

	S01	S02	S03	S04	S05	S06	S07	S08	S09	S10	S11	S12	S13	S14	S15	S16	S17	S18	S19	S20
D01	0.06	0.35	0.32	0.16	0.19	0.51	0.11	2.88	0.16	0.17	2.04	0.85	0.20	0.37	2.88	0.20	0.83	0.61	1.18	0.36
D02	0.16	0.08	0.46	0.23	0.31	0.49	0.20	2.79	0.25	0.31	2.58	0.99	0.35	0.52	3.44	0.33	1.06	0.75	1.60	0.52
D03	0.24	0.52	0.07	0.37	0.11	0.37	0.29	2.29	0.57	0.42	2.07	0.40	0.26	0.15	3.05	0.28	1.13	0.69	1.16	0.39
D04	0.11	0.20	0.37	0.07	0.24	0.41	0.20	3.18	0.11	0.41	3.05	0.90	0.25	0.44	4.59	0.25	0.66	0.80	1.25	0.57
D05	0.12	0.36	0.16	0.25	0.09	0.14	0.13	2.39	0.24	0.24	1.82	0.59	0.16	0.20	2.85	0.12	0.66	0.60	0.96	0.46
D06	0.29	0.61	0.35	0.47	0.36	0.07	0.33	2.63	0.35	0.29	1.82	0.75	0.13	0.34	3.12	0.11	0.45	0.33	0.75	0.55
D07	0.15	0.24	0.31	0.20	0.18	0.53	0.05	2.78	0.19	0.29	2.08	0.96	0.22	0.31	3.07	0.20	0.77	0.70	1.20	0.45
D08	3.54	2.69	1.86	2.82	2.80	2.96	2.88	0.54	2.80	2.66	2.41	1.90	2.65	2.03	2.96	2.57	2.60	1.86	1.53	1.80
D09	0.11	0.29	0.41	0.15	0.24	0.41	0.22	3.04	0.38	0.58	2.50	1.01	0.24	0.45	5.88	0.27	0.47	0.77	0.74	0.50
D10	0.37	0.46	0.47	0.55	0.40	0.35	0.39	1.94	0.44	0.13	2.07	0.82	0.41	0.44	2.93	0.38	1.18	0.23	1.38	0.36
D11	3.16	2.65	1.57	2.71	2.26	1.90	2.71	2.17	2.51	2.33	0.10	1.63	2.12	1.49	1.40	2.38	2.17	1.72	1.58	1.72
D12	1.05	0.89	0.45	1.10	0.74	0.91	0.94	1.67	1.00	1.60	2.02	0.12	0.78	0.49	3.36	0.84	1.71	0.91	1.59	1.02
D13	0.12	0.32	0.38	0.20	0.18	0.18	0.26	2.49	0.25	0.29	2.14	3.78	0.05	0.35	3.88	0.10	0.57	0.46	0.84	0.58
D14	0.39	0.46	0.14	0.54	0.23	0.36	0.39	1.99	0.46	0.49	1.97	0.38	0.33	0.05	2.80	0.33	1.14	0.81	0.99	0.53
D15	5.59	4.54	3.45	5.03	4.43	3.59	4.87	2.95	4.65	4.09	1.45	5.65	5.68	3.10	6.19	4.30	2.82	2.94	3.13	2.66
D16	0.24	0.46	0.26	0.37	0.17	0.08	0.22	2.43	0.54	0.19	1.96	0.85	0.23	0.26	3.12	0.07	0.50	0.53	0.83	0.44
D17	0.44	0.75	0.51	0.64	0.51	0.38	0.54	1.55	0.51	0.41	1.89	3.70	0.54	0.41	2.25	0.35	0.14	0.63	0.64	0.42
D18	0.81	0.70	0.56	0.77	0.69	0.62	0.78	1.55	0.69	0.40	1.77	0.86	0.62	0.64	2.82	0.73	1.23	0.08	1.12	0.58
D19	0.75	0.73	0.45	0.77	0.54	0.56	0.69	1.70	0.58	0.80	1.75	0.68	0.44	0.48	2.80	0.44	0.49	0.69	0.41	0.78
D20	0.45	0.64	0.49	0.70	0.58	0.45	0.49	1.87	0.48	0.31	1.70	0.78	0.56	0.49	2.78	0.52	0.90	0.35	1.22	0.20

<표 5> $\beta=8$ 에서 각 형상들간의 픽셀수
<Table 5> Pixel amount of each images at $\beta=8$

	SRC	SK	SKN	SK/SRC(%)	SKN/SRC(%)	SKN/SK(%)
D01	12233	268	57	1.6	0.5	28.8
D02	7161	216	74	2.9	1.0	35.9
D03	2861	268	96	5.9	2.3	39.3
D04	16821	135	47	0.8	0.3	34.8
D05	10265	186	54	1.9	0.5	27.7
D06	16252	168	52	1.0	0.3	31.0
D07	11634	227	54	2.0	0.6	28.2
D08	20781	463	120	2.2	0.6	26.9
D09	7325	158	51	2.2	0.7	32.3
D10	11526	293	84	2.5	0.7	28.7
D11	7420	278	113	3.7	1.5	40.6
D12	7325	355	106	4.5	1.4	31.3
D13	16523	176	38	1.1	0.2	21.6
D14	4600	217	72	4.7	1.6	33.2
D15	9444	377	180	4.0	1.9	47.7
D16	10044	149	46	1.5	0.5	30.9
D17	5906	205	61	3.5	1.0	29.8
D18	9530	295	86	3.1	0.9	29.2
D19	2920	119	60	4.1	2.1	50.4
D20	9932	287	99	2.9	0.9	31.0

SRC:원 영상
SK :스캐러된 변환 후의 영상
SKN:정규화후의 영상

<표 6> β 값에 따른 회전및왜곡된 형상과 원래의 패턴간의 정합된 거리값
<Table 6> Matching distance between rotated, distorted shapes and original patterns at β

	5	8	10	15	20	25	30
D1	18	69	48	120	203	267	471
D2	51	88	197	339	475	362	990
D3	25	70	90	767	1546	1764	2894
D4	24	79	125	311	804	1053	1704
D5	38	95	126	384	628	2222	3469
D6	28	78	124	177	199	286	402
D7	22	54	69	216	260	426	567
D8	227	543	1300	2256	4559	6945	9615
D9	52	80	123	314	601	2896	2239
D10	53	134	134	322	663	840	1154
D11	91	100	271	208	374	240	87
D12	14	122	193	479	3259	5269	8366
D13	16	51	127	329	496	774	2402
D14	17	56	78	758	1381	2122	3071
D15	103	195	395	1070	2000	2347	3017
D16	32	78	92	1155	1922	5458	8127
D17	38	145	235	1320	2292	3380	5351
D18	29	80	154	295	539	735	1360
D19	102	410	1033	2383	4055	5608	8820
D20	66	206	424	923	1392	1736	2817
평균거리	52.3	136.65	256.9	706.3	1381.9	2236.5	3346.1
SQR. D	7.23	11.68	16.34	25.58	37.19	47.29	57.85

SQR. D:평균거리의 제곱근값

참 고 문 헌

- [1] Edward R. Dougherty, "Digital Image Processing Methods," Dekker, 1994.
- [2] Rafael C. Gonzalez, Richard E. Woods, "Digital Image Processing," Addison Wesley, 1992.
- [3] Keinosuke Fukunaga, "Introduction to Statistical Pattern Recognition," Academic Press, 1990.
- [4] Tamas Sziranyi, "Statistical Subpixel Pattern Recognition by Histograms," 11th IAPR International Conf. on Patt. Recog., August. 1992.
- [5] A. Del Bimbo, L. Landi, S. Santini, "Dynamic neural estimation for autonomous vehicles driving," IEEE Proc. IAPR., Vol.2, pp.350-354, Aug. 1992.
- [6] Eun Ryung Lee, et al., "Automatic Recognition of a Car License Plate Using Color Image Processing," IEEE proc. IAPR., pp.301-305, 1994.
- [7] C.H.Chen and G.H.You, "On Class-Sensitive Neural Network and Recognition Applications," World Congress On Neural Networks, Vol. IV, pp.43-46, July. 1993.
- [8] Wei Gong, Qinyun Shi, and Minde Cheng, "Shape and Image Matching by Use of Morphology," IEEE Proc. IAPR., Vol.2, pp.673-676, Aug. 1992.
- [9] Ping-Fai Yang and Petros Maragos, "Morphological Systems for Character Image Processing and Recognition," IEEE Proc. IAPR., Vol.5, pp. 97-100, 1993.
- [10] Xiaobo Li and Jun Yang, "Application of Mathematical Morphology an Multi-Object Motion Detection," ICARCV '94, pp.153-157, Nov. 1994.
- [11] H. Blum, "Picture Processing and Psychopictoris," Academic Press, 1970.
- [12] H. Blum, "Biological Shapes and visual science," J.Theor.Biol, Vol.38, pp.205-187, 1973.
- [13] O. Philbrick, "Shape recognition with the medial axis transformation," Proc. Symposium on Automated Photointerpretation, 1968.
- [14] P. Maragos and R. Schafer, "Morphological Skeleton Representation and Coding of Binary images," IEEE Tran. On Acoustics, Speech, and Signal Processing, Vol. ASSP-34, No.5, pp. 1228-1245, OCT., 1986.
- [15] J.R. Mandeville, "Novel Method for Analysis of printed circuit images," IBM J. Res. Develop., Vol.29, pp.73-86, 1985.
- [16] J.C.Kotelly, "A mathematical model of Blum's theory of pattern recognition," Air Force Cambridge Res. Labs. Bedford, MA, Rep., pp.63-164, April, 1963.
- [17] L. Calabi, "A Study of the skeleton of plane figures," Parke Mathematical Labs., Carlisle, MA, Rep. June, 1965.
- [18] Rosenfeld and Pfaltz, "Sequential operations in Digital picture processing," J. Assoc. Comput. Mach., Vol.13, pp.471-494, Oct. 1966.
- [19] U. Montanari, "A method for obtaining skeletons using a quasi-euclidean distance," J. Assoc. Comput. Mach., Vol.15, pp.600-624, Oct. 1975.
- [20] G. Matheron, "Random Sets and Integral Geometry," Wiley, 1975.
- [21] J. Serra, "Image Analysis and Mathematical Morphology," Academic Press, 1982.
- [22] Robert M. Haralick and Linda G. Shapiro, "Computer and Robot Vision," Addison Wesley, 1992.
- [23] S. Loncaric and A. P. Dhawan, "A Morphological Signal Transform For Shape Description," Pattern Recognition, Vol.26, No.7, pp.1029-1037, 1993.
- [24] I. Pitas and A.N. Venetsanopoulos, "Morphological Shape Representation," Pattern Recognition, Vol.25, No.6, pp.555-565, 1992.
- [25] P.E. Trahanias, "Binary Shape Recognition Using the Morphological Skeleton Transform," Pattern Recognition, Vol.25, No.11, pp.1277-1288, 1992.
- [26] L. Ji and J. Piper, "Fast Homotopy Preserving Skeleton Using Mathematical Morphology," IEEE Trans. PAMI, Vol.14, No.6, pp.653-664,

June, 1992.

[27] Shoji Kojima, Tatsuo Miyakawa, "Morphological Processor for Gray scale Image," ICARCV '94, pp.148-152, 1994.

[28] A. Broggi, G.Conte, G.Burzio, L.M.Reyneri, "A Real-time Morphological Image Processor," IEEE Proc., pp.654-658, 1994.

[29] Shoji Kojima, Yoshinobu Ebisawa, "Fast Morphology Hardware Using Large Size Structuring Element," 전자정보통신학회, vol.J76-D-II, No.6, pp.1106-1113, July, 1993.

[30] C. Lantuejoul, "Skeletonization in quantitative metallography," Issues of Digital Image processing, R. M. Haralick and J. C. Simon(eds.), Sijthoff and Noordhoff, Groningen, The Netherlands, 1980.

[31] 손진우외, "수리형태론 연산자를 이용한 도로정보의 특징 추출에 관한 연구," 전자공학회 논문지, Vol.32-B, No.11, pp.132-141, 1995.

[32] 최중호, "형태론적 패턴인식 시스템의 개발-형상 함수를 이용한 형태론적 형상분해," 전자공학회 논문지, Vol.32-B, No.8, pp.83-92, 1995.

[33] 강대성, "칼라영상처리와 형태학적 필터를 이용한 물체인식 시스템에 관한 연구," 한국통신학회 논문지, 20권 7호, pp.1981-1991, 1995.



장 주 석

1982년 영남대학교 전자공학과 졸업(학사)
 1984년 영남대학교 대학원 전자공학과(공학석사)
 1995년 영남대학교 대학원 전자공학과(박사수료)
 1984년~현재 대구전문대학 전자계산과 교수

관심분야:영상처리 패턴인식



손 윤 구

1956년 서울대학교 전기공학과 졸업(학사)
 1973년 공학박사 학위취득
 1996년~현재 영남대학교 전산공학과 교수 공과대학장

관심분야:Real Time Operating System, Computer Vision