



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](#)

工學碩士 學位論文

유전자 알고리즘과 다해상도 기법을 이용한 활성 윤곽선 모델 기반의 물체 추출

Object Extraction Based-on Active Contour Model Using
a Genetic Algorithm and Multiresolution Method



指導教授 趙 爽 濟

2010年 2月

韓國海洋大學校 大學院

制 御 計 測 工 學 科

李 基 煥

목 차

1. 서 론	1
2. 에너지 함수에 의한 율곽선 검출	3
2.1 활성 율곽선 모델	3
2.2 유전자 알고리즘을 이용한 에너지 최소화	9
2.2.1 유전자 알고리즘	9
2.2.2 유전자 알고리즘을 이용한 율곽 검출	15
2.3 피라미드 기법	16
3. 유전자 알고리즘과 피라미드 기법을 이용한 활성 율곽선 모델 ..	19
3.1 제안한 피라미드 기법	19
3.2 제안한 유전자 알고리즘	22
3.2.1 염색체 구조	22
3.2.2 유전연산자	24
4. 실험 및 고찰.....	29
6. 결 론	39
참고 문헌	41

Object Extraction Based-on Active Contour Model Using a Genetic Algorithm and Multiresolution Method

Ki-Hwan, Lee

*Dept. of Control & Instrumentation Engineering,
Graduate School, Korea Maritime University*



Active contour model is so-called snake model. In this paper, we propose a object extraction based-on active contour model using a genetic algorithm for initial contour problem and effective boundary extraction. Image pyramid method used to acquire low resolution from original input image and set a initial contour. Two neighborhood parameters above set contour is selected genes and than genetic operator is applied to find object boundary. Low resolution image that is used to extract a contour enlarged step by step, compute localized active contour energy in interpolation area. Insert new contour parameter location which is minimum energy. The proposed method extracted efficiently boundary of object.

제 1 장 서 론

컴퓨터 시각에서 영상 분할은 목적에 따라 영상의 표현을 좀 더 의미 있고 해석하기 쉽게 변환하는 것이다[1]. 영상의 배경과 물체를 분리하여 영상에서 원하는 정보 추출하는 것으로 영상검색, 물체인식, 컴퓨터 비전 등 많은 분야에서 활용되고 있다. 영상 분할의 결과는 전체 영상을 부분적으로 포함하는 지역의 집합이거나, 영상으로부터 추출된 경계의 집합이다. 추출된 경계의 집합을 기준으로 영상의 관심 있는 영역과 그 밖의 영역으로 구분하여 효과적으로 해석할 수 있다. 그러므로 영상에서 물체의 경계를 추출하는 것은 영상처리분야에서 매우 중요한 부분을 차지한다[2-25].

물체의 경계 추출 방법 중 윤곽선을 추출하는 방법이 있다. 대표적인 윤곽선 추출 방법으로 레벨 셋 방법(level set method)[2, 3]과 활성 윤곽선 모델(active contour model)[4-25]가 있다. Kass[4] 등이 제안한 활성 윤곽선 모델은 물체의 경계부분에서 최소값을 가지는 에너지 함수를 정의하고 에너지 최소화 과정을 거쳐 물체의 경계를 추출하는 방식이다. 활성 윤곽선 모델은 저대비 영상(low contrast image)이나 잡음이 있는 영상에 강인한 특징이 있다. 그러나 일반적인 활성 윤곽선 모델은 초기 윤곽선을 올바르게 설정해야 하고 오목한 경계(boundary concavity)부분에서 잘못된 경계를 찾는 문제점을 가지고 있다. 그리고 계산 과정에서 변분법을 이용하여 에너지를 최소화하므로 고차 미분의 영향으로 불안정하고 계산복잡도가 높다. 이러한 초기 활성 윤곽선 모델의 문제점들을 해결하기위해 다양한 방법들이 제안되었다. 그중 동적 계획법(dynamic programming)[5]는 안정적이고 잡음에 강인하지만 계산복잡도가 높아지는 단점이 있다. 탐욕 알고리즘(greedy algorithm)[6]을 적용할 경우 추출 방법이 단순하여 유연하게 경계를 빠르게 찾지만 오목한 경계부분에서 정확도가 떨어진다. 그 밖에 초기 윤곽선 문제를 해결하는 방법으로 기울기 벡터 흐름-gradient vector flow)[7]이 제안되었으나 윤곽선이 이동해야할 방향을 자

동으로 탐색하지 못하는 문제점이 있다.

최근 활성 윤곽선 모델의 에너지 최소화 과정에 최적화 이론을 적용하여 윤곽선을 추출하는 방법들이 제안되고 있다[8-15]. 그중 대표적인 방법이 L. A. MacEachern[8] 등과 K. J. Mun[9] 등이 제안한 유전자 알고리즘을 적용하여 물체를 추출하는 방법이다. 유전자 알고리즘은 큰 탐색영역을 다룰 수 있고 최적해의 탐색방향을 찾아갈 수 있다는 장점 때문에 활성 윤곽선 모델에 적용하는 많은 연구가 이루어졌다. 하지만 염색체구조가 복잡하고 계산복잡도가 증가하는 문제점을 보였다.

본 논문에서는 유전자 알고리즘과 피라미드 기법을 이용한 활성 윤곽선 모델 기반의 물체 추출기법을 제안한다. 유전자 알고리즘을 활성 윤곽선 모델에 적용하여 기존의 활성 윤곽선 모델의 문제점인 초기 윤곽선문제와 오목한 경계 등 특정 물체의 윤곽을 효과적으로 추출한다. 그리고 노드의 좌표를 이진코드화 하여 염색체구조를 간소화 시킨다. 정확도를 높이기 위해 많은 세대를 반복하게 되어 계산복잡도가 높아지는 문제는 피라미드 기법을 적용하여 개선한다.

제안한 방법은 가우시안 피라미드 기법(gaussian pyramid method)으로 입력 영상의 근사 영상을 얻는다. 근사 영상에 초기 윤곽선을 설정하고 활성 윤곽선 모델과 유전자 알고리즘을 이용하여 물체의 경계를 추출한다. 경계가 추출된 근사 영상을 원 영상의 크기로 확대하고 노드간의 벌어진 공간에 국부적으로 활성 윤곽선 에너지를 계산하여 근사 영상에서 추출된 윤곽선 사이에 새로운 윤곽선 노드를 삽입한다. 이러한 과정을 거쳐 정확한 경계를 효율적으로 추출한다. 이로 인해 기존의 문제점인 초기 윤곽선 문제에 강인하며 오목한 경계 등 특정한 물체의 형태에 상관없이 경계를 효율적으로 정확히 추출하였다.

본 논문의 구성은 제 2장에서 기본적인 활성 윤곽선 모델, 유전자 알고리즘 그리고 피라미드 기법에 대해 기술한다. 제 3장은 본 논문에서 제안하는 방법에 관해 설명하고, 제 4장에서는 실험을 통하여 제안한 방법의 성능을 분석하고 기존의 방법들과 비교 고찰한다. 마지막으로 제 5장에서는 전체 내용에 대해 결론을 내린다.

제 2 장 에너지 함수에 의한 윤곽선 추출

2.1 활성 윤곽선 모델

활성 윤곽선 모델은 Kass[4] 등이 제안한 알고리즘으로 스네이크 모델(snake model)이라고도 하며 윤곽선 분할(contour segmentation), 모델 적합(model fitting) 및 윤곽선 추적(contour tracking)등의 방법으로 널리 사용되고 있다. 활성 윤곽선 모델은 물체의 윤곽부분의 특징을 에너지 함수로 정의하고, 에너지가 최소가 될 때까지 반복하여 물체의 윤곽선을 추출하는 방법이다. 활성 윤곽선 모델을 이용하여 윤곽선을 추출하는 예를 그림 2.1과 같다. 그림 2.1 (a)는 추출하고자 하는 물체근처에 초기 윤곽선을 설정하고 에너지 함수를 계산하여 에너지가 최소가 되는 부분인 물체의 윤곽을 찾으며 그림 2.1 (b)와 같은 윤곽선 결과를 구한다.

활성 윤곽선 모델의 에너지 함수는 윤곽선을 구성하는 노드들 간의 관계로 나타내며 그림 2.2와 같이 표현된다. 윤곽선을 구성하는 각각의 노드에서 에너지를 계산하고 노드 v_i 를 에너지가 가장 낮은 물체의 윤곽부분인 v'_i 의 위치로 이동시켜 물체의 윤곽을 추출한다.

전체 활성 윤곽선 모델의 에너지 함수 E 는 식 (2.1)과 같이 정의할 수 있다 [16].

$$E = \int [\alpha E_{cont} + \beta E_{curv} + \gamma E_{img}] ds \quad (2.1)$$

여기서 E_{cont} 는 연속성 에너지(continuity energy), E_{curv} 는 곡률에너지(curvature energy) 그리고 E_{img} 는 영상에너지(image energy)를 나타낸다. α, β, γ 는 각 에너지 항들의 가중치(weight)를 나타내고 s 는 활성 윤곽선 모델

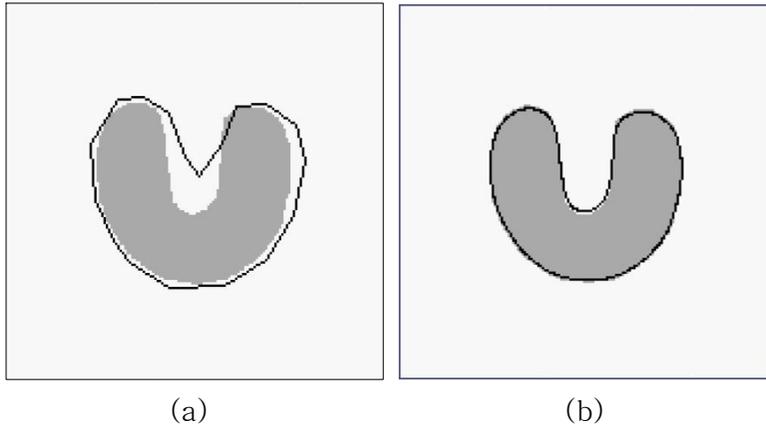


그림 2.1 활성 윤곽선 모델의 예
(a) 초기 윤곽선 (b) 결과 영상

Fig. 2.1. Example of active contour models
(a) Initial contour (b) Result image

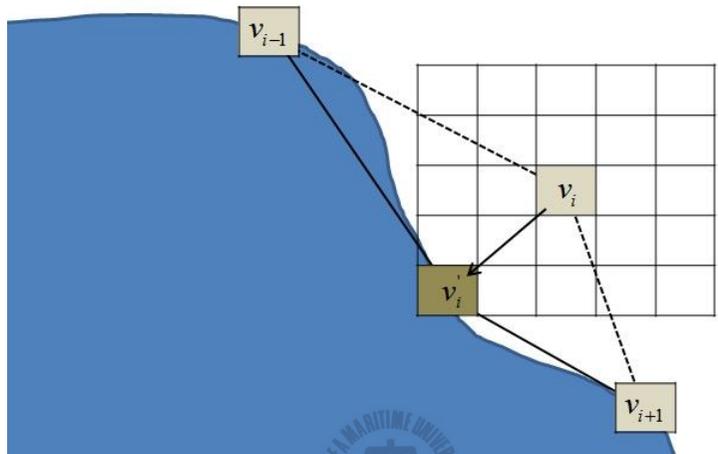


그림 2.2 활성 윤곽선 모델의 구조

Fig. 2.2. The structure of active contour models

을 구성하는 노드들이다. E_{cont} 와 E_{curv} 는 물체의 윤곽선의 형태에 영향을 미치는 에너지로 내부 에너지(internal force)라고 하고, E_{img} 는 물체의 에지나 밝기와 같은 특징으로 윤곽선을 끌어당겨 영상에 포함된 물체의 형태로 표현하게 하며 외부 에너지(external force)라고 한다[4, 7, 9, 17].

윤곽선의 연속성을 나타내는 에너지 함수 E_{cont} 는 윤곽을 구성하는 각 좌표의 간격이 일정할수록 최소값을 가지며 식 (2.2)와 같이 나타낸다.

$$\begin{aligned} E_{cont} &= E_{cont}(v_1, v_2, \dots, v_{n-1}, v_n) \\ &= E_{cont_1}(v_n, v_1) + \sum_{i=1}^{n-1} E_{cont_{i+1}}(v_i, v_{i+1}) \end{aligned} \quad (2.2)$$

여기서 v_i 는 활성 윤곽선 모델의 i 번째 노드의 좌표이고 i 는 1부터 n 까지 노드의 색인(index)이다. 각 노드에서의 연속성 에너지는 식 (2.3)과 같이 전체 노드간의 평균거리 \bar{d} 와 연속된 두 노드간의 거리차로 구할 수 있다.

$$E_{cont_i}(v_{i-1}, v_i) = |\bar{d} - |v_{i-1} - v_i|| \quad (2.3)$$

여기서, 노드간의 평균 거리 \bar{d} 는 식 (2.4)와 같이 나타낸다.

$$\bar{d} = \frac{1}{n} (\sum |v_{k-1}, v_k|) \quad (2.4)$$

즉, 평균 거리와 노드간의 거리차가 적을수록 연속성 에너지의 합은 최소가 된다.

윤곽선의 곡률을 나타내는 에너지 함수 E_{curv} 는 윤곽의 곡률이 작을수록 최소값을 가지며 식 (2.5)와 같이 나타낸다.

$$\begin{aligned}
E_{curv} &= E_{curv}(v_1, v_2, \dots, v_{n-1}, v_n) \\
&= E_{curv_1}(v_n, v_1, v_2) + \sum_{i=1}^{n-2} E_{curv_{i+1}}(v_i, v_{i+1}, v_{i+2}) \\
&\quad + E_{curv_2}(v_{n-1}, v_n, v_1)
\end{aligned} \tag{2.5}$$

여기서 v_i 는 활성 윤곽선 모델의 i 번째 노드의 좌표이고 i 는 1부터 n 까지 노드의 색인이다. 각 노드에서의 곡률 에너지는 기준노드인 v_i 와 주변 노드인 v_{i-1} 과 v_{i+1} 의 관계로 나타내며 식 (2.6)과 같다.

$$\begin{aligned}
&E_{curv_i}(v_{i-1}, v_i, v_{i+1}) \\
&= \frac{1}{\Delta s} \sqrt{\left(\frac{\Delta x_i}{\Delta s_i} - \frac{\Delta x_{i+1}}{\Delta s_{i+1}}\right)^2 + \left(\frac{\Delta y_i}{\Delta s_i} - \frac{\Delta y_{i+1}}{\Delta s_{i+1}}\right)^2}
\end{aligned} \tag{2.6}$$

여기서 Δx_i , Δy_i , Δs_i 및 Δs 는 식 (2.7)과 같다.

$$\begin{aligned}
\Delta x_i &= |x_i - x_{i-1}| \\
\Delta y_i &= |y_i - y_{i-1}| \\
\Delta s_i &= \sqrt{\Delta x_i^2 + \Delta y_i^2} \\
\Delta s &= \frac{(\Delta s_i + \Delta s_{i+1})}{2}
\end{aligned} \tag{2.7}$$

x_i , y_i 는 각각 영상의 좌표를 나타낸다. 단, $E_{curv_1}(v_n, v_1, v_2)$ 와 $E_{curv_n}(v_{n-1}, v_n, v_1)$ 의 경우 처음과 마지막 노드의 좌표를 이용한다.

영상의 에너지는 영상의 에지를 계산하여 정의할 수 있다. n 개의 노드로 구성된 윤곽선의 영상 에너지 E_{img} 는 식 (2.8)과 같이 표현된다.

$$\begin{aligned}
E_{img} &= E_{img}(v_1, v_2, \dots, v_{n-1}, v_n) \\
&= \sum_{i=1}^n E_{img_i}(v_i)
\end{aligned} \tag{2.8}$$

노드 v_i 에서의 영상 에너지 $E_{img_i}(v_i)$ 는 식 (2.9)과 같이 해당 노드의 영상 기울기 값의 역수이다.

$$E_{img_i}(v_i) = \frac{1}{|\nabla I(v_i)|^2} \tag{2.9}$$

여기서 $I(v_i)$ 는 노드 v_i 에서의 영상의 명암도를 나타낸다. 만약 $|\nabla I(v_i)|^2$ 의 값이 0이 될 경우 식 (2.9)가 정의되지 않기 때문에 이와 같은 경우 영상에너지는 명암도의 최대값으로 표현한다.

기존의 활성 윤곽선 모델은 각 노드의 에너지 함수를 계산하여 에너지가 최소로 되는 곳으로 노드를 이동시켜 물체를 추출한다. 그러나 기존의 Kass 등이 제안한 활성 윤곽선 모델은 초기 윤곽선 설정문제, 오목한 경계에서의 잘못된 경계추출, 그리고 계산복잡도와 같은 문제점이 있다. 추출하고자 하는 물체에 대한 지식을 기반으로 그 물체 주위에 윤곽선 획득범위를 정의하고 초기 윤곽선을 설정해야하며, 초기 윤곽선을 잘못 설정할 경우 윤곽선이 추출되지 않거나 잘못된 결과로 수렴하게 된다. 잘못된 초기 윤곽선 설정에 따른 윤곽선 추출에 실패한 경우를 그림 2.3과 같이 나타내었다. 물체 주위에 초기윤곽선을 설정하지 않은 경우를 그림 2.3 (a)와 같이 나타내었다. 그에 따른 윤곽이 추출된 결과가 그림 2.3 (b)와 같다. 물체 근처에 초기 윤곽선을 설정한 경우 물체의 윤곽을 찾아가지만 그렇지 않은 윤곽선은 찾아가지 못하는 것을 볼 수 있다. 이를 해결하기 위해 다해상도 기법(multiresolution methods)[18], 압력 힘(pressure forces)[19], 거리 위치(distance potentials)[20] 등의 방법이 제

안되었다. 이 방법들은 외부에너지의 윤곽선 탐색 범위를 확장하거나, 윤곽선의 탐색방향을 제시하여 원하는 경계를 찾는 문제를 해결하였다. 오목한 경계 문제는 특정 부분에서 윤곽선이 수렴하지 않아 잘못된 경계를 추출하는 문제로 그림 2.4와 같이 나타난다. 초기 윤곽선을 그림 2.4의 (a)와 같이 물체 근처에 설정했지만 에너지 함수의 관계에 의해 그림 2.4의 (b)처럼 오목한 경계로 윤곽선이 진행되지 않고 멈추는 현상이 나타난다. 이러한 문제는 압력 힘 (pressure forces)[19], 조절 점(control points)[20], 기울기 벡터 흐름 (gradient vector flow)[21] 등의 방식이 제안되어 문제를 해결하였다. 이처럼 기존의 문제점을 해결하기위해 많은 방법이 제안되었지만 초기 윤곽선과 오목한 경계에서의 수렴문제 모두를 해결할 수 없었다.

2.2 유전자 알고리즘을 이용한 에너지 최소화

2.2.1 유전자 알고리즘



유전자 알고리즘(Genetic Algorithm)은 자연계의 진화 원리를 모방하여 컴퓨터 알고리즘과 결합시킨 방법으로 Holland[26]에 의해 처음 제안되었다. 최적의 해를 찾는 도구로써 널리 사용되고, 확률에 기반을 두고 자연 진화 과정을 모방해 문제 풀이 또는 모의실험에 이용하는 최적화 방법이다. 문제해결을 위해 하나의 해를 다루기보다는 집단을 취급하며, 기존의 순수 임의 탐색 알고리즘에서 얻을 수 없는 탐색과정의 정보를 보관할 뿐만 아니라 이 정보를 이용하여 새로운 탐색에 참고하는 기능과 주어진 판단기준에 따라 해를 탐색하는 적응기능을 가지고 있다.

유전자 알고리즘의 순서도는 그림 2.5와 같다. 초기집단 생성 단계에서는 주어진 문제의 해가 될 가능성이 있는 개체들의 집합인 집단이 형성된다. 초기점들은 해공간 내에 무작위로 분포되도록 선택되거나 아니면 경험적인 방법으

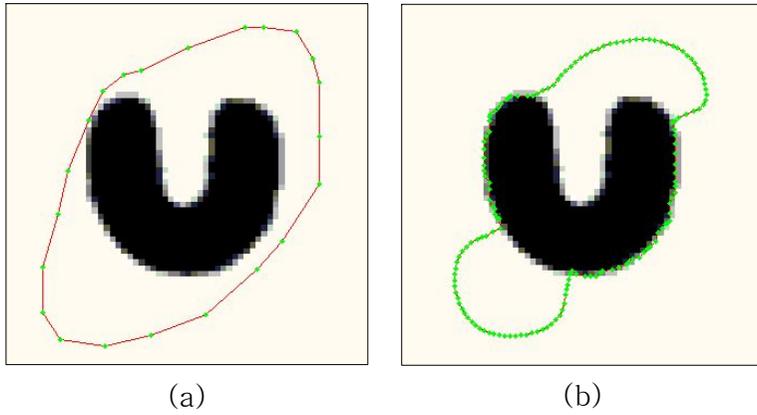


그림 2.3 잘못된 윤곽선 추출
 (a) 초기 윤곽선 (b) 윤곽선 추출 실패

Fig. 2.3. False contour detection
 (a) Initial curve (b) Failed to detect the curve

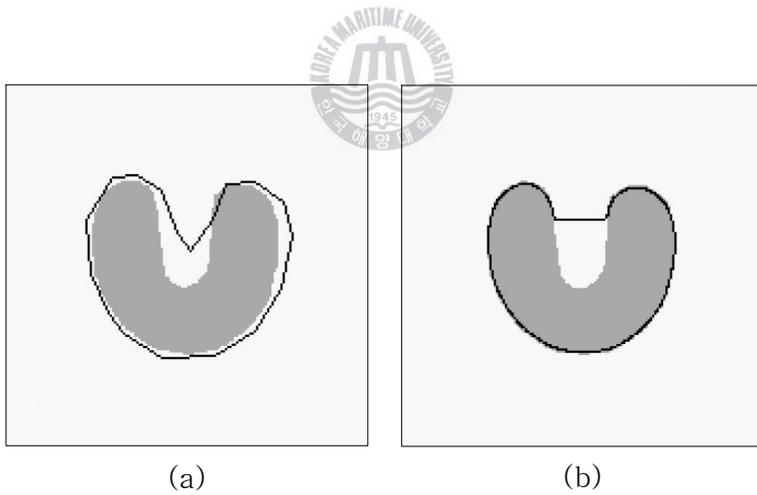


그림 2.4 잘못된 윤곽선 추출
 (a) 초기 윤곽선 (b) 윤곽선 추출 실패

Fig 2.4. False contour detection
 (a) Initial curve (b) Failed to detect the curve

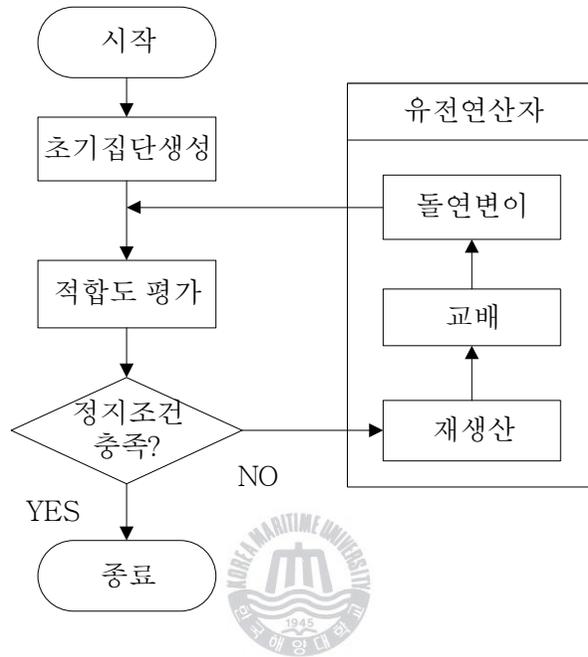


그림 2.5 유전자 알고리즘의 순서도

Fig. 2.5. Flow graph of a genetic algorithm

로 선택된다. 적합도 평가 단계에서는 개체들의 강점과 약점이 적합도 항목으로 평가된다.

여기서 각 개체는 복호화 되어 목적함수를 제공하고 이로부터 적합도가 계산된다. 계산된 적합도에 따라 성능이 좋은 개체들은 더 많이 선택되어 복제되고, 성능이 나쁜 개체들은 집단으로부터 소멸된다. 그리고 유전연산자를 적용하여 집단을 변형시키고 현 세대의 개체집단에서 정지조건을 만족할 경우 유전자 알고리즘을 종료하고 그렇지 않은 경우 정지조건을 만족할 때까지 같은 과정을 반복한다.

유전연산자의 경우 재생산(reproduction), 교배(crossover), 그리고 돌연변이(mutation)가 대표적인 유전연산자이다. 재생산은 적합도를 기반으로 개체를 선택하여 새로운 집단을 형성하는 과정이다. 교배는 부모 염색체가 가지고 있는 유전자 정보를 교환하여 새로운 자식 염색체를 생성한다. 돌연변이는 선택된 개체의 하나 이상의 유전정보를 임의로 변경하여 집단에 새로운 정보를 제공한다.

초기집단의 생성 단계에서 주어진 문제의 해가 될 수 있는 염색체들의 집단을 그림 2.6과 같이 형성한다. 염색체는 한 개 이상의 유전자로 구성된다. k 세대에서의 집단 $P(k)$ 는 염색체로 구성되는 N 개의 개체들의 집합으로 정의되며 식 (2.10)과 같다.

$$P(k) = \{s_1(k), s_2(k), \dots, s_i(k), \dots, s_N(k)\} \quad (2.10)$$

여기서 $s_i(k)$ 는 i 번째의 염색체로서 탐색공간 상의 한 점을 나타내며 $N(>1)$ 는 집단크기(population size)를 나타낸다.

유전자 표현은 방식에 따라 이진 코딩, k -진수 코딩, 그레이 코딩, 실수 코딩 등으로 분류할 수 있다. 이진 코딩 방식은 이진 스트링 모양으로 염색체를 표현하는 방식으로 교배시 교배점이 발생할 수 있는 위치가 k -진수 코딩보다 많

아저 다양한 변이가 이루어지게 한다.

재생산은 자연환경의 적자생존(survival of the fittest) 또는 자연도태(natural selection)현상을 모방하려는 인위적인 방법으로 적합도 값을 기반으로 집단 $P(k)$ 내의 개체들을 선택하고 교배급원 $\bar{P}(k+1)$ 을 형성해준다. 이러한 선택은 전체 집단을 강하게 해주는 특징을 가지며, 이와 같은 과정에서 성능이 좋은 개체들은 더 많이 선택되어 복제되고 반면 성능이 나쁜 개체들은 집단으로 소멸하게 된다.

교배는 두 부모 염색체가 가지고 있는 유전자 정보의 일부를 교환하여 새로운 자식 염색체를 생성해내는 과정이다 교배 방식으로는 일점교배, 다점교배, 순환교배, 균등교배 등이 있다. 이진 코드의 일점 교배 연산은 그림 2.7과 같이 교배점이 정해지면 유전자 정보를 교환하여 새로운 자식 염색체를 만든다.

돌연변이는 모의진화가 계속되는 동안 재생산과 교배 연산자는 집단을 더욱 강하게 해주고 이로 인해 염색체들이 서로 닮아가게 된다. 그러나 이러한 현상이 세대 초기에 발생하게 되면 유전자의 다양성 결핍으로 준 최적해(suboptimal solution)나 사점(dead corner)에 빠지는 요인이 된다. 이러한 문제를 벗어나기 위한 방법으로 염색체 내의 무작위로 선택된 비트를 돌연변이 확률을 토대로 변경시킨다. 그림 2.8은 단순 돌연변이 연산으로 무작위로 선택된 비트를 임의의 값으로 바꾼다.

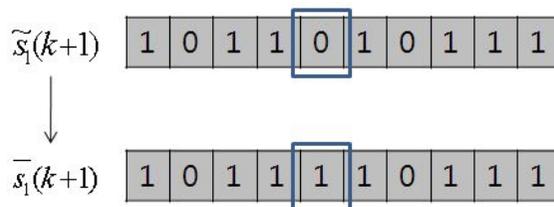


그림 2.8 이진 코드의 단순 돌연변이 연산자의 예

Fig. 2.8. Examples of simple mutation operators of binary code

2.2.2 유전자 알고리즘을 이용한 윤곽 추출

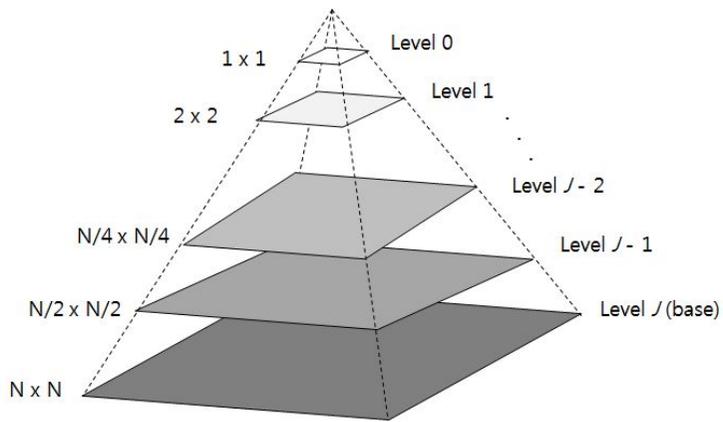
유전자 알고리즘의 에너지 최적화 과정과 활성윤곽선 모델의 에너지 최소화 과정이 비슷하다. L. A. MacEachern[8]은 위와 같은 유사성을 이용하여 활성 윤곽선 모델에 유전자 알고리즘을 적용하여 윤곽선을 추출할 수 있음을 확인하였다. 이를 기초로 활성윤곽선 모델에 유전자 알고리즘을 적용하여 윤곽을 추출하는 방법들이 제안되었다[8-14].

배대규 등[12]는 각 노드의 좌표와 노드의 형(Type) 그리고 에너지 값으로 검색체를 구성하여 유전연산자를 적용하였다. 노드의 형을 이용하여 초기 윤곽선을 구성하게 되는데 폐곡선이 아닌 개곡선으로 초기 윤곽선을 설정한다. 이때 노드의 처음과 끝 점의 위치를 유지시켜 주기 위해 처음과 끝 노드들은 생성이나 소멸하지 않은 형으로 정의하고, 다른 노드들은 자유롭게 이동하는 형으로 정의하였다. 이 방법은 개곡선으로 윤곽을 찾을 수 있지만 초기 윤곽선의 양 끝점의 위치가 고정되어 있어서 형태가 부자연스러울 수 있고, 양 끝점을 찾고자 하는 물체의 윤곽에 위치시켜야만 정확한 윤곽을 얻을 수 있는 단점이 있다. K. J. Mun[9]는 활성 윤곽선 모델의 영상 에너지를 극좌표의 개념으로 수정하여 유전자 알고리즘을 적용하였다. 그러나 계산 속도를 줄이기 위해 검색범위(searching range)를 지정해주어 검색범위 영역에서 윤곽을 찾기 때문에 물체의 윤곽이 검색범위 밖의 영역에 존재할 경우 윤곽을 찾지 못하는 문제가 발생한다. 박선영 등은 B-spline으로 활성 윤곽선 모델을 정의하고 윤곽선의 좌표를 검색체로 정의하여 윤곽을 찾았다. 하지만 움푹한(concave) 형태를 갖는 복잡한 물체의 경계를 찾고자 할 때 최적화된 노드의 개수를 설정하지 않을 경우 적절한 분할 결과를 얻지 못하고 꼬인 곡선을 만드는 단점이 있다. 그리고 초기 곡선의 위치에 따라 수렴속도와 실험결과가 달라지는 단점이 있다 [11]. 이외에도 많은 연구[13, 14]가 진행되었지만 윤곽선 추출의 효율성을 높이기 위해 복잡한 검색체 구조를 설정해야 하며, 계산복잡도 또한 높아지는 문제점이 발생하였다.

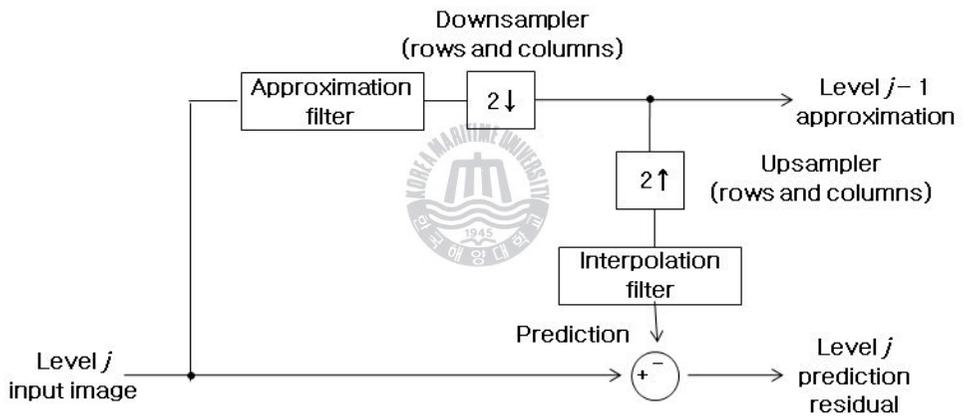
기존의 방법들은 유전자 알고리즘을 활성 윤곽선 모델에 적용할 경우 유전자 알고리즘의 장점을 제대로 살리지 못하는 문제를 확인할 수 있다.

2.3 피라미드 기법

피라미드 기법은 여러 해상도의 영상을 표현할 수 있으며 뛰어난 성능을 나타낼 뿐만 아니라 개념적으로도 간략한 구조를 지닌다. 영상 피라미드는 피라미드 형상으로 감소하는 해상도의 영상들을 정렬시켜 모아놓은 것으로 그림 2.9 (a)와 같은 구조를 지닌다. 가장 아래의 레벨 J 는 기저 영상(base image)으로 처리될 영상의 고해상도를 표현하며 피라미드 위쪽으로 올라 갈수록 크기와 해상도가 감소하는 근사 영상(approximation image)을 표현하게 된다. 레벨 J 의 영상의 크기는 $2^J \times 2^J$ 또는 $N \times N$ 이기 때문에, $J = \log_2^N$ 로 표현되며 피라미드 위쪽으로 한 단계씩 올라갈수록 영상의 크기는 1/4 만큼 줄어든다. 피라미드 기법은 레벨 $j-1$ 단계의 근사 영상과 j 단계의 예측차 영상(prediction image)을 이용하여 j 단계의 영상을 복원할 수 있다는 장점이 있다. 예측차 영상과 근사 영상을 구성하기 위한 조직 블록도는 그림 2.9 (b)와 같다. 레벨 j 의 입력영상을 근사 필터(approximation filter)를 거쳐 다운표본화(down_sampler)를 하여 레벨 $j-1$ 단계의 근사 영상을 획득한다. 획득된 레벨 $j-1$ 단계의 근사 영상을 다시 업표본화(upsampler)를 하고 보간 필터(interpolation filter)를 거친 후 레벨 j 의 입력영상과 차를 계산하여 레벨 j 의 예측차 영상을 구하게 된다. 레벨 $j-1$ 단계의 근사 영상과 레벨 j 단계의 예측차 영상으로 레벨 j 단계의 원 영상을 복원할 수 있는데 예측차 영상을 구할 때 사용한 보간 필터를 근사 영상에 적용시킨 후 예측차 영상과 근사 영상을 합하게 되면 원 영상을 구할 수 있다. 그림 2.10은 예측차 영상과 근사 영상을 이용하여 원 영상을 구하는 것을 나타내고 있다.



(a)



(b)

그림 2.9 영상 피라미드

(a)피라미드의 구조 (b) 블록 다이어그램

Fig. 2.9. Image pyramid

(a) Structure of pyramid (b) The block diagram

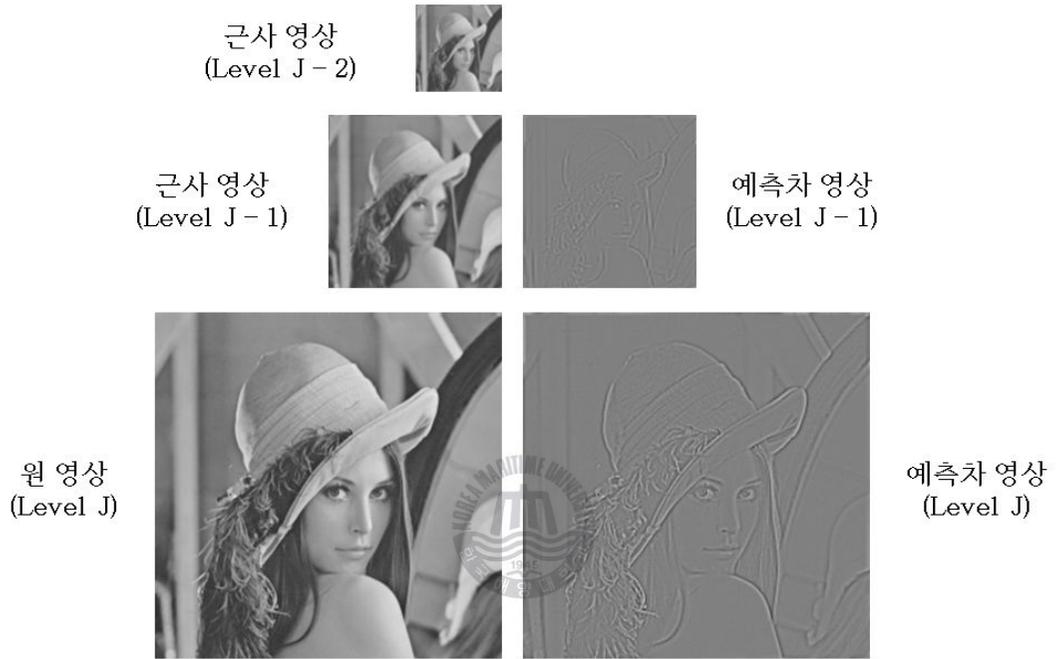


그림 2.10 근사 영상과 예측차 영상

Fig. 2.10. Approximation images and prediction images

제 3 장 유전자 알고리즘과 피라미드 기법을 이용한 활성 윤곽선 모델

활성 윤곽선 모델은 물체의 경계를 효율적으로 추출할 수 있는 장점을 가지고 있지만 초기 윤곽선 문제와 오목한 경계에서의 잘못된 경계를 추출하는 문제점을 발생시킨다. 본 논문에서는 활성 윤곽선 모델에 유전자 알고리즘을 적용하는 방법을 제안하여 초기 윤곽선 문제와 오목한 경계에서의 잘못된 경계 추출의 문제점을 해결한다. 정확도를 높이기 위해 많은 세대를 반복하게 되어 계산복잡도가 높아지는 문제를 해결하기 위해 피라미드 기법을 적용하는 방법을 제안한다. 제안한 방법의 순서도는 그림 3.1과 같다.

3.1 제안한 피라미드 기법

입력받은 원 영상을 피라미드 기법으로 정해진 단계로 영상을 축소하여 저해상도 근사 영상을 획득한다. 평범한 방법으로 영상을 축소시킬 경우 축소된 부분에 대한 영상의 정보가 손실된다. 이를 해결하기 위해 제안한 방법은 원 영상의 정보를 포함하여 근사 영상을 구할 수 있는 가우시안 피라미드를 이용한다. 일반 피라미드와 가우시안 피라미드를 비교한 것은 그림 3.2와 같다. 일반 피라미드인 그림 3.2 (b)의 경우 근사 영상으로 축소됨에 따라 영상의 세밀한 정보가 손실되지만, 가우시안 피라미드인 그림 3.2 (a)의 경우 근사 영상으로 축소됨에도 영상의 세밀한 정보를 포함하는 것을 확인할 수 있다. 예측차 영상을 생성시엔 3차 회선 보간법(bicubic interpolation)을 이용하여 보간한다.

유전자 알고리즘을 이용하여 물체의 경계를 추출한 후 피라미드 기법으로 원 영상의 크기로 확대를 시킨다. 영상 확대 시에도 3차 회선 보간법을 사용한다. 영상 확대시 구해진 물체 경계의 활성 윤곽선 노드의 위치도 같이 확대하게 된다. 축소된 영상을 원 영상의 크기로 확대할 경우 활성 윤곽선은 그림

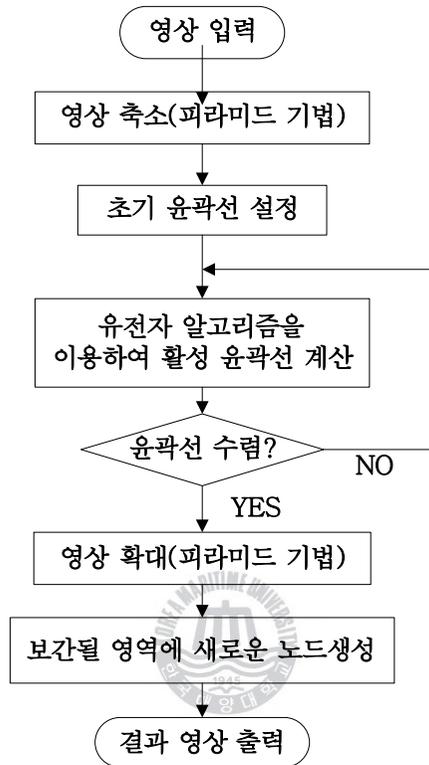


그림 3.1 제안한 방법의 순서도

Fig. 3.1. Flow chart of the proposed method

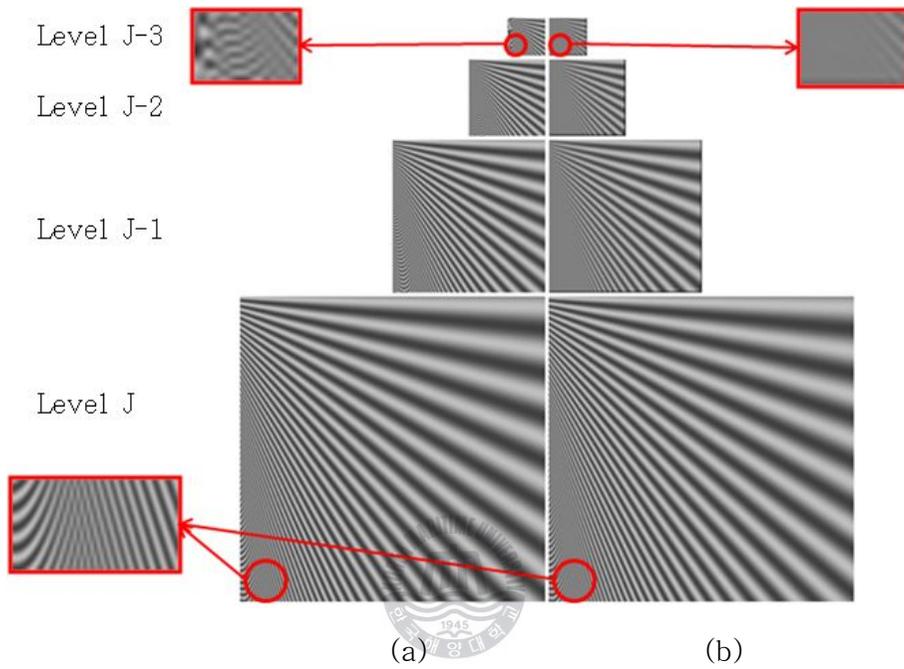


그림 3.2 근사 영상의 비교
 (a) 가우시안 피라미드 (b) 일반 피라미드

Fig. 3.2. Comparison of the approximation image
 (a) Gaussian pyramid (b) Regular pyramid

3.3과 같이 그 간격이 벌어지게 된다. 그림 3.3의 $j-2$ 레벨 영상에서 검정색 사각형이 축소된 근사영상에서 구한 활성 윤곽선의 노드이다. 이를 그림 3.3의 $j-1$ 레벨 영상으로 확대 시키면 $j-2$ 레벨에서 구한 노드의 간격이 벌어지게 된다. 벌어진 윤곽선을 그대로 사용할 경우 그림 3.3의 $j-1$ 레벨 영상의 점선처럼 물체의 경계를 세밀하게 추출하지 못하는 문제점이 발생한다. 제안한 방법은 이를 보완하기 위해 영상 확대시 발생하는 보간 영역에 국부적으로 활성 윤곽선 에너지를 계산하여 그 값이 최소가 되는 지점에 새로운 노드를 생성하고 윤곽선 상에 삽입한다. 그 결과 그림 3.3의 실선처럼 좀 더 정밀하게 물체의 윤곽을 구하게 된다.

3.2 제안한 유전자 알고리즘

3.2.1 염색체구조

피라미드 기법으로 얻어진 저해상도 근사 영상에 활성 윤곽선 모델을 이용해 임의의 초기 윤곽선을 생성한다. 생성된 활성 윤곽선의 노드를 이용하여 활성 윤곽선 모델의 에너지 함수를 구한다. 제안한 방법은 유전자 알고리즘 적용을 위해 윤곽선의 노드를 이진 코드화하여 염색체를 표현한다. 노드를 이진 코드화하면 염색체 구조가 간단해져서 활성 윤곽선 모델에 적용하기 용이해진다. 염색체 집단을 이루는 유전자 개체는 활성 윤곽선 노드의 두 좌표와 임의의 두 좌표 그리고 영상좌표의 최대값과 최소값을 사용하여, 총 6개의 좌표를 이진 코드화 하여 염색체를 구성한다. 이와 같이 염색체를 구성할 경우 기존의 유전 알고리즘을 적용한 방법에 비해 염색체 집단이 줄어들기 때문에 적합도 평가 및 최적 개체를 선택하는 과정에서 시간이 단축되는 효과를 얻을 수 있다. 염색체의 길이는 영상의 크기를 가지고 구하게 되는데 영상의 크기가 $2^n \times 2^n$ 일 경우 n 의 크기만큼 이진 코드의 길이가 정해진다.

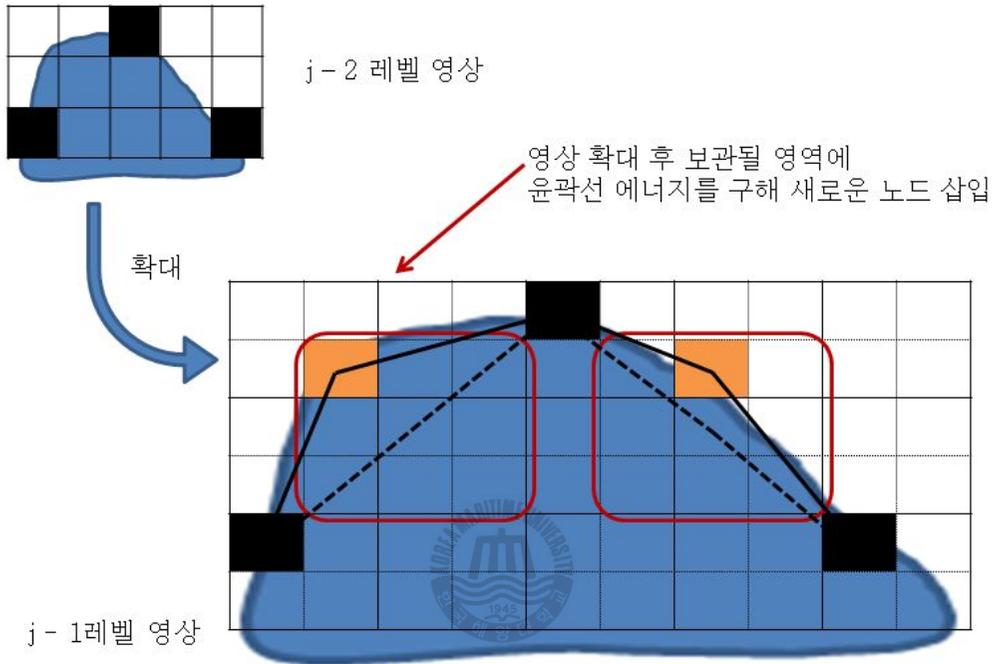


그림 3.3 확대된 영상의 윤곽선 사이에 새로운 좌표 삽입

Fig. 3.3. Insert the new coordinates to a space in enlarge image contours

제안한 방법의 염색체 집단 생성의 예는 그림 3.4와 같다. 활성 윤곽선의 한 노드와 그 다음 노드의 좌표, 임의의 두 좌표 그리고 영상좌표의 최대좌표와 최소좌표를 이진 코드화하여 각 6개씩 염색체를 생성하여 x 축과 y 축 각각의 부모 염색체 집단을 생성한다. x 축과 y 축으로 분리하여 염색체 집단을 생성할 경우 유전연산자 적용 후 자식 염색체 집단을 구성하기 편하고 파라미터 (parameter)수가 적어지는 장점이 있다.

3.2.2 유전연산자

구해진 염색체 집단을 이용하여 유전 연산자에 적용한다. 제안한 방법에서 교배방식은 일점교배(onepoint crossover)방식을 사용하였고, 룰렛 휠(roulette wheel)방식을 적용시켜 교배위치의 확률을 다르게 하여 유전연산자를 계산한다. 일점교배 방식은 유전자 알고리즘의 교배방법 중 가장 기본적인 방식으로 구현이 쉽고 성능 또한 우수하다. 룰렛 휠 방식은 룰렛 휠처럼 선택확률을 다르게 정의되고 그림 3.5와 같이 나타나며 f_n 이 교배점의 위치가 된다.

제안한 방법은 각각의 염색체 집단에 임의로 두 개의 염색체씩 선택하여 교배를 하게 된다. 교배 후 같은 크기의 자식 염색체 집단들이 그림 3.6과 같이 만들어진다. x 축 염색체 집단의 경우 x 축 염색체 집단에서 임의로 두 개의 염색체씩 선택하여 염색체 쌍을 3개를 만들고 룰렛 휠 방식으로 교배점의 위치를 선택한다. 일점 교배 방식으로 교배를 한 뒤 염색체 쌍을 다시 합쳐 새로운 자식 x 축 염색체 집단을 생성한다. y 축 염색체 집단도 같은 방법으로 자식 y 축 염색체 집단을 생성한다. 생성된 자식 염색체 집단들을 순서에 맞게 배열한 후 복호화 하여 6개의 좌표를 생성하고, 활성 윤곽선 에너지를 계산하여 에너지가 가장 낮은 우수한 염색체를 선택하고 나머지 염색체는 소멸시킨다. 선택된 염색체가 활성 윤곽선 모델의 새로운 윤곽선 노드가 된다. 이와 같은 방법을 모든 활성 윤곽선 노드에 적용하여 새로운 활성 윤곽선을 재생성하고 물체의 경

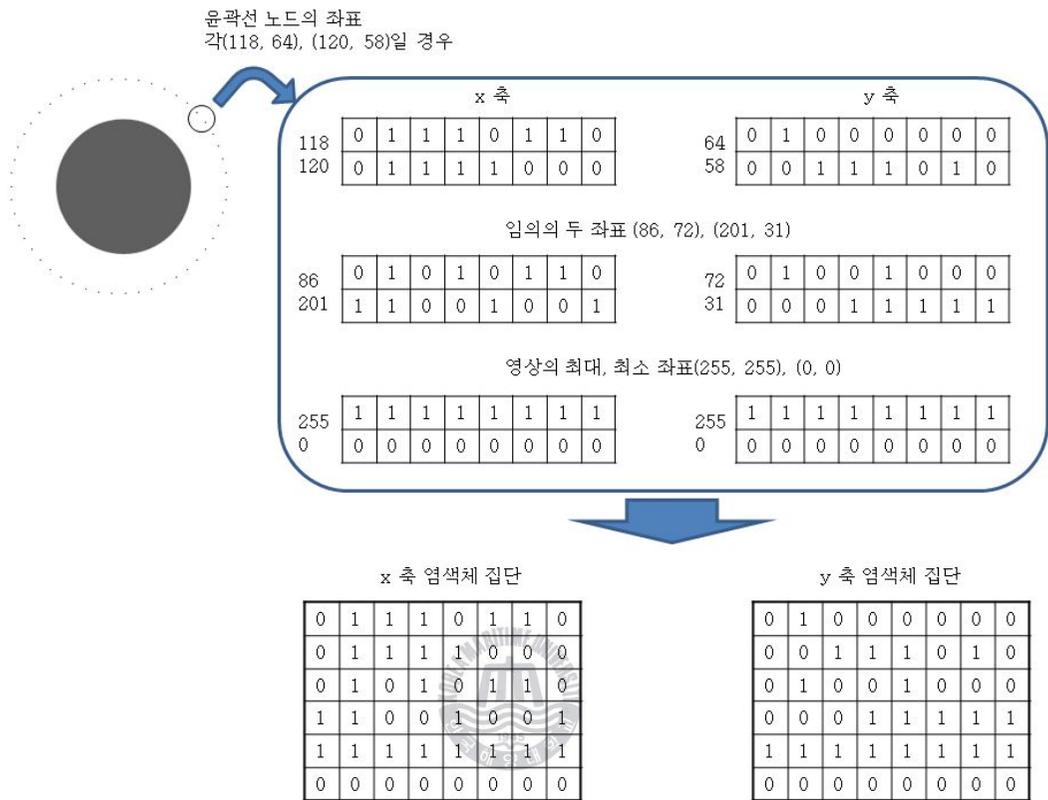


그림 3.4 제안한 방법의 염색체 집단의 예(영상의 크기 $2^8 \times 2^8$)

Fig. 3.4. Examples of the proposed method of the chromosome population(image size $2^8 \times 2^8$)

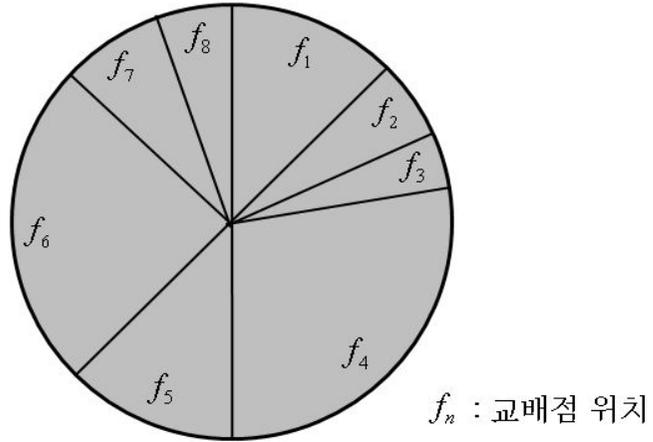


그림 3.5 교배점 선택을 위한 룰렛 휠

Fig. 3.5. Roulette wheel for crossover point selection

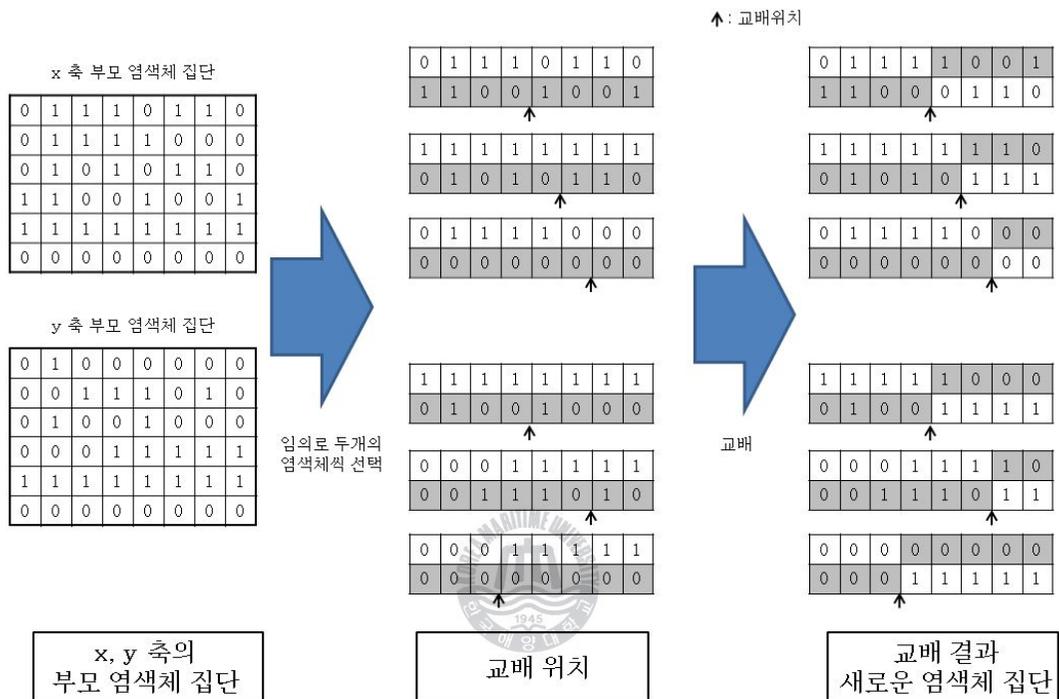


그림 3.6 제안한 교배법의 예

Fig. 3.6. Examples of proposed crossover method

계를 찾을 때까지 반복한다.

제안한 방법에서 적합도 함수는 활성 윤곽선 모델의 에너지 함수를 그대로 적용하여 적합도를 평가하며 식(3.1)과 같다.

$$Fitness = \frac{1}{1 + \left(\sum_v (\alpha E_{cont,v} + \beta E_{curv,v} + \gamma E_{img,v}) \right)} \quad (3.1)$$

윤곽선이 경계를 찾아감에 따라 에너지 함수의 값은 작아지고 적합도 함수의 값은 커지게 된다.



제 4장 실험 및 고찰

제안한 방법에서 계산복잡도가 감소함을 검토하기 위해 별모양 영상을 사용하여 피라미드 기법으로 저해상도로 축소시킨 근사 영상과 원 영상을 가지고 유전연산자 반복횟수를 비교하였다. 별모양 영상을 각각 저해상도로 두 단계 축소시킨 근사 영상과 원 영상을 유전자 알고리즘 반복횟수에 따라 활성 윤곽선의 위치를 그림 4.1과 그림 4.2와 같이 나타낸다. 그림 4.1의 (a)부터 (d)의 경우 10회, 50회, 75회 그리고 100회 반복한 그림이고, 그림 4.2의 (a)부터 (d)의 경우는 100회, 250회, 300회 그리고 500회 반복한 그림이다. 근사 영상의 경우 약 80회 정도의 반복 횟수 만에 물체의 경계에 도달하였으나, 원 영상의 경우 약 500회 정도의 반복 횟수를 거쳐 물체의 경계를 도달하였다. 수행시간 또한 원 영상은 182.67초가 걸렸지만, 근사 영상의 경우 2.744초가 걸려 추출시간이 약 66배 정도로 향상되었다. 이 결과 근사 영상이 원영상보다 계산 복잡도가 감소하는 것을 확인할 수 있다.

제안한 방법에서 유전자 알고리즘을 적용시 초기 윤곽선 위치에 강인함을 확인하기 위해 여러 가지 초기 윤곽선의 위치를 설정한 뒤 기존의 방법과 제안한 방법을 비교했다. 초기 윤곽선의 위치를 다르게 하여 윤곽선 추출결과를 그림 4.3과 같이 나타냈다. 4.3 (a)는 물체 외부에 초기 윤곽선을 설정하였고, 그림 4.3 (b)는 타원 형태로 물체에 걸치게 초기 윤곽선을 설정하였다. 그림 4.3 (c)는 조금 더 긴 타원 형태로 초기 윤곽선을 설정했다. 기존의 방법 중 기울기 벡터 흐름 법으로 윤곽선을 추출한 것은 그림 4.3 (d)부터 그림 4.3 (f)이다. 물체와 가까이 있는 윤곽선의 경우 윤곽을 찾지만 멀리 있는 윤곽선의 경우는 찾지 못하는 결과를 나타낸다. 그림 4.3 (g)부터 그림 4.3 (i)는 제안한 방법으로 윤곽을 추출한 결과를 나타낸다. 초기 윤곽선의 위치에 관계없이 윤곽이 추출됨을 알 수 있다.

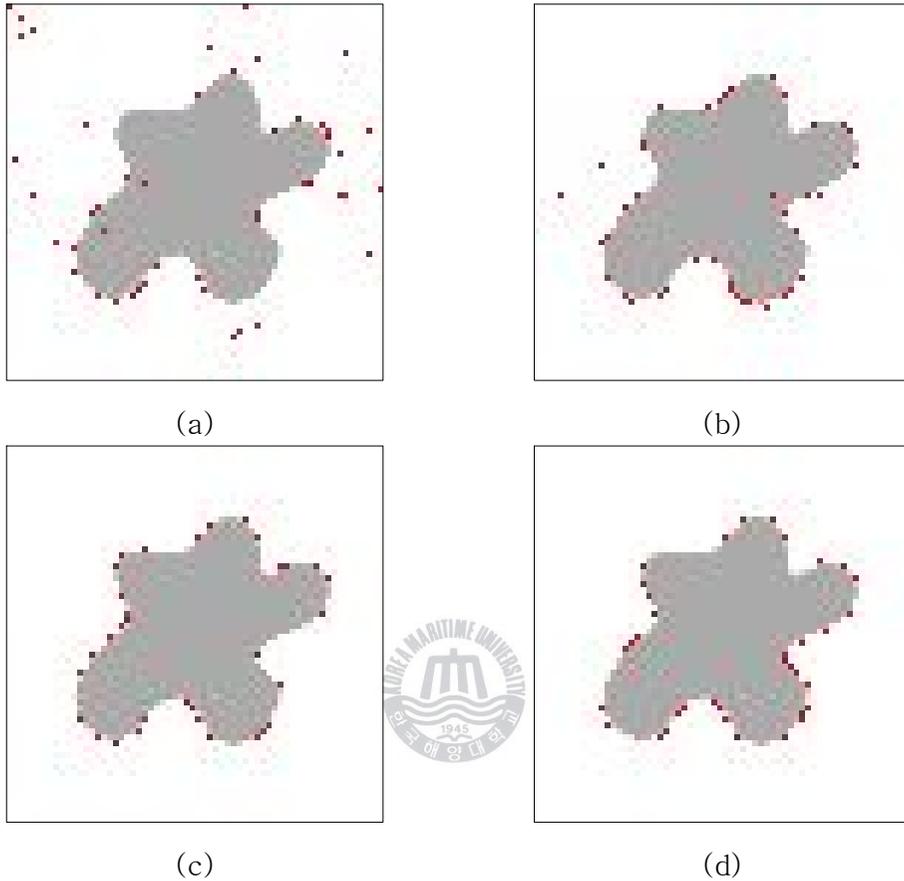


그림 4.1 근사 영상에서 반복횟수에 따른 노드 분포(2단계 축소)
 (a) 10번 반복 (b) 50번 반복 (c) 75번 반복 (d) 100번 반복

Fig. 4.1. Distribution of nodes according to the number of iterations in approximation image(two step reduction)
 (a) 10 times iteration (b) 50 times iteration
 (c) 75 times iteration (d) 100 times iteration

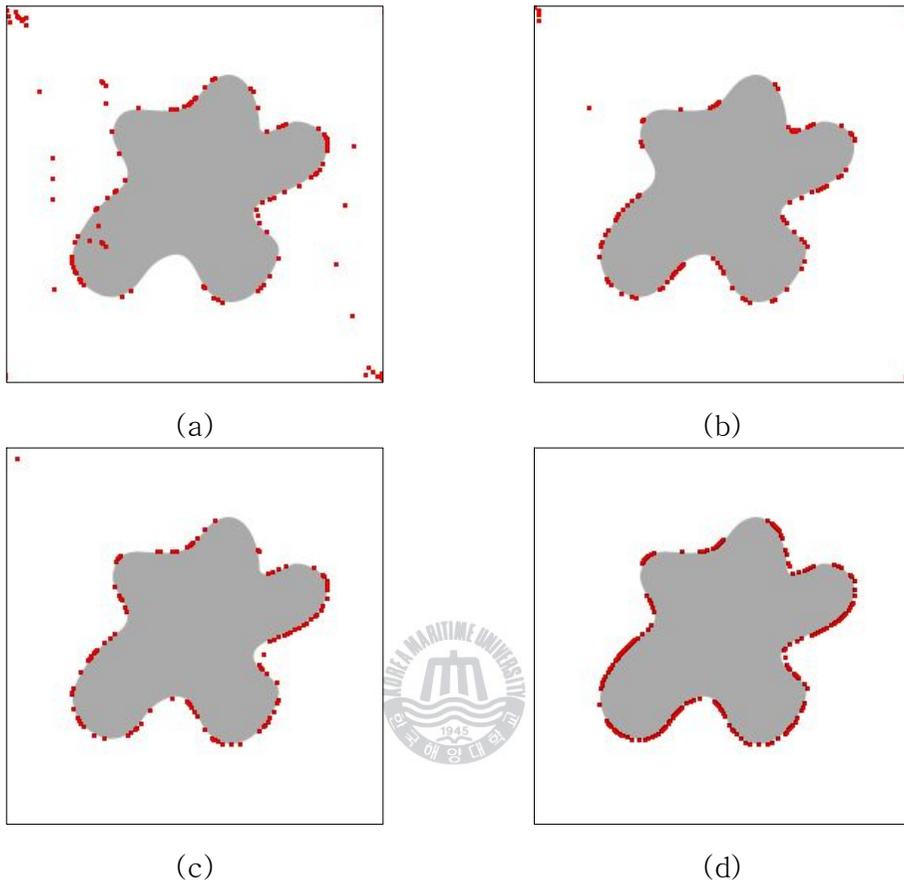


그림 4.2 원 영상에서 반복횟수에 따른 활성 윤곽선 노드 분포
 (a) 100번 반복 (b) 250번 반복 (c) 300번 반복 (d) 500번 반복

Fig. 4.2. Distribution of nodes according to the number of iterations in original image

- (a) 100 times iteration (b) 250 times iteration
- (c) 300 times iteration (d) 500 times iteration

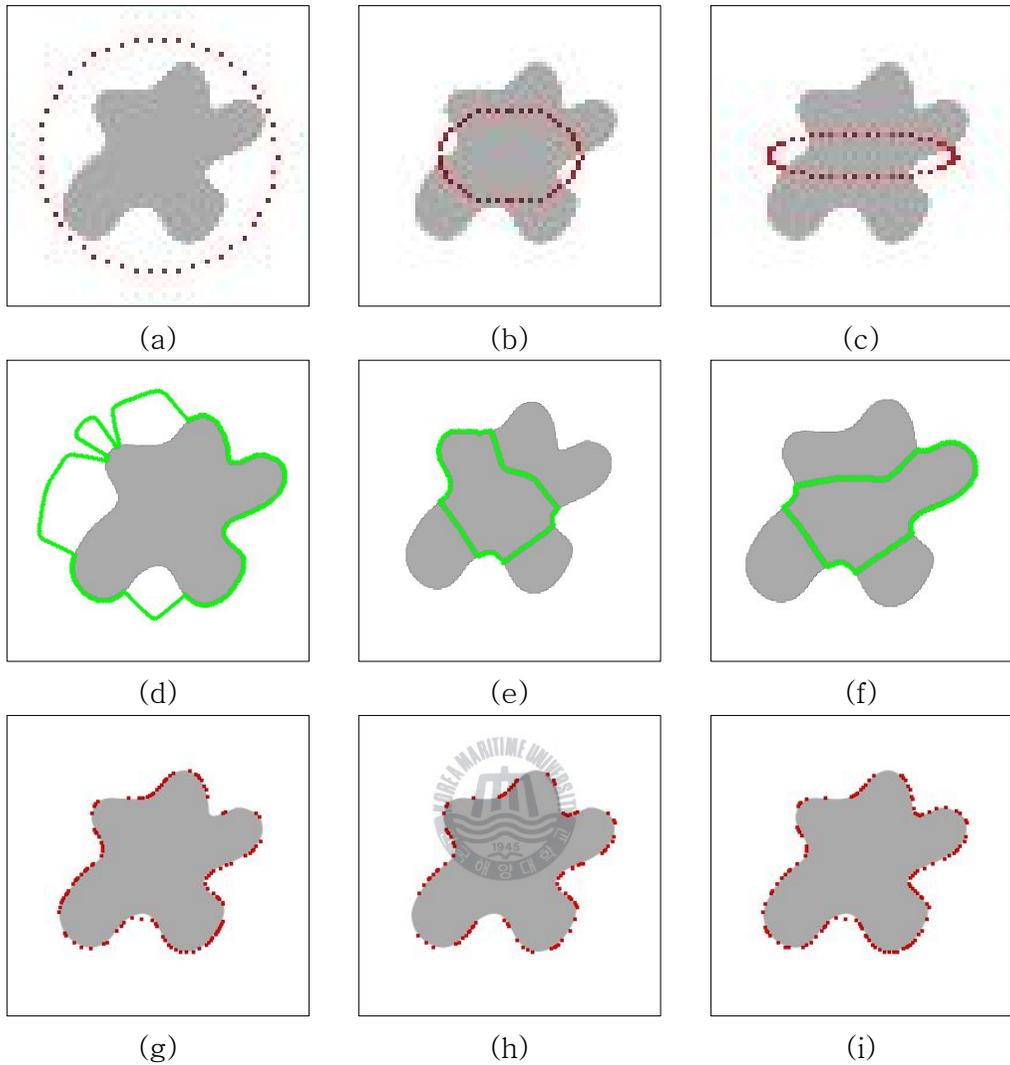


그림 4.3 초기윤곽선 위치에 따른 윤곽 추출

(a)~(c) 여러 가지 초기 윤곽선 (d)~(f) 기울기 벡터 흐름 법으로 구한 결과
(g)~(i) 제안한 방법의 결과

Fig. 4.3. Contours extraction according to the initial curve
(a)~(c) Various initial contour (d)~(f) Results obtained with gradient
vector flow method (g)~(i) Results of proposed method

제안한 방법의 적합도 성능을 별모양 영상을 이용하여 비교하였다. 유전자 알고리즘의 반복횟수에 따른 적합도를 원영상과 근사 영상으로 나누어 그래프로 비교한 것은 그림 4.4와 같다. 초기 적합도는 원영상과 근사 영상 모두 50%정도이고 알고리즘이 반복됨에 따라 적합도가 향상되는 것을 볼 수 있다. 원영상의 경우 그래프 곡선이 완만하게 증가하는 것을 볼 수 있지만 근사 영상의 경우 초기에 그래프 곡선이 급격히 증가하여 높은 적합도를 나타내는 것을 확인할 수 있다. 또한 반복횟수는 0.8의 적합도의 경우 원영상은 약 300회 반복해야 도달하지만, 근사 영상은 약 50회 만 반복해도 적합도를 만족하는 것을 확인할 수 있다. 그래프 전체를 비교할 경우 근사 영상이 원 영상의 적합도에 비해 높다는 것을 확인할 수 있다.

제안한 방법을 여러 형태의 물체 영상에 적용하여 윤곽선을 추출하였다. 제안한 방법을 적용하여 구한 윤곽선이 그림 4.5와 그림 4.6에 나타난다. 그림 각각의 (a)는 원 영상이고 (b)는 두 단계 축소한 근사 영상이며 (c)는 초기윤곽선을 설정한 그림이다. 그리고 (d)는 최종 결과 영상을 나타낸다. 물체의 형태가 바뀌어도 윤곽을 정확히 추출하는 것을 확인할 수 있다.

제안한 방법을 실제영상에도 적용됨을 확인한다. 먼저 인물사진의 얼굴윤곽을 추출에 제안한 방법을 적용하였으며 그림 4.7과 같이 나타난다. 그림 4.7 (a), (c) 그리고 (e)는 인물의 원 영상을 나타내고 그림 5.7 (b), (d) 그리고 (f)는 얼굴윤곽을 추출한 결과를 나타낸다. 얼굴 윤곽이 효율적으로 추출됨을 확인할 수 있다. 제안한 방법으로 물체의 윤곽을 추출하였으며 그림 4.8과 같이 나타난다. 그림 4.8 (a)와 그림 4.8 (b)는 물체의 원 영상을 나타내고 그림 4.8 (c)와 그림 4.8 (d)는 물체의 윤곽을 추출한 사진이다. 물체의 윤곽 또한 효율적으로 추출됨을 확인할 수 있다.

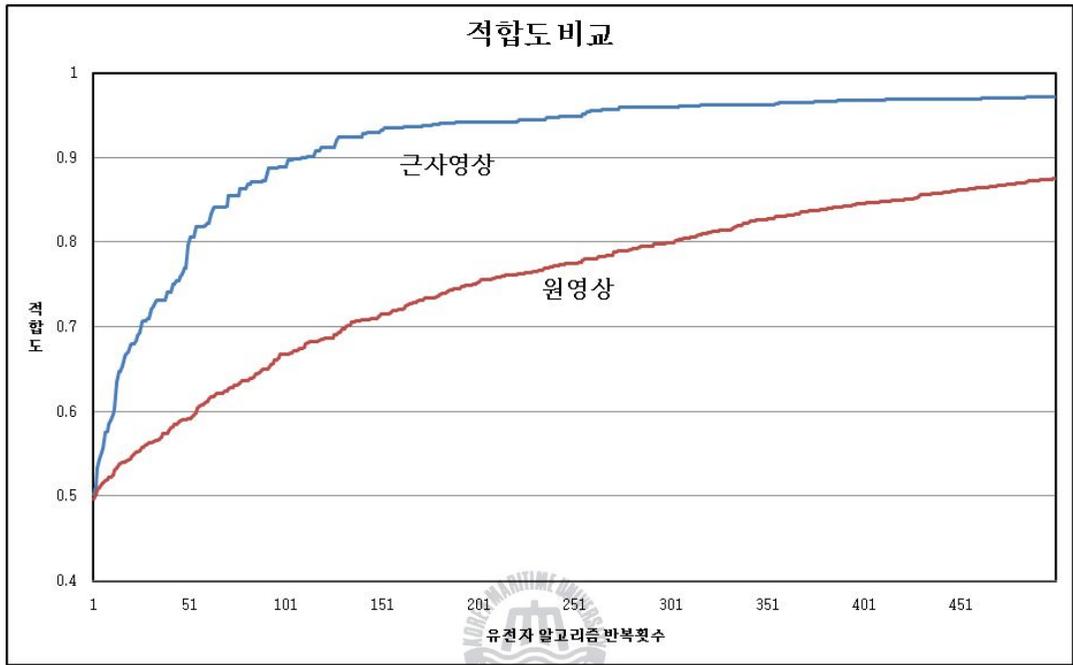


그림 4.4 유전자 알고리즘의 반복횟수에 따른 적합도

Fig. 4.4. Fitness according to the number of iteration of genetic algorithm

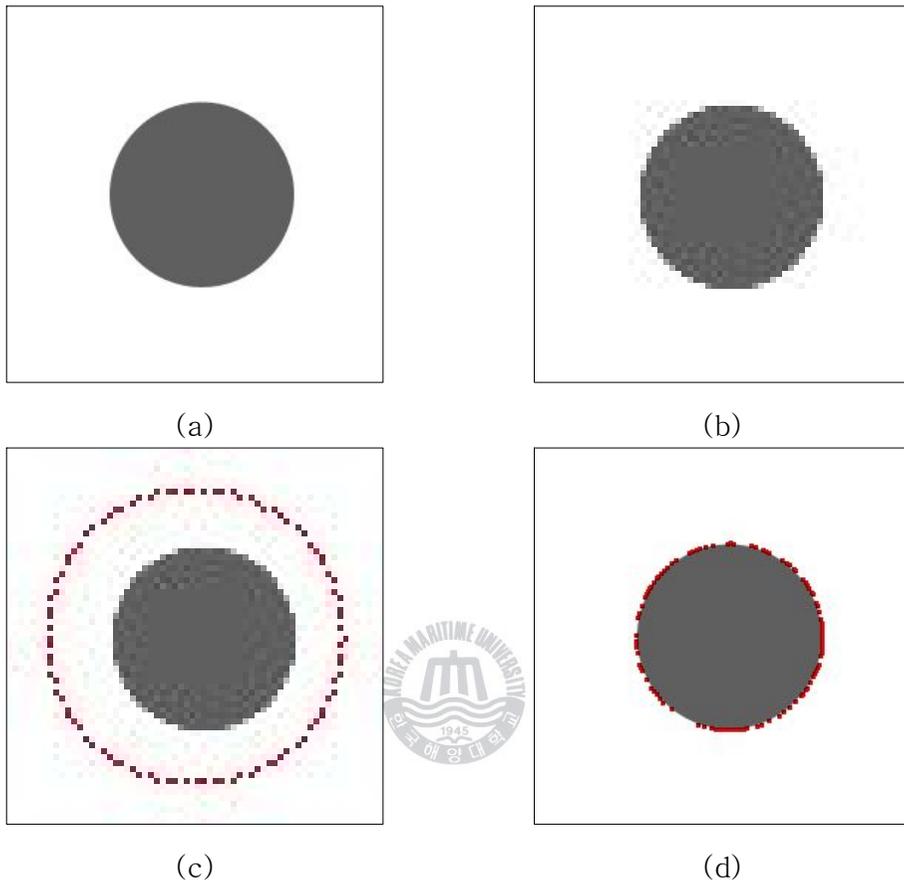


그림 4.5 제안한 방법을 적용하여 구한 물체 윤곽선(영상 1)
 (a) 원 영상 (b) 두 단계 축소한 근사 영상
 (c) 초기 윤곽선 설정 (d) 결과 영상

Fig. 4.5. Object contour with the proposed method(image 1)
 (a) Original image (b) two step approximation image
 (c) Initial contour (d) Result image

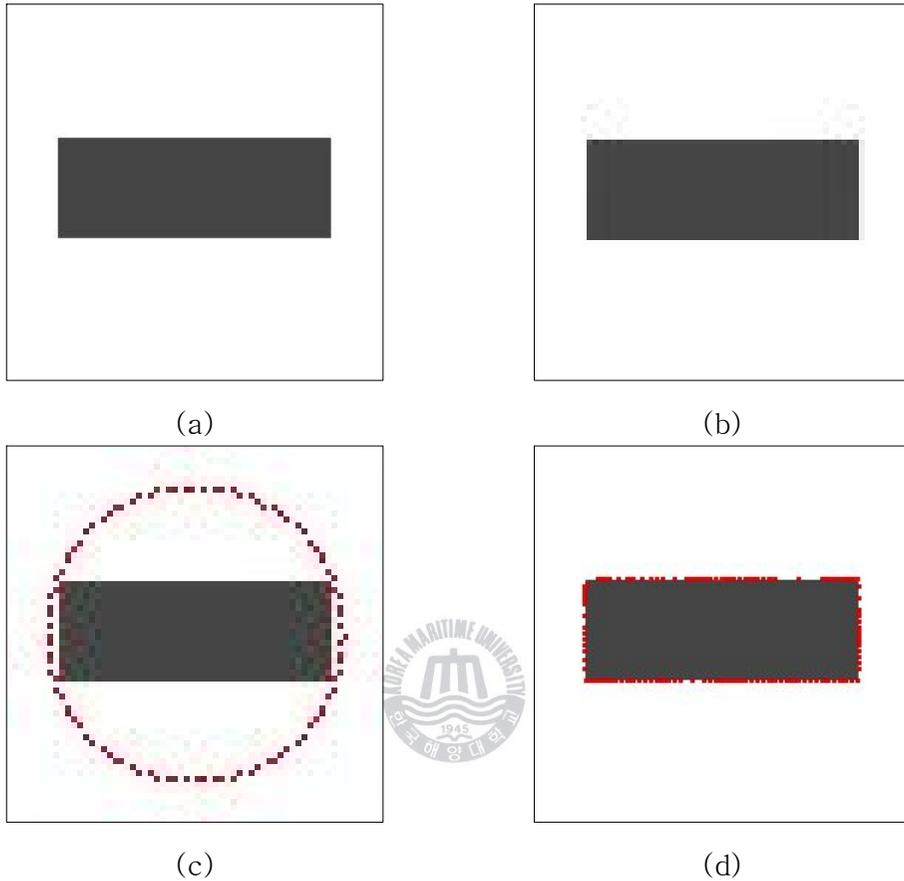


그림 4.6 제안한 방법을 적용하여 구한 물체 윤곽선(영상 2)
 (a) 원 영상 (b) 두 단계 축소한 근사 영상
 (c) 초기 윤곽선 설정 (d) 결과 영상

Fig. 4.6. Object contour with the proposed method(image 2)
 (a) Original image (b) two step approximation image
 (c) Initial contour (d) Result image



그림 4.7 제안한 방법을 적용한 얼굴 윤곽 추출 결과
 (a) 원 영상(얼굴 1) (b) 원 영상(얼굴 2) (c) 원 영상(얼굴 3)
 (d) 결과 영상(얼굴 1) (e) 결과 영상(얼굴 2) (f) 결과 영상(얼굴 3)

Fig. 4.7. The result of face contour detection with the proposed method
 (a) Original image(face 1) (b) Original image(face 2)
 (c) Original image(face 3) (d) Results image(face 1)
 (e) Result image(face 2) (f) Results image(face 3)



(a)



(b)



(c)



(d)

그림 4.8 제안한 방법을 적용한 물체 윤곽 추출 결과

(a) 원 영상(물체 1) (b) 원 영상(물체 2)

(c) 결과 영상(물체 1) (d) 결과 영상(물체 2)

Fig. 4.8. The result of object contour detection with the proposed method

(a) Original image(object 1) (b) Result image(object 1)

(a) Original image(object 2) (b) Result image(object 2)

제 5 장 결 론

본 논문에서는 유전자 알고리즘과 다해상도 기법을 이용한 활성 윤곽선 모델 기반의 물체 추출방법을 제안하였다. 제안한 방법은 물체의 경계를 추출하기 위해 활성 윤곽선 모델을 사용하였고, 초기 윤곽선 문제와 특정 물체에서 윤곽선 수렴문제를 해결하기 위해 유전자 알고리즘을 사용하였다. 정확도를 높이기 위해 많은 세대가 반복되어 계산복잡도가 증가하는 문제가 발생하는데 이를 다해상도 알고리즘인 피라미드 기법을 이용하여 저해상도 근사 영상에 알고리즘을 적용하여 계산복잡도 문제를 해결하였다. 원 영상으로 확대시 벌어진 윤곽선 간격에 국부적으로 활성 윤곽선 에너지를 계산하고 윤곽선사이를 보간하여 정밀도를 향상시켰다.

제안한 방법은 유전자 알고리즘의 큰 탐색영역을 가지는 특징을 활성 윤곽선 모델에 적용하여 기존의 문제점인 초기 윤곽선 설정 문제를 해결하였고, 오펜한 경계등 특정한 물체의 경계를 효과적으로 추출하였다. 윤곽선의 노드를 이진 코드화 한 뒤 연색체를 정의하여 연색체 표현을 간소화하였다. 그리고 기존의 유전자 알고리즘을 적용한 활성 윤곽선 모델의 문제점을 어느정도 해결하였다.

유전자 알고리즘 적용할 경우 계산복잡도가 증가하는 문제를 피라미드 알고리즘으로 영상을 저해상도 근사 영상으로 크기를 축소시킨 후 계산하여 해결하였다. 가우시안 피라미드 기법을 적용하여 원 영상의 정보손실을 최소화 하여 정확한 근사 영상을 얻을 수 있었다.

원 영상의 크기로 확대 보간시 윤곽선 간격이 멀어져 정밀하게 추출되지 못하는 문제를 국부적인 활성 윤곽선 에너지 계산하여 새로운 활성 윤곽선 노드를 생성, 삽입하여 정밀도를 향상시켰다. 그리고 제안한 방법을 실제 영상에 적용해 여러 형태의 물체 추출에도 효과적임을 확인하였다.

그러나 물체의 윤곽이 복잡할 경우 정밀하게 찾지 못하는 단점이 있었다. 따

라서 윤곽이 복잡한 부분과 윤곽이 단순한 부분을 분리한 뒤 부분적으로 알고리즘을 적용하는 방법을 연구해야겠고, 에너지 함수에 쓰이는 적당한 가중치를 자동으로 산출할 수 있는 방법과 제안한 방법의 정확도를 표현할 수 있는 객관적인 평가함수에 관한 연구도 계속 되어야 하겠다.



참고 문헌

- [1] R. C. Gonzalez and R. E. Woods, *Digital image processing, Third edition*, Prentice-Hall, 2008.
- [2] C. Li, C. Xu, C. Gui, and M. D. Fox, "Level set evolution without re-initialization: a new variation formulation," *Computer Vision and Pattern Recognition*, vol. 1, pp. 430-436, 2005.
- [3] 김대희, 호요성, "Level Set 방법을 이용한 영상분할 알고리즘," 대한전자공학회 논문지, vol. 40, no. 5, pp. 1-9, 2003.
- [4] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: active contour models," *International Journal of Computer Vision*, vol. 1, pp. 321-331, 1987.
- [5] A. A. Amini, S. Tehrani, and T. E. Weymouth, "Using dynamic programming for minimizing the energy of active contours in the presence of hard constraints," *Second International Conference on Computer Vision*, vol. 1, pp. 95-99, 1988.
- [6] D. J. Williams and M. Shah, "A fast algorithm for active contours and curvature estimation," *Computer Vision, Graphic, and Image Processing: Image Understanding*, vol. 55, no. 1, pp. 14-26, 1992.
- [7] C. Xu and J. L. Prince, "Snakes, shapes, and gradient vector flow," *IEEE Transactions on Image Processing*, vol. 7, pp. 359-369, 1998.

- [8] L. A. MacEachern and T. Manku, "Genetic algorithms for active contour optimization," *IEEE International Symposium on Circuits and Systems*, vol. 4, pp. 229-232, 1998.
- [9] K. J. Mun, H. T. Kang, H. S. Lee, Y. S. Yoon, C. M. Lee, and J. H. Park, "Active contour model based object contour detection using genetic algorithm with wavelet based image processing," *International of Control, Automation, and Systems*, vol. 2, no. 1, pp. 100-106, 2004.
- [10] J. J. Rousselle, N. Vincent, and N. Verbeke, "Genetic algorithm to set active contour," *Computer Analysis of Images and Patterns*, vol. 2756, pp. 345-352, 2003.
- [11] 박선영, 박주영, S. Grosskopf, 김명희, "유전자 알고리즘을 이용한 활성 윤곽선모델의 에너지 최소화 기법," 한국정보과학회 발표논문집, vol. 24, no. 2, pp. 709-712, 1997.
- [12] 배대규, 한준희, "유전자 알고리즘을 이용한 윤곽선 추출," 한국정보과학회 논문지, vol. 24, no. 10, pp. 1063-1072, 1997.
- [13] M. Xiao, S. Xia, and S. Wang, "Geometric Active Contour Model with Color and Intensity Priors for Medical Image Segmentation", *Engineering in Medicine and Biology Society*, vol. 1, pp. 6496-6499, 2006.
- [14] C. Y. Hsu, C. Y. Liu, and C. M. Chen, "Automatic segmentation of

liver PET images," *Computerized Medical Imaging and Graphics*, vol. 32, no. 7, pp. 601-610, 2008.

[15] L. B. Theng and C. A. Ling, "GANN snake for object extractions from high resolution satellite imagery," *International Conference on Advanced Computer Theory and Engineering*, pp. 1005-1009, 2008.

[16] 김문수, 피부색과 활성 윤곽 모델을 이용한 얼굴 영역 추출, 한국해양대학교 대학원, 2000.

[17] 문창수, 유봉길, 오승제, 정종필, 전희정, "Active contour model을 이용한 임의의 물체 윤곽선 추출," 한국컴퓨터정보학회 논문지, vol. 4, no. 4, pp. 77-85, 1999.



[18] B. Leroy, I. Herlin, and L. D. Cohen, "Multi-resolution algorithms for active contour models," *International Conference on Analysis and Optimization of Systems*, pp. 58-65. 1996.

[19] L. D. Cohen, "On active contour models and balloons," *Computer Vision, Graphic, and Image Processing : Image Understanding*, vol. 53, pp. 211-218, 1991.

[20] L. D. Cohen and J. L. Prince, "Finite-element methods for active contour models and balloons for 2-D and 3-D images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1131-1147, 1993.

- [21] C. Davatzikos and J. L. Prince, "An active contour model for mapping the cortex," *IEEE Transactions on Medical Imaging*, vol. 14, pp 65-80, 1995.
- [22] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Transactions on Image Processing*, vol. 10, no. 2, pp. 266-277, 2001.
- [23] D. D. Giusto, F. Massidda, and C. Perra, "fast active-contour curvature-based evolution," *Signal Processing : Image Communication*, vol. 19, pp. 517-538, 2004.
- [24] C. Xu, A. Y. Jr, and J. L. Prince, "On the Relationship between Parametric and Geometric Active Contours," *In Proc. of 34th Asilomar Conference on Signal, Systems, and Computers*, pp. 483-489, 2000.
- [25] 김대희, 안충현, 호요성, "모델공간 모델을 이용한 영상분할 알고리즘," *대한전자공학회 논문지*, vol. 41, no. 2, pp. 41-50, 2004.
- [26] 진강규, *유전 알고리즘과 그 응용*, 교우사, 2000.