creative commons

COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

[UCI]I804:24011-200000236171

*2008*년 *2*월

석사학위논문

에지 방향성 히스토그램을 이용한 실시간 물체 인식 시스템

조선대학교 대학원

전산통계학과

송 맹 규

에지 방향성 히스토그램을 이용한 실시간 물체 인식 시스템

The Object Recognition System uing
RealTime Orientation Histogram

2008년 2 월 25 일

조선대학교 대학원
전산통계학과
송 맹 규

에지 방향성 히스토그램을 이용한 실시간 물체 인식 시스템

지도교수 이 웅 기

이 논문을 이학 석사학위신청 논문으로 제출함

2007년 10 월

조선대학교 대학원
전산통계학과
송 맹 규

송맹규의 석사학위논문을 인준함

위 원 장 조선대학교 교수 _____ 인

위 원 조선대학교 교수 _____ 인

위 원 조선대학교 교수 _____ 인

2007년 11 월

조선대학교 대학원

목 차

목 차 ····· i
표 목 차 ····· iii
그림목차 ····· iv
Abstract ····· vi

Ⅰ. 서론 ····· 1
1.1 연구 배경 ····· 1
1.2 물체 인식 관련연구 ····· 2
1.2.1 신경회로망을 이용한 물체 인식 ····· 2
1.2.2 지식기반의 물체 인식 ····· 5
1.2.3 속성문법에 의한 물체인식 ····· 7
1.3 논문의 구성 ····· 9

Ⅱ. 전체 시스템 개요 ····· 10

Ⅲ. 전처리 과정 ····· 11
3.1 노이즈 검출 ····· 11
3.1.1 침식(Erosion) 연산 ····· 11
3.1.2 팽창(Dilation) 연산 ····· 15
3.1.3 제거(Opening) 연산 ····· 18
3.1.4 채움(Closing) 연산 ····· 20

Ⅳ 물체 인식 ····· 22
4.1 에지 검출 ····· 23
4.1.1 편미분 연산자 ····· 26
4.1.2 마스크(MASK) ····· 27
4.2 에지 방향성 히스토그램 생성 ····· 30

V 실험 결과 ····· 33
5.1 실험 결과 ····· 33
5.1.1 노이즈 제거 ····· 35
5.1.2 에지 추출 ····· 36
5.1.3 에지 방향성 히스토그램 생성 ····· 37
5.1.4 물체 인식 ····· 38

VI 결론 ····· 39

참고 문헌 ····· 40

표 목 차

[표 3.1] 침식연산 구현 ····· 14
[표 3.2] 팽창연산 구현 ····· 12
[표 3.3] 제거연산 구현 ····· 18
[표 3.4] 채움연산 구현 ····· 20
[표 4.1] 에지검출 결과 ····· 29
[표 4.2] 에지 방향성 히스토그램 생성 구현 ····· 31
[표 5.1] 물체 인식 실험 결과 ····· 38

그 림 목 차

[그림 1-1] 신경회로망에 의한 물체인식시스템의 구성 ····· 2
[그림 1-2] 출력변수의 결정 예 ····· 3
[그림 1-3] 실험의 전체 흐름도 ····· 4
[그림 1-4] 한 개의 모델 클래스가 선정된 경우 ····· 6
[그림 1-5] 한 개의 모델0 클래스가 선택된 경우 ····· 6
[그림 1-6] 인식시스템의 구성도 ····· 3
[그림 2-1] 전체 시스템 ····· 9
[그림 3-1] 침식이 적용된 이미지 ····· 12
[그림 3-2] 원 영상과 침식 영상 ····· 12
[그림 3-3] 침식 마스크 ····· 13
[그림 3-4] 침식 마스크의 연산 방법 ····· 13
[그림 3-5] 원 영상과 침식 연산 후의 결과 영상 ····· 14
[그림 3-6] 팽창 연산의 마스크 ····· 15
[그림 3-7] 팽창이 적용된 이미지 ····· 16
[그림 3-8] 원 영상과 팽창 영상 ····· 16
[그림 3-9] 원 영상과 팽창 연산후의 결과 영상 ····· 16
[그림 3-10] 제거 연산 마스크 ····· 18
[그림 3-11] 원 영상과 제거 영상 ····· 18
[그림 3-12] 제거 영상 실험결과 ····· 19
[그림 3-13] 채움 연산 마스크 ····· 20
[그림 3-14] 원 영상과 채움 영상 ····· 20
[그림 3-15] 채움 연산 실험결과 ····· 21
[그림 4-1] 물체 인식 시스템 ····· 22
[그림 4-2] 에지의 개념 ····· 23
[그림 4-3] 밝기의 변화에 따른 에지의 종류 ····· 24
[그림 4-4] 밝기 변화에 의한 에지 기울기 변화 ····· 25
[그림 4-5] 미분연산자에 의한 에지의 파형 ····· 26
[그림 4-6] 마스크 ····· 27
[그림 4-7] 원 영상에서 3×3 마스크의 진행방향 ····· 28
[그림 4-8] 에지 검출 실험 결과 ····· 29

[그림 4-9] 방향성 히스토그램 ····· 30
[그림 4-10] 에지 방향성 히스토그램 생성 실험결과 ····· 30
[그림 5-1] 모델 물체 영상 ····· 34
[그림 5-2] 물체 인식 시스템에서 모델 영상 Load ····· 35
[그림 5-3] 노이즈 제거 ····· 35
[그림 5-4] 노이즈 제거 실험 결과 ····· 35
[그림 5-5] 에지 추출 ····· 36
[그림 5-6] 에지 추출 실험 결과 ····· 36
[그림 5-7] 에지 방향성 히스토그램 생성 ····· 37
[그림 5-8] 에지 방향성 히스토그램 생성 실험 결과 ····· 37
[그림 5-9] 방향성 히스토그램 ····· 38
[그림 5-10] 물체 인식 실험 결과 ····· 38
[그림 5-11] 인식률 측정 ····· 39
[그림 5-12] 모델 물체의 데이터베이스 ····· 39

ABSTRACT

The Object Recognition System uing RealTime Orientation Histogram

BY Song Maeng Kyu.
Advisor : Prof. Lee Woong Ki.
Department of Computer Science and Statistics,
Graduate School of Chosun University.

In this study of object recognition, The edge of all pixel can be get after elimination of noises on the screen by the edge orientation histogram. And with the threshold value, the decisions are made(follows by decision of edge for distinguishing) whether those pixels are edge or not. After generating the edge orientation histogram on the decided pixels, object recognition should be done in comparison between the edge orientation histogram, driven by mapping on the 36 saving area of all those pixels produced, and model screen.

It's very sensitive to the variation of lights and the size of the subjects because the physical recognition of edge orientation histogram generally depends on the distribution of those pixels. Therefore, it requires a lot of practices, and the video images need to be processed with normalization to make their size even.

The edge orientation histogram use the value of the pixel image so the process is very sensitive to the noises. In this paper, noises has been detected and removed to retain a certain recognition level.

The method, using the edge of the histogram suggested by this these, performs very fast, and produces reliable recognition rate in object recognition.

However, because it is using the value of the pixel, in a complex background conditions, like with objects with color or objects in the video stacking, there is a lot difficulties in recognition. There is a plan to solve these problems and to develop more stable object recognition algorithms in the future.

Ⅰ 서론

1.1 연구 배경

급격하는 컴퓨터 기술의 발전과 더불어 이미지, 그래픽, 오디오, 미디어 등을 활용한 많은 정보들이 쏟아져 나오며, 패턴인식 같은 인간의 지각 능력을 본떠서 만든 프로그램에 의하여 컴퓨터가 도형이나, 문자, 음성 따위를 식별하는 일이나 신경망 이론을 바탕으로 인간의 시각, 청각 등과 유사한 지능적인 처리가 가능하도록 개발하는 컴퓨터 등 다양하게 연구 되고 있다.

산업이 급속하게 발전하는 환경에서 자동화된 물체 인식 시스템의 개발은 생산 라인, 의료 및 군사 등 여러 분야에서 노동 인력을 대처할 수 있는 중요한 위치를 차지하고 있다. 특히, 부품조립 및 검사, 군사장비 등과 같은 물체를 자동적으로 인식하기 위한 연구는 컴퓨터 비전과 로봇비전 분야로 나누어져 있다. 그리고 이러한 연구에서는 컴퓨터에 감지된 물체가 무엇인지를 판단하고, 판단 결과를 이용하여 프로그램 된 대로 적절한 임무를 수행할 수 있는 능력을 제공해 줌으로써 사람이 할 수 있는 시각 능력을 컴퓨터에 부여해 주고자 하는 연구가 진행되고 있다.

일반적인 물체 인식 기법은 인식하고자 하는 물체가 특정 짓는 불변의 특징에 의존한다. 그러나 이러한 물체의 특징은 물체가 다른 물체와 접해 있거나 겹쳐짐에 의해 소실되므로 정확한 인식이 불가능하게 된다. 이러한 특징 소실 현상은 컴퓨터 비전 응용 및 공장 자동화 분야서 중요한 문제로 대두되고 있으며, 특징 소실 문제를 해결하기 위해 중복 물체 인식에 대한 인식 기법이 필요하다.

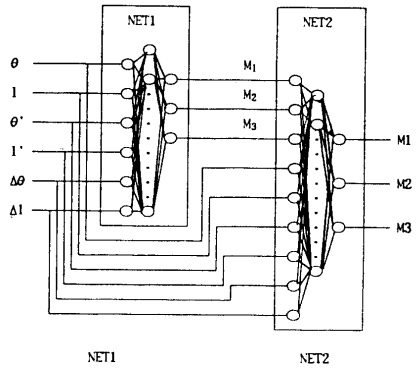
일반적으로 이미지에 대한 많은 특정 정보를 컴퓨터에 저장시켜 이를 기반으로 물체를 인식하는 방법을 이용한다. 이는 이미지에 인식하고자 하는 물체 이미지 특징을 추출하여 데이터베이스에 저장하고, 입력 영상으로부터 추출된 특징들과 모델물체를 비교하여 인색해 내는 방법이다.

이미지에서 물체 인식에는 인식하고자 하는 이미지의 특성과 그 이미지들의 기하학적 연관 관계를 표현하는 특징들의 기술 문제 그리고 복잡한 배경과 물체간의 정합문제 및 겹쳐진 이미지의 인식 등이 가장 중요하게 다루어져야 할 사항이다.

1.2 물체 인식 관련 연구

1.2.1 신경회로망을 이용한 물체 인식

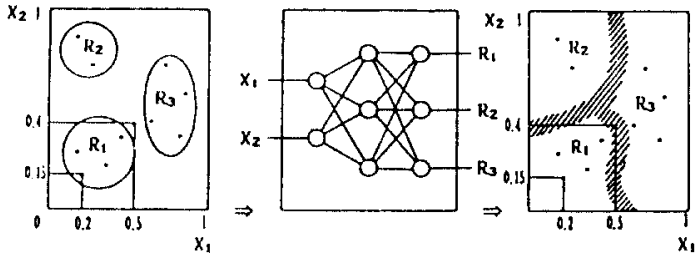
물체를 구성하고 있는 각 선형선소에 대한 국부특징량이므로 각 모델을 구성하고 있는 선형선조들 사이에는 서로 유사한 선형선소들이 다수 존재하게 되어 인식의 애매성이 발생한다. 또한, 유사한 국부특징량들로 이루어져 있기 때문에 각 특징들이 특정 공간상에 혼재해 있게 되어 신경회로망의 학습시 수렴특성이 좋지 못하게 되고, 더불어 신경회로망의 특성을 결정짓는 경계 구분능력의 저하를 가져오게 되어 인식률이 저하되는 문제점이 발생한다. 이 문제를 보완하기 이해서는 신경회로망이 다단연결된 물체인식시스템을 구성하게 한다. 구성된 물체인식 시스템은 2개의 신경회로망으로 이루어져있으며, 첫 번째 회로망 에서는 개략적인 인식을 위한 과정으로서 대국적인 모델 단위의 정보를 출력 하도록 한다. 또한, 두 번째 회로망은 첫 번째 신경회로망의 출력을 입력층의 입력으로 하여, 첫 번째 회로망의 개략적인 인식에서 발생하는 오인식의 영향을 최소화 하고, 보다 향상된 인식 결과를 얻도록 하는 최종인식 과정 회로망이 된다. 그림에서 첫 번째 신경회로망의 입력층에 나타난 출력 M1, M2, M3는 각 모델을 표현한 것이며, 인식된 결과를 나타낸다.



【그림 1-1】 신경회로망에 의한 물체인식시스템의 구성

물체 인식 시스템에 의한 물체인식 과정을 크게 NET1의 출력결정, NET 의 출력 결정으로 나눌 수 있으며, 세부적으로 살펴보면 다음과 같다.

먼저 첫 번째 신경회로망에 의한 모델단위의 출력을 결정한다. 즉, 출력 y에 관련하는 어떤 입력변수 x1,x2,x3…x_{in}을 설정하면, 회로망의 입력중에 주어지는 입력데이터 x1=(x1,x2,x3…x_{in})가 되고, 여기서는 입력변수의 l번째 데이터가 된다. 다음으로 입력변수에 대한 출력데이타를 결정하기 위해 입출력 데이터를 n개로 분할하고 신경회로망에 의해 학습시킨다. 즉 예를 들어 2개의 입력변수에 대한 3개의 출력 변수를 결정하는 NET 1의 출력결정과정 예를 그림1-1에 나타내었으며 다음과 같다.



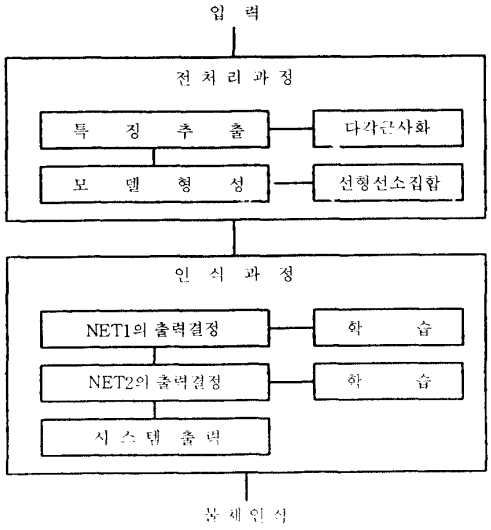
【그림 1-2】 출력변수의 결정 예

먼저 3층 신경회로망의 입력층의 입력데이타 중의 하나를 x_1=(0.2 0.15)라 하면 이 입력데이타를 3개로 하여, (M1, M2, M3)=(1,0,0)으로 함으로서 신경회로망의 입력층과 출력층의 데이터가 결정된다. 다음으로, 결정된 입력데이터와 출력데이터를 각각 신경회로망의 입력층의 입력과 출력층의 교사 데이터로 주어 오차 역진파 학습 알고리즘으로 학습을 행한다. 따라서, 학습이 끝난후 주어진 데이터(학습데이터)이외의 미지 데이터를 신경회로망의 입력층에 넣으면, 미지 데이터에 대한 출력 값이 결정된다.

다음으로, 첫 번째 신경회로망에 의해 결정된 출력 값을 이용하여, 두 번째 신경회로망의 출력을 결정한다. 두 번째 신경회로망에 있어서 입력은 첫 번째 신경 회로망의 출력인 모델 정보와 선형선소의 국부특징량을 동시에 이용한다. 또한 출력을 모델단위의 인식결과를 나타내도록 하고, 첫 번째 과정과 같이 오차 이진파 학습 알고리즘에 의해 학습한 후 미지터에 대한 두 번째 신경회로망의 출력을 결정함으로써 최종인석 결과를 나타내도록 한다.

따라서, 이 과정은 첫 번째 신경회로망의 출력을 두 번째 신경회로망의 입력으로 다시 적용하기 때문에 결과적으로는 입력으로 제시되는 입력패턴에 대한 특징량의 증가를 가져온다. 이것은 인식대상에 대한 특징표현의 다양화를 가져오기 때문에 인식의 효율성을 제공한다. 또한, 다단 결함된 신경회로망의 성능을 살펴보기 위해 각각의 신경회로망에 대한 수렴 특성을 그림1-3에 학습횟수대 오차의 관계로서 나타내었다. 그림에서 알 수 있듯이 NET 1에서의 열화된 수렴특성이 NET 2를 거치면서 향상된 특성을 보이고 있다.

전처리과정은 물체 인식을 위한 특징추출과 모델형성으로 이루어지고, 인식과정은 신경회로망에 의한 물체인식시스템에 의해 이루어진다.



【그림 1-3】 실험의 전체 흐름도

인식실험을 위한 영상의 입력은 그림자 및 물체 표면의 반사와 같은 잡음의 영향을 허용 하기 위해 자연조명하에서 받아들여졌으며, 실험은 IBM PC-AT호환 기종에서 C 언어로 수행되었다. 물체인식은 단독 물체인식과 중복 물체인식으로 나누어 실험하였으며, 물체인식시 물체의 변형에 따른 인식결과를 알아보기 위해 크기 및 형태 변형이 일어난 미지영상에 대해서도 실험하였다. 실험에 사용된 3개의 모델영상은 프라이어, 가위 및 스패너와 같은 공구로서 256X256화소 크기로 구성되었다.

1.1.2 지식기반의 물체인식

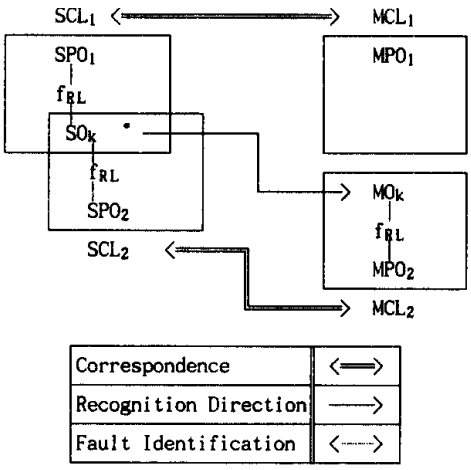
영상물체에 대응되는 모델 물체시각을 찾는 탐색시간은 모델 데이터베이스에 저장되는 모델 물체시각의 수가 많아질수록 그에 비례해서 증가하게 된다. 이와 같은 문제점을 해결하기 위해 모든 물체들을 몇 개의 클래스 및 클래스 으로 분류 한다.

영상물체들을 클래스로 분류하게 되면 영상물체에 가장 잘 대응되는 모델 물체시각을 찾기 위해 모든 모델 물체시각을 찾아 볼 필요 없이, 영상 클래스에 대응하는 모델 클래스내의 모델 물체시각 만율 찾아보면 되기 때문에 영상물체를 빨리 인식 할 수 있다.

영상 클래스를 형성하기 위해서는 먼저 영상에서 prior object를 설정해야 한다. prior objcet의 설정은 각 물체들마다 다른 물체와 인접되어 있는 횟수를 비교하여 그 횟수가 많은 물체에 우선순위를 부여하고, 우선순위가 높은 물체들을 prior object SP_{osc}(1 ≤mc≥m)로 채택하는 방식으로 이루어진다. 영상물체 SPO_{sc}를 인식한 다음 조건을 만족하는 영상물체 SOi들로 영상 클래스 SCL_{sc}를 형성한다.

a. 모델 클래스 수에 따른 영상 클래스 형성 방법

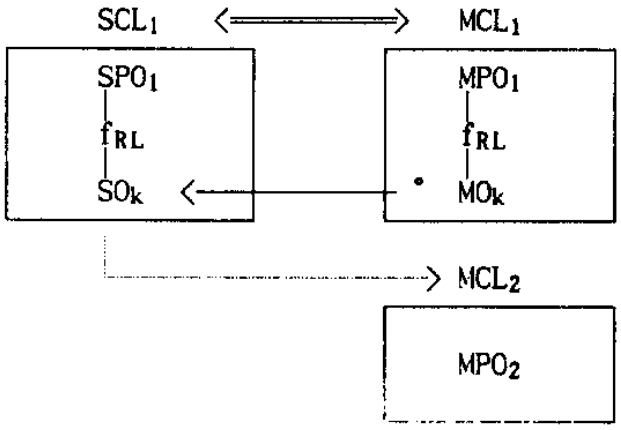
(1) 한 개의 클래스가 선택된 경우



【그림 1-4】 한 개의 모델 클래스가 선정된 경우

위에서와 같이 영상클래스를 형성하고 나면, 영상 클래스 내의 각 물체는 가설과 검증과정을 거쳐서 인식된다. 이때 입력영상에서 선정된 클래스에 속하는 물체들 중 인식 되지 못하는 물체가 있을 수 있다. 즉 영상 물체 SPO_1에 이루는 클래스는 SCL_1이고, 이에 대응되는 모델 클래스는 MCL_1이다.

이때 SCL1에 속하는 영상물체 SOk가 인식 되지 못하는 경우, 만일 SOk가 영상물체 SPO2와도 상대적 위치관계 fRL를 만족한다면 SO는 물체 SOp2가 이루는 클래스 SCL2에 속한다고 보고 인식을 한다. 이와는 달리 물체 SOk가 영상의 모든 prior bbject들과 f_{RL}(SPOsc, SOk) = NULL을 만족하면 SOk는 클래스에 포함되는 물체로 판단 한다.



【그림 1-5】 한 개의 모델 클래스가 선택된 경우

(2) 두 개 이상의 클래스가 선정된 경우

영상의 한 prior objcet SPO_1을 인식할 때 다른 물체에 의한 부분적인 가려짐 등으로 인하여 인식이 부정확해질 수 있는데 이와 같은 경우에는 물체 SPO_1이 두 개 이상의 모델의 prior objcet로 인식 될 수 있다. 이에 따라 두 개 이상의 모델 클래스가 선정되는 경우가 있다. 즉, 영상 클래스 SCL_1에 실제 대응되는 모델 클래스는 MCL_1이지만 MCL_2도 클래스 SCL_1에 대응된다고 잘못 판단할 수 있다. 이와 같은 경우 하나의 모델 클래스를 선택해서 이에 대응되는 클래스를 형성 한다.

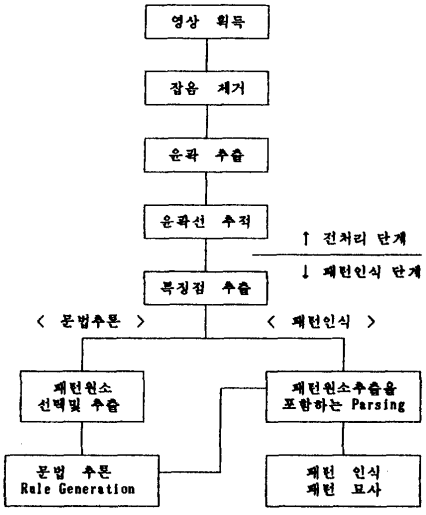
(3) 모델 클래스의 설정이 잘못된 경우

입력 영상의 한 클래스가 대한 모델 클래스가 잘못 설정 되는 경우가 발생할 수 있다. 즉, 한 클래스 내에서 인식되지 못한 물체가 입력영상의 다른 prior object와 상대적 위치관계를 가지고 있지 못하면서 클래스 에 속하는 물체들도 인식되지 못하는 경우를 말하는데. 이때에는 모델 클래스의 설정을 다시 하게 된다.

1.1.3 속성문법에 의한 물체 인식

구론적 인식방법은 1950년에 이미 Chomsky에 의해 정립된 형식언어 이론을 바탕으로 1960년대 이후 급속한 발전을 이른 방법으로, 패턴의 구조적인 면에 중점을 두어 패턴원소로 나타내고, 이 패턴원소들의 조합으로 표현된 전체 패턴을 생성규칙에 따라 분석, 파악함으로써 패턴을 인식을 한다. 그러나 이 방법은 잡음이 있는 패턴을 다루는데 약하고, 수치적 정보를 이용하지 못하므로 크기가 다르거나, 방향이 틀린 패턴의 인식에 있어서 매우 많이 생성규칙을 필요로 하게 되어 복잡성을 배제할 수 없다.

이상과 같이 물체인식에 있어서 어느 한 방법에 의한 경우의 복잡성 및 모호성을 배제하기 위해 이 두 가지의 접근방법을 결합하여 종합적인 패턴인식을 하기위한 연구가 1970년대 말에부터 시작되었다.



【그림 1-6】 인식시스템의 구성도

패턴을 구성하는 각 패턴원소에 크기를 갖는 속성을 부여하고 속성에 대해서는 결정론적 인식방법을, 패턴구조에 대해서는 구론적 인식 방법을 적용한 속성문법을 도입하여 종합적인 패턴인식과 상태묘사를 할 수 있도록 하였다.

속성분법이 강체로 된 물체가 조정되어 변화되었거나, 크기가 다른 경우 및 방향이 달라진 경우에 대한 효율적인 이식 방법을 제공하였고 우선적인 판단기준을 설정하기 위한 Rale generation 및 패턴 원소 추출을 포함하는 Parsing 방법을 도입하여 인식시스템을 효율성을 증대시킬 수 있다.

1.3 논문의 구성

본 논문은 자연 영상에서 실시간으로 물체를 인식하기 위해서 에지 방향성 히스토그램 알고리즘을 적용하였고, 에지 방향성 히스토그램을 생성하기 위해서 영상으로부터 노이즈 제거 및 에지 추출과 에지 방향성 히스토그램을 생성 하여 모델 물체와 비교하여 효과적인 물체 인식을 고찰하였다.

본 논문은 다음과 같은 구성되었다.

1장에서는 연구배경, 물체 인식 관련연구, 논문의 구성을 설명하였다.

2장에서는 전체 시스템의 개요를 설명하였다.

3장에서는 물체 영상의 정규화를 위한 노이즈 검출을 설명하였다.

4장에서는 물체 인식을 위한 에지 검출, 에지 방향성 히스토그램 생성을 설명하였다.

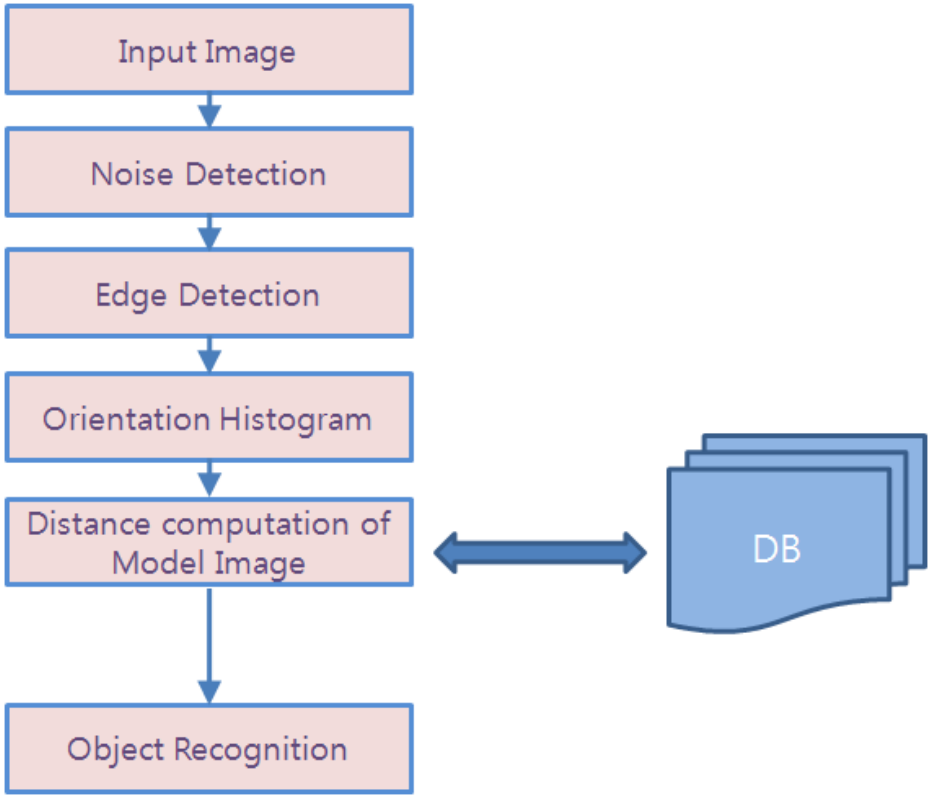
5장에서는 본 논문의 실험 결과를 기술하였다.

마지막으로, 6장에서는 본 논문의 결론에 대하여 기술하였다.

II 전체 시스템 개요

본 논문에서 제안하는 물체 인식 시스템은 이미지 영상에서 물체만을 인식하도록 수행한다.

첫 번째로 입력 받은 이미지를 노이즈를 제거하고, 두 번째는 노이즈를 제거한 이미지의 에지를 검출한다. 세 번째로는 영상의 히스토그램 평활화를 이용하는 단계이고, 네 번째는 영상을 정규화 하고, 마지막에는 물체를 인식한다.



[그림 2-1] 전체 시스템

전처리에 해당하는 부분은 물체를 인식하기 위해 영상에서 노이즈 제거한다. 인식 부분에서는 노이즈의 제거한 영상의 에지를 구하여 에지 방향성 히스토그램을 생성하여 데이터베이스의 물체를 비교하여 물체 인식을 수행 한다.

III 물체 영상의 정규화

3.1 노이즈 검출

그레이 영상에서의 노이즈 검출은 모폴로지 기법을 이용한다. 모폴로지는 영상 전처리 작업이나 초기 객체분류 등 또는 이러한 처리 뒤에 이어서 물체의 내재된 구조를 명확히 하는데 이용된다. 모폴로지 기법에는 침식, 팽창, 제거, 채움 4가지로 나눠지는데 독립적으로 사용하여 효과를 얻을 수도 있지만 대부분의 영상처리 기법에서와 마찬가지로 4가지의 방법을 조합하여 사용하는 것이 더욱 효과적일 경우가 많다.

3.1.1 침식(Erosion) 연산

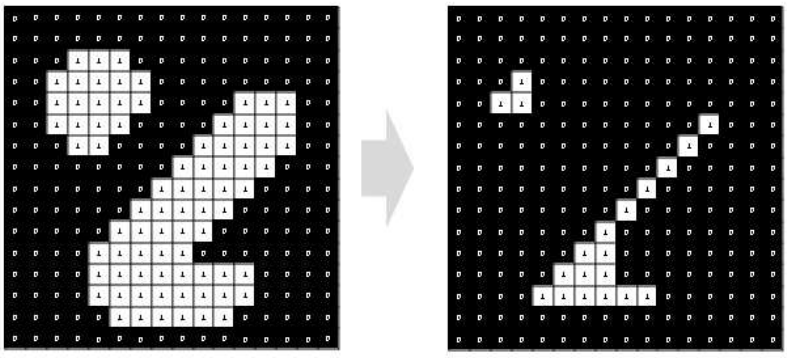
침식 연산은 배경에 대한 배경을 확장시키고 물체의 크기를 축소하는 역할을 한다. 침식 연산은 주로 물체와 배경사이의 스파크 잡음 제거와 같이, 전체 영상에서 아주 작은 물체를 제거하거나 또는 전체 영상에서 배경확장에 따른 물체를 축소하는 역할을 한다. Z^2 내의 집합 A와 B에 대해, B에 의한 A의 침식은 [식3-1]과 같이 정의 할 수 있다.

$$A \ominus B = \{x \mid (B)x \subseteq A\} \qquad \text{[식 3-1]}$$

[그림 3-1]에서는 M이라는 글자를 볼 수 있을 것이다. 자세히 보면 M 주변에 점선이 보이는데 이점선이 원래의 M 글자 부분이다. [그림 3-1]에서는 원래 점선 크기의 글자 M의 주변 픽셀이 삭제되어 형태적으로 축소된 결과를 보여준다. 다시 말하면 최외각의 한 픽셀만큼 없어 진 것으로 될 것이다.

좀더 구체적으로 살펴보면 [그림 3-2]에서는 원 영상과 침식된 영상을 0(검정색)과 1(흰색)으로 표현한 것으로서 2진 형태의 영상임을 알 수 있다. 원 영상에서 침식작용이 되었을 때는 영상의 최 외각 한 픽셀의 흰색(1)부분이 검정색(0)으로 대치되었음을 알 수 있다.

[그림 3-1] 침식이 적용된 이미지



[그림 3-2] 원 영상과 침식 영상

침식 연산을 프로그램으로 하려면 매번 수식에 의해서 침식연산을 수행하는 것도 좋지만 마스크 연산을 사용하는 것이 더 쉬운 방법이다.

본 논문에서 사용할 마스크는 3×3의 마스크로 물체의 명도 값을 255로 설정하고 배경의 명도 값을 0으로 설정할 경우 [그림 3-3]과 같은 구조를 갖는다.

마스크의 연산방법은 침식 마스크가 입력 영상 위를 차례로 지나갈 때의 효과를 생각해 보자. 3×3 마스크가 영상 위를 지나갈 때 원 영상의 3×3씩 계산이 이루어지는데 원 영상의 3×3의 픽셀 값과 3×3 마스크의 픽셀 값이 정확히 일치하지 않으면 결과 영상의 다시 할당되는 가운데 명도 값은 0(검정색)이 들어가 물체의 축소가 일어난다. 마찬가지로 마스크와 원 영상의 3×3 블록과의 픽셀을 비교한 것이 정확히 일치할 경우에는 255(흰색)가 할당된다.

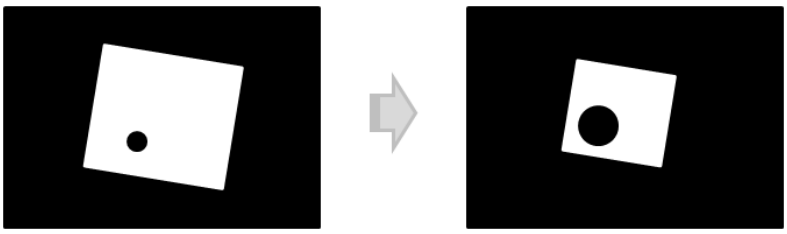
255	255	255
255	255	255
255	255	255

[그림 3-3] 침식 마스크

[그림 3-4]에서는 침식 기법을 그림으로 나타낸 것으로 침식 마스크와 원 영상의 3×3 부분 영역과의 비교 후 새로운 값을 할당하는 마스크 연산 방법을 설명해 주고 있다.

[그림 3-4] 침식 마스크의 연산 방법

침식 영상의 몇 가지 예를 들어보면 [그림 3-5]는 간단한 영상으로 침식 연산의 특징을 더욱 잘 나타내주는 그림이며 배경색인 어두운 부분이 강조되어 진다는 점이다. 침식연산의 응용은 매우 다양하며 알고리즘의 수정으로 더욱 좋은 결과를 얻을 수 있다.



[그림 3-5] 원 영상과 침식 연산 후의 결과 영상

[표 3-1]은 본 논문에서 구현한 침식 연산 구현한 프로그램 소스입니다.

```
void CPCADoc::Erosion()
{
unsigned char temp[3][3]={0,0,0,0,0,0,0,0,0};
unsigned char mask[3][3]={0,0,0,0,0,0,0,0,0}; //마스크
int sum=0,x=0,y=0,min=1000;

  for( y=0;y<256;y++)
  {
        for( x=0;x<256;x++)
        {
                for(int row=0;row<3;row++)
                {
                        for(int column=0;column<3;column++)
                        {
                        temp[row][column]=m_OpenImg[row+y][column+x];
                        }
                } //이미지 분리.3x3

                        for(int i=0;i<3;i++)
                        {
                                for(int j=0;j<3;j++)
                                {
                                sum=temp[i][j]+mask[i][j]; //픽셀단위 더하기
```

```
                                if(min>sum)min=sum;//minimum value 픽셀찾기
                                }
                        }
                        m_ResultImg[1+y][1+x]=min;
                        sum=0;
                        min=1000;
                }
        }
}
```

[표 3-1] 침식 연산 구현

3.1.2 팽창(Dilation) 연산

팽창 연산은 물체의 최 외각 픽셀을 확장시키는 역할을 한다. 그러므로 물체의 크기는 확장되고 배경은 축소된다. 팽창 연산은 물체 안의 홀과 같은 빈 공간을 메우는 역할을 하거나 서로 짧은 거리만큼 끊어진 영역을 연결시켜준다. [그림 3-6]은 팽창 연산 마스크를 나타낸다.

0	0	0
0	0	0
0	0	0

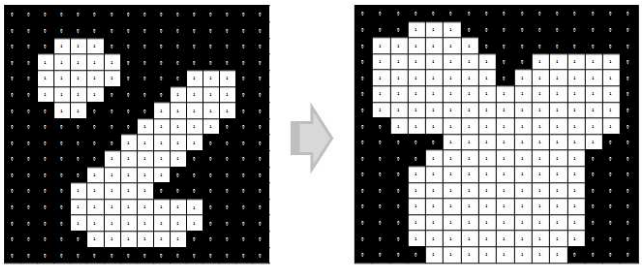
[그림 3-6] 팽창 연산의 마스크

이 마스크는 흰색 물체의 둘레에 한 픽셀을 더하는 역할을 하는데 3×3 마스크와 똑같은 영역을 가지는 영역에 대하서는 값을 바꾸지 않으며 1개 이상의 서로 픽셀이 존재할 경우 마스크의 가운데 픽셀에 흰색 값을 할당하는 역할을 한다.

팽창 연속을 계속하게 되면 흰색 값을 가진 물체의 확장으로 배경 영역이 없어지게 된다. [그림 3-7]은 최 외각 픽셀이 확장된 팽창 영상을 나타내고 있다.

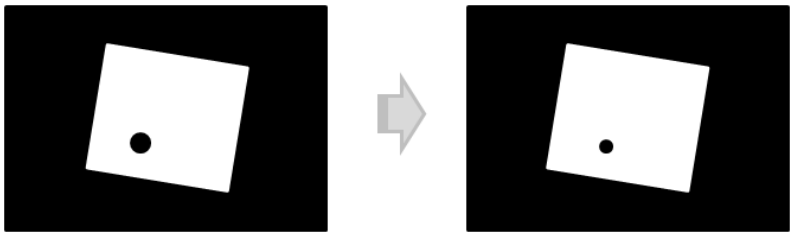
[그림 3-7] 팽창이 적용된 이미지

또한 [그림 3-8]은 검정색(0) 배경과 흰색(1) 물체를 표현하여 팽창 연산의 예를 보여주고 있다. 이것은 침식 연산과 매우 잘 비교되는 그림이라 할 수 있을 것이다. 이 그림은 떨어져 있던 물체를 붙여주는 역할을 하는 팽창 연산의 특징을 나타내고 있다. 물론 너무 멀리 떨어져있는 물체끼리는 붙지 않겠지만 가까이 있는 물체는 흰색 픽셀의 확장으로 두 영역이 연결된다.



[그림 3-8] 원 영상과 팽창 영상

[그림 3-9]은 사각형 물체의 흰 픽셀이 확장되어 물체 안이 검정색 배경이 축소됨을 보여준다.



[그림 3-9] 원 영상과 팽창 연산후의 결과 영상

[표 3-2]은 본 논문에서 구현한 침식 연산 구현한 프로그램 소스입니다.

```
void CPCADoc::Dilation()
{
        unsigned char temp[3][3]={0,0,0,0,0,0,0,0,0};
        unsigned char mask[3][3]={0,0,0,0,0,0,0,0,0};
        int sum=0,x=0,y=0,max=(-1000);
        for( y=0;y<256;y++)
        {
                for( x=0;x<256;x++)
                {
                        for(int row=0;row<3;row++)
                        {
                                for(int column=0;column<3;column++)
                                {
                        temp[row][column]=m_OpenImg[row+y][column+x];
                                }
                        }//이미지 분리.3x3
                        for(int i=0;i<3;i++)
                        {
                                for(int j=0;j<3;j++){
                                        sum=temp[i][j]+mask[i][j];
                                        if(max<sum)max=sum;
                                }
                        }
                        m_ResultImg[1+y][1+x]=max;
                        sum=0;
                        max=(-1000);
                }
        }
}
```

[표 3-2] 팽창 연산 구현

3.1.3 제거(Opening) 연산

제거 연산은 침식 연산 다음에 팽창연산을 바로 사용하는 알고리즘이다. 즉, 원영상을 침식 연산으로 영상의 최 외각을 한 픽셀씩 없애고 다시 팽창 연산으로 최 외각을 한 픽셀씩 확장시키는 방법이다.

언뜻 보기에 한 픽셀씩 없애고 다시 확장시키면 원래의 영상이 될 것 같지만 전혀 그렇지 않다. 침식 연산을 사용한 영상은 원 영상의 크기를 잃어버리는 단점을 가지고 있지만 제거 연산은 침식 연산으로 미세한 잡음을 제거하고 다시 팽창 연산을 수행함으로 원 영상의 크기를 유지할 수 있다.

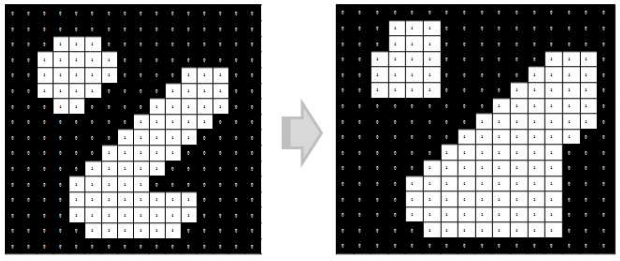
제거 연산의 마스크는 침식 연산과 팽창 연산의 마스크를 번갈아 적용하면 되는데 알고리즘 역시 두 연산자를 차례로 수행한 것과 동일하다. [그림 3-10]은 제거 연산 마스크를 나타낸 것이다.

1	1	1
1	1	1
1	1	1

0	0	0
0	0	0
0	0	0

[그림 3-10] 제거 연산 마스크

다음으로 제거연산을 0과1의 조합으로 표현한 [그림 3-11]은 침식과 팽창한 후의 모습을 예상할 수 있게 해준다.

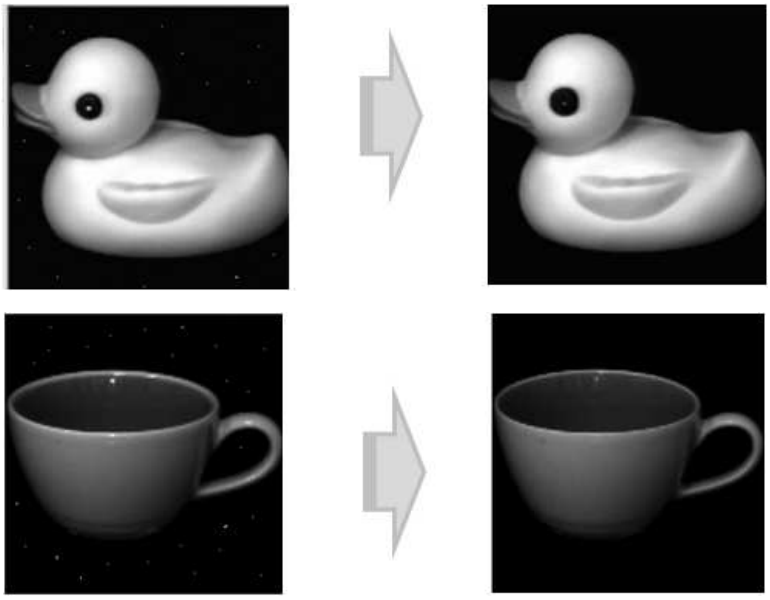


[그림 3-11] 원 영상과 제거 영상

[표 3-3]은 본 논문에서 구현한 침식 연산 구현한 프로그램 소스와 [그림 3-12]은 제거 연산 실험 결과 입니다.

```
void CPCADoc::Opening()
{
        Erosion();
        for(int y=0; y<256; y++)
        {
                for(int x=0; x<256; x++)
                {
                        m_OpenImg[y][x]=m_ResultImg[y][x];
                }
        }
        Dilation();
}
```

[표 3-3] 제거 연산 구현



[그림 3-12] 제거 영상 실험결과

3.1.4 채움(Closing) 연산

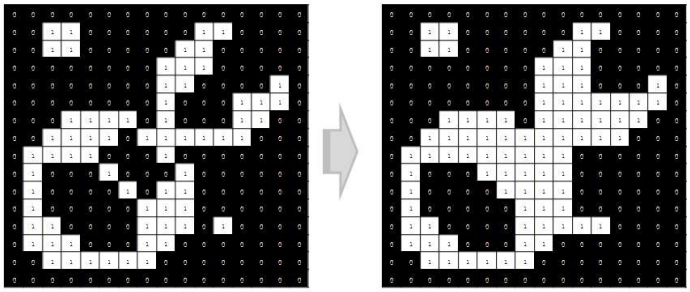
채움 연산은 제거 연산의 반대의 기능을 가지고 있다. 물론 침식 연산과 팽창 연산을 적당히 조합하여 쓰는 것은 같지만 조합순서가 제거 연산과 반대이다. 채움 연산은 팽창 연산으로 물체의 확장을 수행한 뒤 침식 연산으로 다시 축소 연산을 행한다.

이 채움 연산은 확장이 먼저 일어나므로 작은 홀들은 메꾸어 지고 축소 작용 때에 원 영상의 크기로 복원된다. 채움 마스크는 [그림 3-13]와 같다. 그리고 [그림 3-14]은 0과 1로 나타낸 채움 연산의 보기이다. 마찬가지로 형태를 유지한 팽창과 침식의 결과를 보여준다.

	0	

1	1	1
1	1	1
1	1	1

[그림 3-13] 채움 연산 마스크



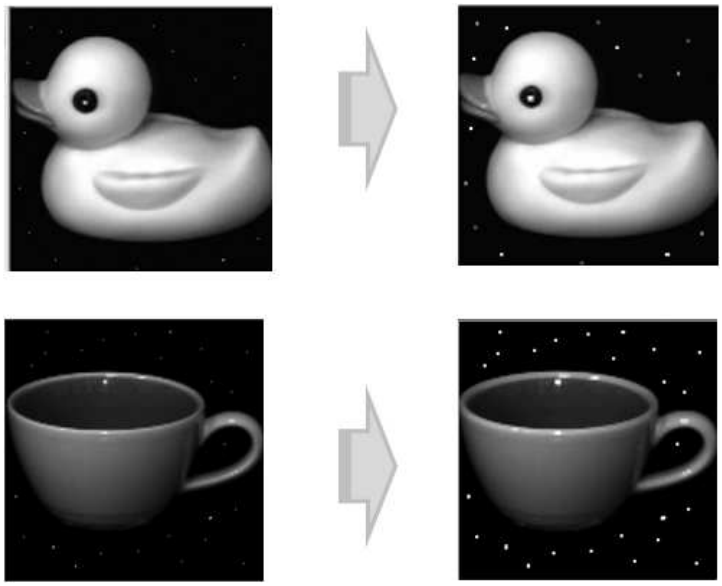
[그림 3-14] 원 영상과 채움 영상

[표 3-4]은 본 논문에서 구현한 침식 연산 구현한 프로그램 소스와 [그림 3-15] 제거연산 실험 결과입니다.

```
void CPCADoc::Closing()
{
```

```
        Dilation();
        for(int y=0; y<256; y++)
        {
                for(int x=0; x<256; x++)
                {
                        m_OpenImg[y][x]=m_ResultImg[y][x];
                }
        }
        Erosion();
}
```

[표 3-4] 채움 연산 구현

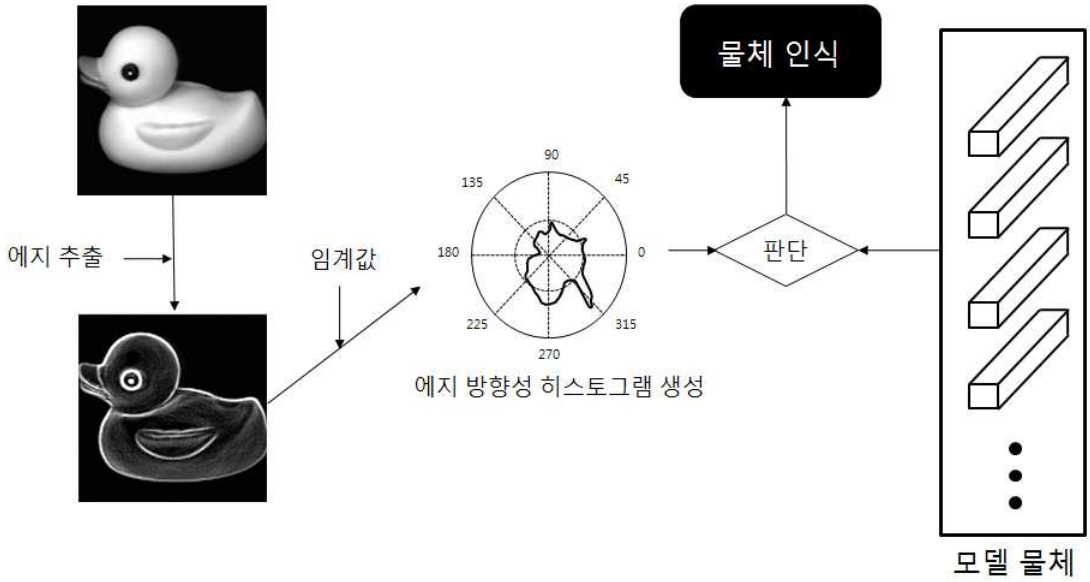


[그림 3-15] 채움 연산 실험결과

4. 물체 인식

영상으로부터 특정한 물체를 분리하는 일은 컴퓨터 비전이나 인식 등 많은 분야에서 기본적인 부분이면서도 중요하다. 그러나 배경 이미지 영상에서 물체의 형상을 추출하여 산술적 차이만을 비교하여 물체의 형상을 인식을 하는 것은 쉽지 않은 일이다. 본 논문에서는 이미지 영상에서 물체 인식하기 위해 간단하고 빠른 알고리즘(에지 방향성 히스토그램)을 제안 했다.

이미 입력된 이미지 영상은 전처리 과정을 통해 모폴로지 기법을 이용하여 제거 연산(Opening) 연산으로 노이즈를 제거하였다. 이번 물체 인식 단계에서는 편미분 연산을 통해서 물체의 에지를 검출하여 에지 방향성 히스토그램을 생성하고, 추출된 물체 영역을 모델 물체와 비교, 판단하여 물체를 인식하는 단계이다.



[그림 4-1] 물체 인식 시스템

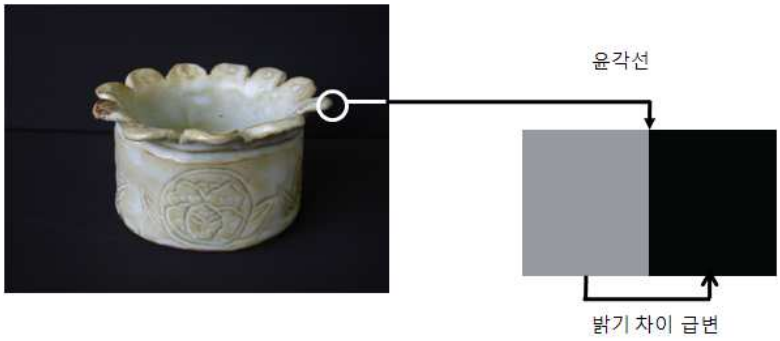
입력 영상에서 특별한 물체를 대표하는 특정 벡터 형성하기 위하여 임계값을 이용하여 방향성 히스토그램을 생성하고 모델 물체와 비교하여 판단하여 물체를 인식한다. [그림 4-1]은 본 논문에서 제안한 물체 인식 시스템을 보여주고 있다.

4.1 에지 검출

에지는 영상 안에서 영역의 경계(예를 들면 대상물과 배경)를 나타내는 특징으로 픽셀 밝기의 불연속 점(픽셀의 밝기가 갑작스럽게 변하는 점)을 나타낸다. 예를 들자면 증명사진에서 사람의 얼굴과 뒷 배경과의 경계선은 픽셀들의 밝기가 갑작스럽게 변하게 되는데 이를 에지라고 말한다. 에지는 영상 안에서 있는 물체의 윤곽에 대응되며 많은 정보를 가지고 있고 물체의 위치, 모양, 크기, 표면의 무늬 등에 대한 정보를 알려준다. 어린 시절에 동전 백원을 종이 밑에 깔고 연필로 신나게 긁으면 종이 위에 대략적인 백원의 모양인 할아버지의 얼굴이 나타났다. 그때의 에지만 으로도 물체의 위치나 모양, 크기 등을 알 수 있었던 것과 같은 원리라고 생각해도 틀리지는 않을 것이다.

우리들이 보는 물체의 정보, 즉 저 물체가 어떻게 생겼는가는 그 물체의 안쪽 면을 보는 것이 아니고 그렇다고 물체의 바깥쪽 면만 보는 것도 아니다. 그것은 바로 물체의 안쪽 면과 물체의 바깥쪽 면 사이에 있는 것은 빛이 반사하는 정도가 물체와 배경과의 경계를 사이에 두고 밝기 차가 눈에 띄게 나타나기 때문이다.

이 밝기 차로 우리는 물체와 배경을 구분하게 된다. 만약에 아주 어두운 곳에서는 물체의 경계가 없어지므로 우리는 물체와 배경을 인식하지 못하는 것이다. [그림 4-2]은 이러한 픽셀의 급작스러운 밝기변화에 의한 에지의 예를 나타낸 것이다.



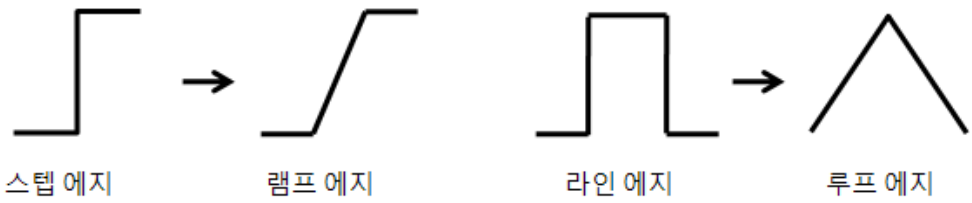
[그림 4-2] 에지의 개념

앞에서 언급했듯이 에지는 영상 안에서 상당한 밝기의 차이가 있는 곳이고 이것은 대개 물체의 경계에 해당하는 곳이므로 대개 픽셀 값의 불연속이나 픽셀 미분 값의 불연속 점에 존재한다. 영상에서의 불연속 점은 다음과 같이 분류할 수 있다

- 스텝 불연속 점 : 영상의 밝기가 갑자기 변하는 곳
- 라인 불연속 점 : 영상의 밝기가 갑자기 변화하나 조금 지나면 다시 돌아오는 곳

스텝 불연속 점에 해당하는 에지를 스텝 에지(step edge)라고 하고, 라인 불연속 점에 해당하는 연산을 라인 에지(line edge)라고 한다.

보통 에지 검출 연산을 하기 전에 잡음을 제거하기 위하여 잡음제거 연산을 시행하는 경우가 많은데(예를 들어 Canny filter가 에지 추출 전 잡음제거의 예가 될 수 있다. 먼저 잡음을 제거하기 위하여 잡음제거 필터를 수행한 뒤 에지를 검출하는 순서를 따른다.) 이럴 경우 잡음제거 연산의 영향으로 픽셀의 값의 변화가 둔화된다. 즉 픽셀 값의 급격한 변화가 완만한 변화로 바뀌게 되는데 이러한 경우 스텝 에지는 램프 에지(ramp edge)로 변하고 라인 에지는 루프 에지(roof edge)로 변한다. 그림에서 이러한 에지들을 일차원적인 모습으로 보여주고 있는데 밝기 변화의 차이를 쉽게 느낄 수 있다. 이들의 밝기는 모두 일정한 거리에 걸쳐서 변화된다. 특별히 아주 작은 잡음과 같은 에지들을 사전에 미리 제거함으로써 원하고자하는 물체의 큰 윤곽을 알아내는 곳에 자주 사용되어진다.



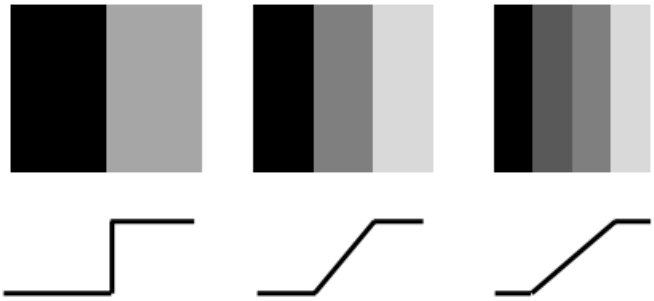
[그림 4-3] 밝기의 변화에 따른 에지의 종류

에지에 해당하는 픽셀을 구하는 방법을 에지 검출이라고 하며, 현재까지 여러 방법이 제안되어 있다. 에지 검출 기법도 에지의 종류에 따라 달라진다. 각 에지 검출 방법은 나름대로 장단점을 가지고 있으며 어떤 에지 검출법은 한 응용 분야에서는 좋은 결과를 나타내지만 다른 응용에서는 좋지 않은 결과를 나타낼 경우도 있다.

간단히 예를 들면, 어떤 오래된 사진이 있는데 사람얼굴뿐 아니라 먼지 같이 생긴 잡

음이 많이 있으면 보통은 이런 사진에서 에지만 추출하는 경우는 드물지만 잡음을 없앤 뒤 에지를 구한 다음 원래 영상에 더하면 사진이 좀 더 확실히 나타나는 경우 가 있다. 모든 영상처리가 마찬가지이지만 이론은 있을지 몰라도 다른 애플리케이션 개발과는 달리 처리할 대상마다 기법을 변형하고 응용하여 좀 더 좋은 영상을 얻게 된다.

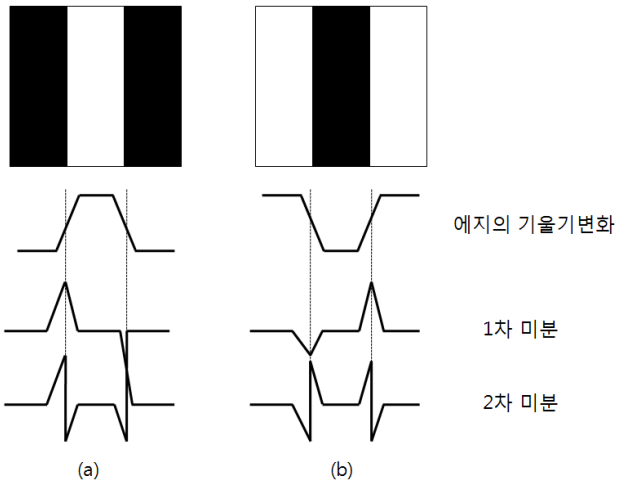
아래의 [그림 4-4]은 밝기 변화에 의한 에지의 기울기를 나타내고 있다. 급격한 밝기 변화를 보이는 경우 기울기 값이 크며 점차적으로 변하는 경우 기울기가 완만해짐을 대략적으로 나타내어 주는 그림이다.



[그림 4-4] 밝기 변화에 의한 에지의 기울기 변화

그러면 에지를 검출하기 위해서는 편미분 연산자 계산에 근거한다. [그림 4-5]은 앞에서 살펴본 [그림 4-4]과 비슷한 그림인데 밝기변화에 의한 에지의 기울기 변화량 그래프에서 1차 미분한 결과와 2차 미분환 결과를 살펴볼 수 있는데 [그림 4-5]의 (a)는 어두운 배경에 밝은 줄무늬 영상과, 이 영상의 수평 주사선을 따라 나타낸 밝기값인 윤곽 파형 그래프와 이 파형을 1차, 2차 미분한 값을 위에서 아래로 나타내고 있다.

[그림 4-5]의 (a)에서 밝기변화 값의 파형을 1차 미분한 그래프를 살펴보면 밝기변화 파형의 앞부분 기울기가 양이므로 미분한 값 역시 음의 값을 나타내고 있다. 또한 일정하게 밝기의 변화가 없는 수평 부분에서는 0의 값을 나타내고 있다. 2차 미분의 파형 그래프를 살펴보면 에지의 어두운 부분에 속하는 변이 부분에 대해서는 양의 값을, 에지의 밝은 부분에 속하는 변이 부분은 음의 값을 나타낸다. 마찬가지로 일정한 밝기 값을 가진 영역에서는 0의 값을 보이고 있다. 이것은 다음과 같은 의미를 나타낸다. [그림 4-5]의 (b)역시 마찬가지로 생각하면 될 것이다.



[그림 4-5] 미분연산자에 의한 에지의 파형

4.1.1 편미분 연산자

영상에서의 그라디언트의 크기를 구하면 에지를 구할 수 있게 된다. 에지는 영상의 밝기 값이 급격히 변화하는 부분에 존재하므로 그라디언트의 크기가 큰 곳이 에지라고 말 할 수 있다. 1차 미분 또는 2차 미분 파형에서 밝기의 차가 급격히 변화하는 곳에서 그라디언트의 크기가 크게 나타난다. 그라디언트의 크기를 구하는 방법은 다음과 같다.

① 먼저 기울기 벡터를 구한다.

$$\nabla F = \begin{vmatrix} Gx \\ Gy \end{vmatrix} = \begin{vmatrix} \frac{\delta f}{\delta x} \\ \frac{\delta f}{\delta y} \end{vmatrix} \qquad \text{[식 4-1]}$$

② 이 벡터의 크기를 구한다

$$\nabla f = mag(\nabla F) = [G_x^2 + G_y^2]^{1/2} \qquad \text{[식 4-2]}$$

이 식은 계산 량도 적고 직관적인 판독이 가능하다. 그라디언트의 방향 역시 중요한 값인데 방향값은 영상 안의 여러 개의 에지로 구분되어진다. 그라디언트의 방향값은 [식 4-3]에 의해서 구해진다.

$$\alpha(x,y) = \tan^{-1}\left(\frac{G_y}{G_x}\right) \qquad \text{[식 4-3]}$$

[식 4-2]과 [식 4-3]에서 주의해야할 점은 영상의 특징은 좌표평면인 2차원 형태를 갖기 때문에 미분도 x축과 y축으로 각각 미분해 주어야 한다. 물론 신호처리와 같은 영역은 1차원이므로 한번만 행하여도 만족한다.

4.1.2 마스크(MASK)

에지 검출 방법에는 여러 가지가 있지만 미분 연산자의 역할을 해주는 여러 가지 마스크들이 나와 있다. 이러한 마스크는 [식 4-2]과 [식 4-3]와 같은 수학적인 조건을 만족하며 동일한 효과를 갖는다.

마스크란 영상 안에서 일정 부분에 위치시키기 위한 일종의 행렬 모양의 구조체 이다. 주로 3×3, 5×5, 16×16 등과 같은 정방 행렬을 많이 쓰는데 [그림 4-6]에서 3×3 마스크의 예를 보여 주고 있다. 하나의 좌표 값은 하나의 픽셀을 나타고 있다.

(-1,-1)	(-1,0)	(-1,1)
(0,-1)	(1,1)	(0,1)
(1,-1)	(1,0)	(1,1)

[그림 4-6] 마스크

본 논문에서 3×3 마스크를 이용한 방법은 원 영상에 이 마스크를 겹쳐서 왼쪽부터 오른쪽으로 위에서 아래 방향으로 진행하면서 계산한 뒤 계산된 값을 (0,0)자리인 중심에다가 다시 할당함으로써 출력영상을 얻는다. [그림 4-7]에서 진행방법을 나타내고 있다.

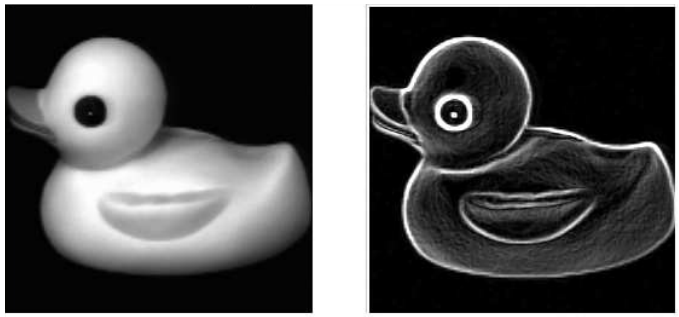
[그림 4-7] 원 영상에서 3×3 마스크의 진행방향

[표 4-1]은 본 논문에서 에지 검출을 구현한 프로그램 소스와 [그림 4-8] 에지 검출한 실험 결과입니다.

```
void CPCADoc::Sobel()
{

        int centerValue1=0;
        int centerValue2=0;
        int sum=0;
        unsigned char m_tmpImg[258][258];

        int mask1[3][3]={ -1,0,1,
                          -2,0,2,
                       -1,0,1};
        int mask2[3][3]={ 1,2,1,
                          0,0,0,
                    -1,-2,-1};
```

```
        for(int k=1; k<257; k++){
                for(int l=1; l<257; l++){
                        m_tmpImg[k][l]=m_OpenImg[k-1][l-1];
                }
        }
        for(int row=0; row<256; row++){
                for(int column=0; column<256; column++){
                        for(int i=0; i<3; i++){
                                for(int j=0; j<3; j++){
                centerValue1 += m_tmpImg[i+row][j+column]*mask1[i][j];
                centerValue2 += m_tmpImg[i+row][j+column]*mask2[i][j];
                                }
                        }
                        sum=abs(centerValue1)+abs(centerValue2);
                        if(sum>255)sum=255;
                        m_ResultImg[row+1][column+1]=(unsigned char)sum;
                        centerValue1=0;
                        centerValue2=0;
                        sum=0;
                }
        }
}
```

[표 4-1] 에지 검출 구현



[그림 4-8] 에지 검출 실험 결과

4.2 에지 방향성 히스토그램 생성

에지 방향성 히스토그램을 이용하면 이미지의 특정 정보로 사용할 수 있다. 에지 방향성 히스토그램을 생성하기 위해서는 먼저 에지를 추출해야 하는데 이미 전 단계에서 이미지 영상에 대한 에지를 추출하였다. 이미지로부터 추출한 에지를 가지고 에지의 방향성을 이용하여 에지 방향성 히스토그램을 생성한다.

이미지 영상 안에 있는 에지를 추출하게 되면 실질적으로 에지가 아닌 부분에서도 에지값을 갖게 된다. 따라서 특정 픽셀이 에지인지 아닌지를 판별하는 과정이 필요하다. 특정 픽셀이 에지인지 아닌지를 판별하기 위해서 임계치에 따라 에지로 판정된 픽셀만을 대상으로 16단계의 방향으로 에지 방향성 히스토그램을 [식 4-5]에 의해 생성한다.

$$direction = \tan^{-1}\frac{s_y(i.j)}{s_x(i,j)} * \frac{36}{\pi} \qquad \text{[식 4-4]}$$

$$\theta(x,y)\arctan[I(x,y)-I(x-1,y), I(x,y)-I(x,y-1)] \qquad \text{[식 4-5]}$$

임계값은 객체와 배경을 구분해 주는 중요한 역할을 수행한다. 임계값이 너무 높게 결정되면 물체에 관한 정보를 잃어버리게 되고 너무 낮게 주어지면 배경의 일부분이 객체 영역으로 판단되는 오류를 갖게 된다.

[그림 4-9]은 임계값(T_h)에 따른 영상 변화의 예를 보여주고 있다. (b)의 경우 임계값을 너무 낮게 주어 배경 부분도 움직임이 존재하는 객체 영역으로 판단되었으며, (c)의 경우는 임계값이 너무 높아 움직임 정보를 잃어버리게 되어 객체 영역이 너무 작게 표현되었다. (d)는 적절한 임계값에 의해 비교적 객체 영역을 잘 구분하고 있다.

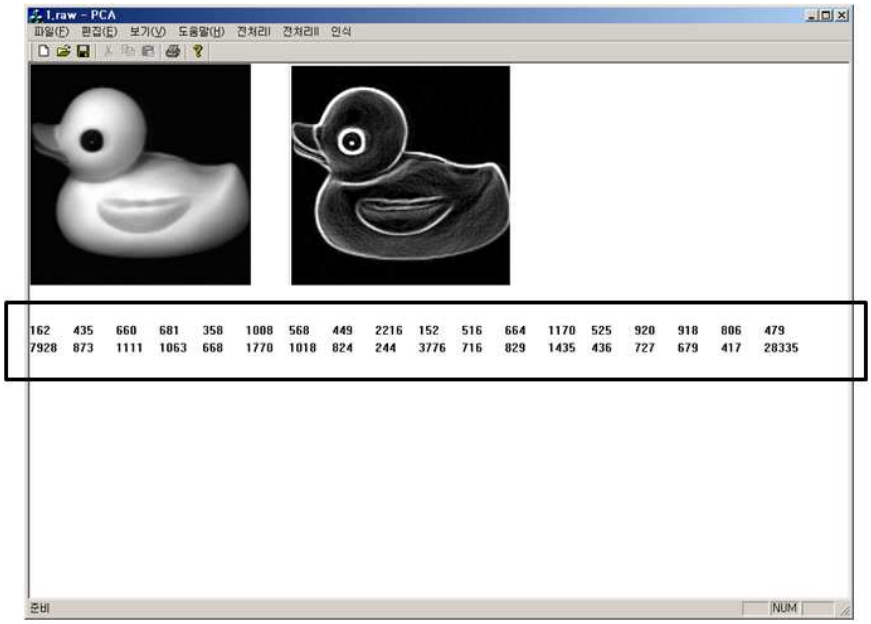
(a) 입력 영상 (b) T_h=25 (c) T_h=84 (d) T_h=42

[그림 4-9] 임계값에 따른 에지 영상

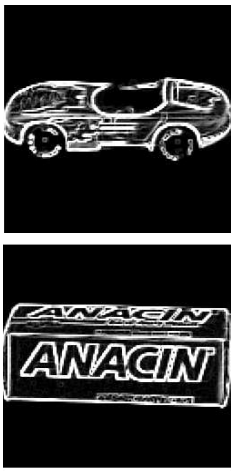
[표 4-2]은 본 논문에서 에지 방향성 히스토그램을 구현한 프로그램 소스와 [그림 4-10]은 에지 방향성 히스토그램 생성 입니다.

```
void CPCAView::OnOHisto()
{
                CPCADoc* pDoc = GetDocument();
                ASSERT_VALID(pDoc);

                for(int i=0;i<36;i++) {pDoc->bin[i]=0;}
                for(int x=0; x<256; x++)
                {
                        for(int y=0; y<256; y++)
                        {
                if(pDoc->m_ResultImg[x+1][y] != pDoc->m_ResultImg[x][y])
                                {
                                int binnumber = (unsigned int)(

                (double)atan((double)(pDoc->m_ResultImg[x][y+1]-
                                                pDoc->m_ResultImg[x][y])
                /(double)(pDoc->m_ResultImg[x+1][y] -
                                pDoc->m_ResultImg[x][y]))*36/PI + 18);

                                        pDoc->bin[binnumber]++;
                                }
                                else pDoc->bin[35]++;
                        }

                }
                Invalidate();
}
```

[표 4-2] 에지 방향성 히스토그램 생성 구현



[그림 4-10] 에지 방향성 히스토그램 생성



178;219;290;247;241;250;219;211;351;128;255;268;350;348;489;637;865;900;4099;929;684;628;464;525;428;367;238;717;303;337;339;262;311;308;286;47865;

442;546;481;434;253;488;303;259;775;115;256;335;545;307;558;775;1023;1871;5210;800;669;682;415;699;533;481;246;1553;563;596;766;467;706;754;576;40054;

[그림 4-11] 에지 방향성 히스토그램 생성 실험 결과

V 실험 결과 및 결론

실험은 펜티엄 듀얼 코어 1.8GHz에 메모리 1G를 가진 PC에서 수행하였으며, 구현한 프로그램은 Visual C++를 사용하여 구현하였다.

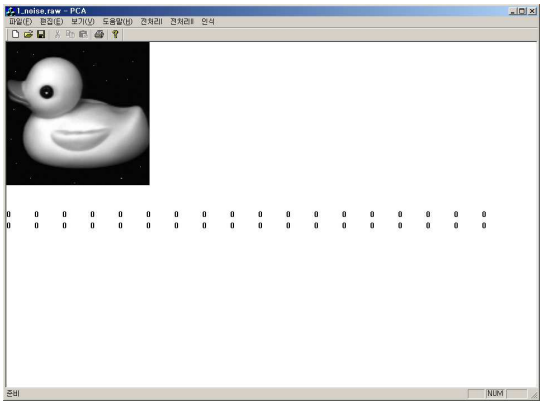
본 논문에서는 에지 방향성 히스토그램을 사용하여 물체를 인식 시스템을 구현하였다.

5.1 실험 결과

실험에 사용된 실험 영상은 [그림 5-1]에서와 같다.

[그림 5-1] 실험 영상

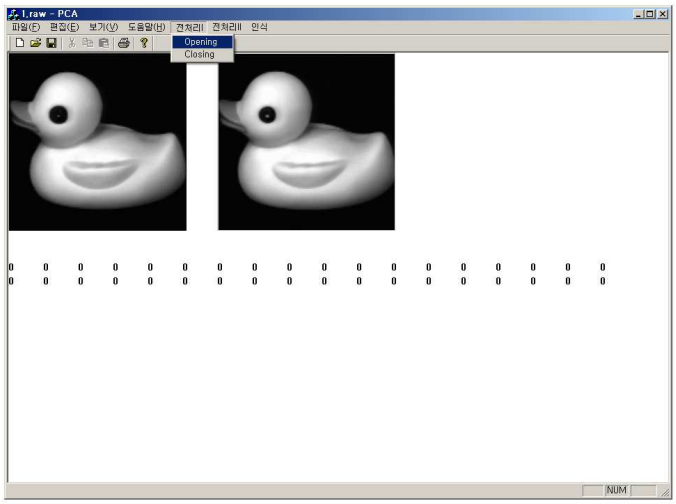
실험 영상의 크기는 256×256으로 정규화 하여 사용하였다. [그림 5-2]은 본 논문의 구현 시스템에서 영상을 불러온 장면이다.



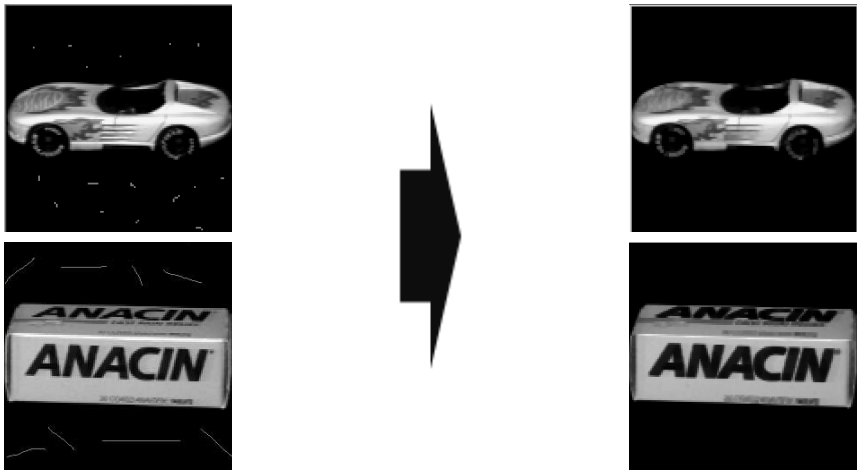
[그림 5-2] 물체 인식 시스템에서 모델 영상 Load

5.1.1 노이즈 제거

노이즈를 제거하기 위해서는 [전처리I]-[Opening]메뉴를 클릭하여 입력한 영상의 노이즈 제거한다. [그림 5-3]은 영상의 노이즈를 제거한 장면이다.



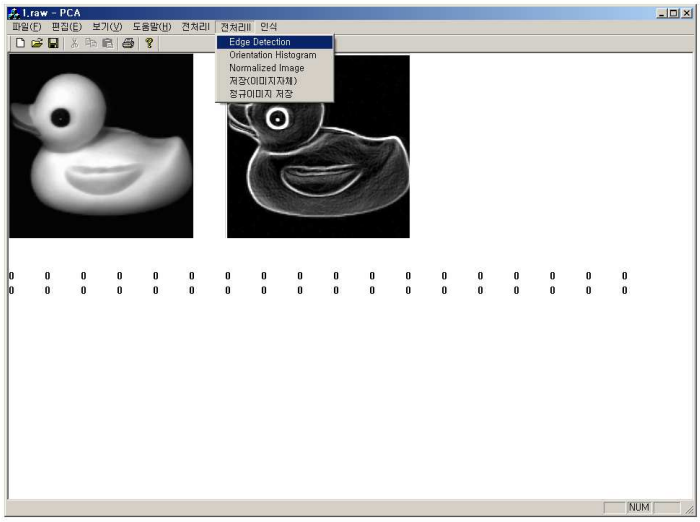
[그림 5-3] 노이즈 제거



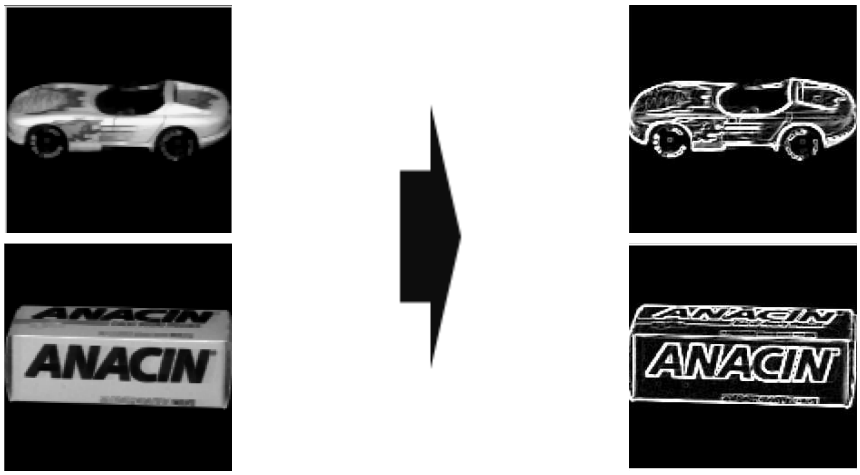
[그림 5-4] 노이즈 제거 실험 결과

5.1.2 에지 추출

에지를 추출하기 위해서는 [전처리II]-[Edge Detection]메뉴를 클릭하여 노이지를 제거한 영상의 에지를 추출한다. [그림 5-5]은 영상의 에지를 장면이다.



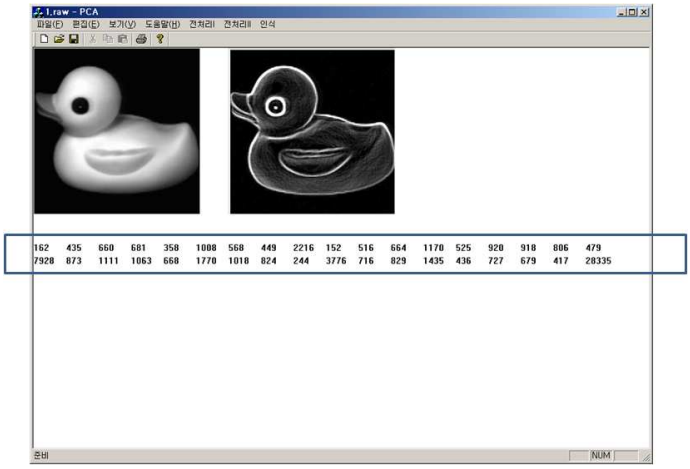
[그림 5-5] 에지 추출



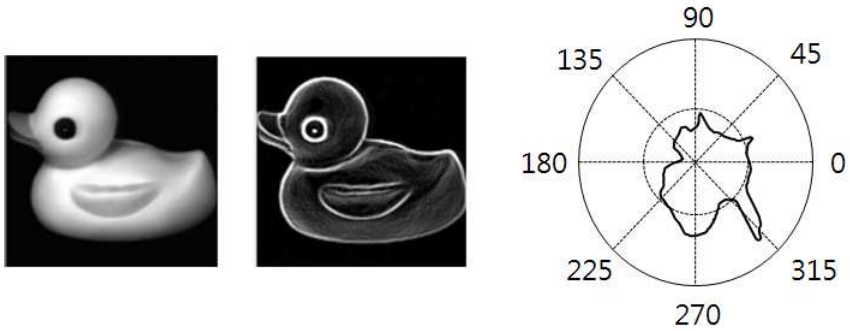
[그림 5-6] 에지 추출 실험 결과

5.1.3 에지 방향성 히스토그램 생성

에지를 추출하기 위해서는 [전처리II]-[Orientation Histogram]메뉴를 클릭하여 에지를 추출한 영상의 에지 방향성 히스토그램을 생성한다. [그림 5-7]은 에지 방향성 히스토그램을 생성한 장면이다.



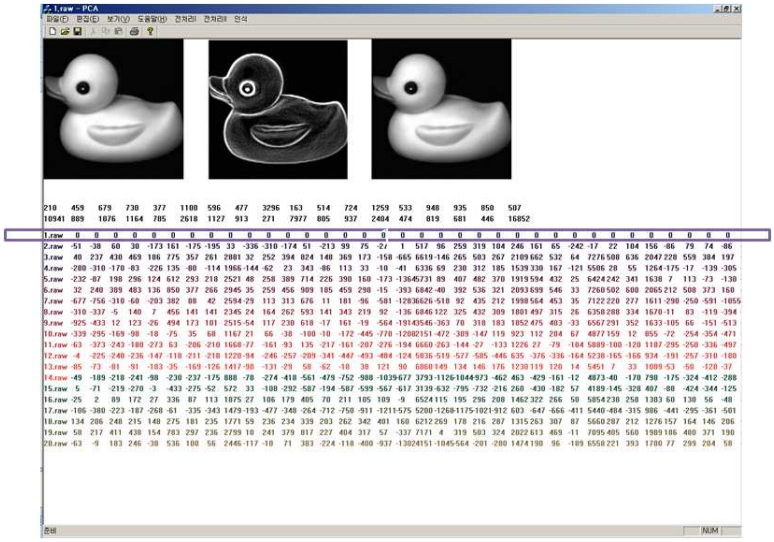
[그림 5-7] 에지 방향성 히스토그램 생성



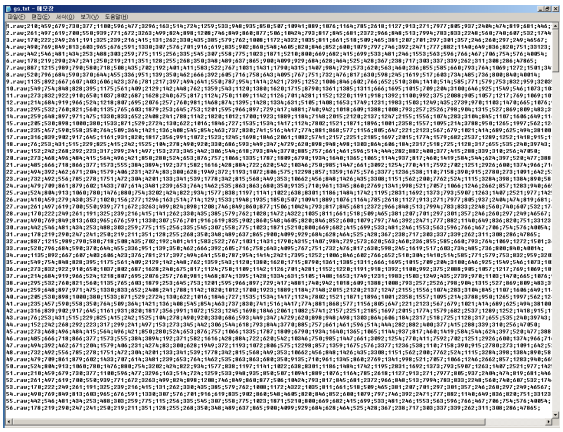
[그림 5-8] 방향성 히스토그램

5.1.4 물체 인식

에지를 추출하기 위해서는 [인식]-[물체인식]메뉴를 클릭하여 입력받은 에지 방향성 히스토그램에서 얻어진 픽셀을 모델물체와 비교하여 인식한다. [그림 5-9]은 물체 인식 실험 결과를 생성한 장면이다.

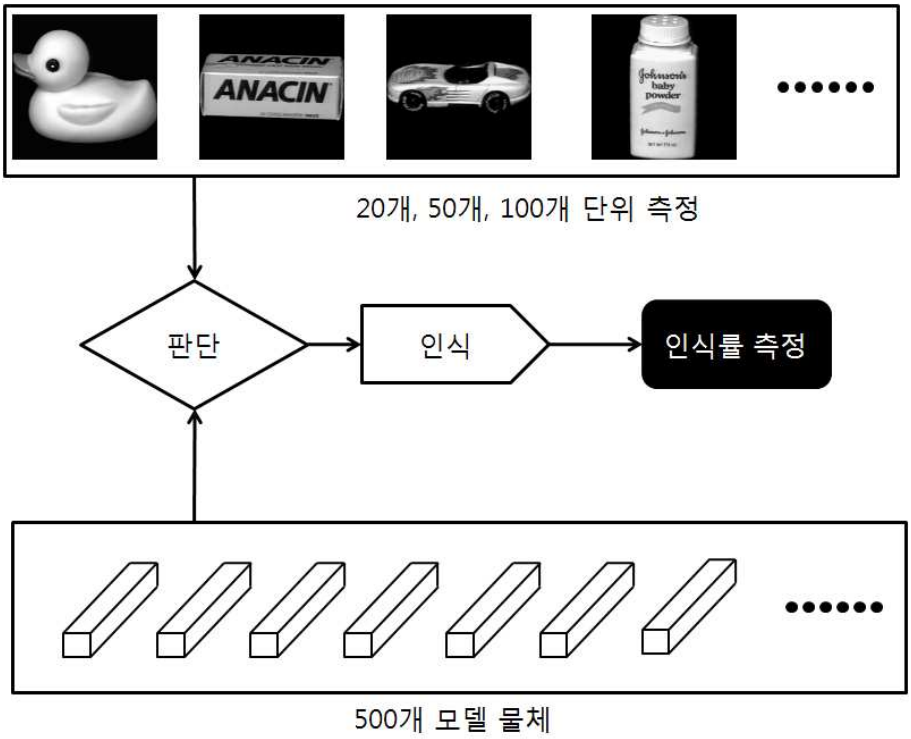


[그림 5-9] 물체 인식 실험 결과



[그림 5-10] 모델 물체의 데이터베이스

인식률을 측정하기 위해 실험 영상을 각각 20개, 50개, 100개의 단위로 모델 물체와 비교하여 [그림 5-12]와 같이 실험 하였다.



[그림 5-12] 인식률 측정

인식률을 측정하여 [표 5-1]과 같은 인식률을 보였다.

알고리즘 / 영상 개수	인식 개수	성공률	실패률	비고
20개	19개	95%	5%	
50개	47개	94%	6%	
100개	91개	91%	9%	

[표 5-1] 물체 인식 실험 결과

Ⅵ 결론

물체 인식을 하기위해 본 논문에서는 에지 방향성 히스토그램이라는 방법을 이용하여 입력 받은 영상에서 노이즈를 제거한 후 에지를 구한다. 다음으로 특정 픽셀이 에지인지 아닌지를 판별하기 위해 임계값에 따라 에지로 판정한다. 판정된 픽셀을 대상으로 에지 방향성 히스토그램을 생성하여 얻어진 픽셀들을 36저장소에 매핑하고 픽셀들의 수로부터 얻어진 방향성 히스토그램을 모델영상과 비교하여 물체를 인식하게 된다.

에지 방향성 히스토그램을 이용한 물체 인식은 픽셀 값의 분포를 이용하기 때문에 복잡한 배경, 노이즈, 크기, 조명 변화에 대해서 매우 민감하다. 그러므로 많은 학습 이미지가 필요하게 되고 영상의 크기를 모두 일정하게 맞추는 정규화 과정이 필요하게 된다.

물체 인식을 하는데 있어 본 논문에서 제안한 에지 방향성 히스토그램은 수행속도가 빠르며 매우 안정적인 인식률을 보였다. 그러나 복잡한 배경, 노이즈, 크기, 조명 변화에서 인식은 매우 어려움이 있다. 향후에는 이런 문제를 해결하여 보다 안정적인 물체 인식 알고리즘을 연구 및 개발할 계획이다.

참고 문헌

[1] J. Ohya and Y.. Kitamura, etc, "Real-Time Reproduction of 3D Human Images in Virtual Space Teleconferencing" in Proc. of '93 IEEE Virtual Reality Annual Int. Symp. pp.408-414, 1993.

[2] J. Segen and S. Kumar, "Shadow Gestures: 3D Hand Pose Estimation Using a Single Camera," CVPR99, vol. 1, pp. 479-485, Fort Collins, Colorado, June, 23-25, 1999.

[3] K. Imagawa, S. Lu and S. Igi, "Color-Based Hand Tracking System for Sign Language Recognition," Proceedings of International Conference on Automatic Face and Gesture
Recognition, Killington, 1996.

[4] Hiroshi Murase and Shree K, Nayar, "Visual Learning and Recogntion 3-D object from appearance", international journal of Computer Vision, Vol,14,1995.

[5] Dataglove model 2 operating manual. VPL Re-search Inc., 1989.

[6] J. R. Bergen and M. S. Landy. Computational modeling of visual texture segregation. In M. S.Landy and J. A. Movshon, editors, Computa-tional Models of Visual Processing, chapter 17.MIT Press, Cambridge, MA, 1991.

[7] M. Bichsel. Strategies of robust object recogni-tion for the automatic identication of humanfaces. PhD thesis, ETH Zurich, 1991. #9467.

[8] A. Blake and M. Isard. 3D position, attitude and shape input using video tracking of hands and lips. In Proceedings of SIGGRAPH 94, pages
185{192, 1994. In Computer Graphics, Annual Conference Series.

[9] R. Cipolla, Y. Okamoto, and Y. Kuno. Robust structure from motion using motion parallax. In Proc. 4th Intl. Conf. Computer Vision, pages 374 { 382, Berlin, Germany, 1993. IEEE.

[10] T. J. Darrell and A. P. Pentland. Space-time gestures. In Proc. IEEE CVPR, pages 335{340, 1993.

[11] J. Davis and M. Shah. Gesture recognition. Technical Report CS-TR-93-11, University of Central Florida, Orlando, FL 32816, 1993.
[12] B. Dorner. Hand shape identication and track-ing for sign language interpretation. In Looking at people workshop, Chambery, France, 1993. IJ-CAI.
[13] S. S. Fels and G. E. Hinton. Glove-talk: a neu-ral network interface between a data-glove and a speech synthesizer. IEEE Trans. Neural Net-works, 4(1):2{7, 1992.
[14] W. T. Freeman and E. H. Adelson. The design and use of steerable lters. IEEE Pat. Anal. Mach. Intell., 13(9):891{906, September 1991.
[15] W. T. Freeman and M. Roth. Orientation his-tograms for hand gesture recognition. Techni-cal Report 94-03, Mitsubishi Electric Research Labs., 201 Broadway, Cambridge, MA 02139,1994.
[16] M. Fukumoto, K. Mase, and Y. Suenaga. Real-time detection of pointing actions for a glove-free interface. In Workshop on Machine Vision Applications, Tokyo, 1992. IAPR.
[17] G. H. Granlund. In search of a general picture processing operator. Comp. Graphics, Image Proc., 8:155{173, 1978.
[18] K. Ishibuchi, H. Takemura, and F. Kishino. Real time hand shape recognition using pipe- line image processor. In IEEE Intl. Workshop on robot and human communication, pages 111{116. IEEE, 1992.
[19] J. Malik and P. Perona. Preattentive texture discrimination with early vision mechanisms. J.Opt. Soc. Am. A, 7:923{931, 1990.
[20] R. K. McConnell. Method of and apparatus for pattern recognition. U. S. Patent No. 4,567,610, Jan. 1986.
[21] R. C. Nelson. Visual homing using an asso-ciative memory. In Proceedings of the DARPA Image Understanding Workshop, pages 245{262,1989.
[22] R. W. Picard and M. Gorkani. Finding percep-tually dominant orientations in natural textures. To appear in Spatial Vision, special Julesz birth-day issue. also available as Percep. Comp. TR #229, M.I.T. Media Laboratory, 1993.

[23] T. Poggio and F. Girosi. Networks for approx-imation and learning. Proc. IEEE, 78(9):1481{1497, 1990.
[24] J. M. Rehg and T. Kanade. Digiteyes: vision-based human hand tracking. Technical Re-port CMU-CS-93-220, Carnegie Mellon School of Computer Science, Pittsburgh, PA 15213,1993.
[25] D. Rubine and P. McAvinney. Programmablenger-tracking instrument controllers. Com-puter Music Journal, 14(1):26{41, 1990.
[26] J. Segen. Gest: a learning computer vision system that recognizes gestures. In Machine Learning IV. Morgan Kauman, 1992. edited by Michalski et. al.
[27] D. J. Sturman. Whole hand input. PhD the-sis, Massachusetts Institute of Technology, 1992. MIT Media Lab.
[28] V. C. Tartter and K. C. Knowlton. Perception of sign language from an array of 27 moving spots. Nature, (239):676{678, Feb. 19, 1981.
[29] Jong-Min Kim, and Hwan-Seok Yang, "A Study on Object Recognition Technology using PCA in Variable Illumination", Lecture Notes in Computer Science LNAI 4093 pp911-918, 2006
[30] Jong-Min Kim, Hwan-Seok Yang and Woong-Ki Lee "Network-Based Face Recognition System Using Multiple Images", Lecture Notes in Artificial Intelligence LNAI 4088 pp626-631, 2006

저작물 이용 허락서

학 과	전산 통계	학 번	20067043	과 정	석 사
성 명	한글: 송 맹 규 한문 : 宋 孟 奎 영문 : Song Maeng-kyu				
주 소	전북 군산시 월명동 다원파크빌 703호				
연락처	HP: 011-9642-0536 / E-MAIL : songmkyu@nate.com				
논문제목	한글 : 에지 방향성 히스토그램을 이용한 실시간 물체 인식 시스템 영어 : The Object Recognition System uing RealTime Orientation Histogram				

본인이 저작한 위의 저작물에 대하여 다음과 같은 조건아래 조선대학교가 저작물을 이용할 수 있도록 허락하고 동의합니다.

- 다 음 -

1. 저작물의 DB구축 및 인터넷을 포함한 정보통신망에의 공개를 위한 저작물의 복제, 기억장치에의 저장, 전송 등을 허락함
2. 위의 목적을 위하여 필요한 범위 내에서의 편집·형식상의 변경을 허락함. 다만, 저작물의 내용변경은 금지함.
3. 배포·전송된 저작물의 영리적 목적을 위한 복제, 저장, 전송 등은 금지함.
4. 저작물에 대한 이용기간은 5년으로 하고, 기간종료 3개월 이내에 별도의 의사 표시가 없을 경우에는 저작물의 이용기간을 계속 연장함.
5. 해당 저작물의 저작권을 타인에게 양도하거나 또는 출판을 허락을 하였을 경우에는 1개월 이내에 대학에 이를 통보함.
6. 조선대학교는 저작물의 이용허락 이후 해당 저작물로 인하여 발생하는 타인에 의한 권리 침해에 대하여 일체의 법적 책임을 지지 않음
7. 소속대학의 협정기관에 저작물의 제공 및 인터넷 등 정보통신망을 이용한 저작물의 전송·출력을 허락함.

동의여부 : 동의(○) 반대( )

2007 년 11 월 28 일

저작자: 송 맹 규 (서명 또는 인)

조선대학교 총장 귀하