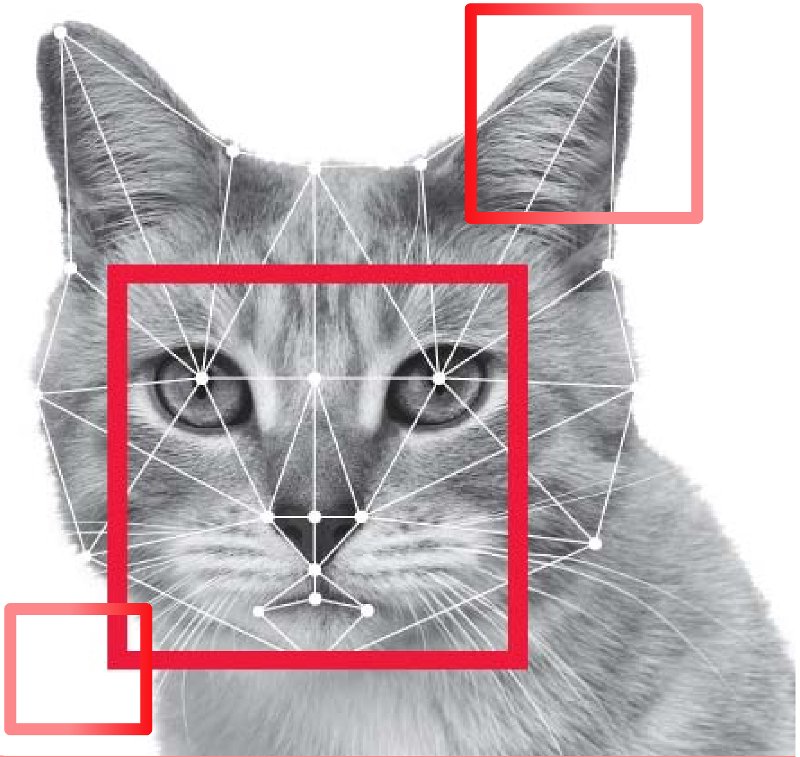
한빛아카데미
Hanbit Academy, Inc.

COMPUTER VISION 컴퓨터 비전

기본 개념부터 최신 모바일 응용 예까지



5장. 영상 분할

PREVIEW

■ 사람 뇌의 영상 분할

(a) 원래 영상 (b) 분할된 영상

그림 5-1 맨 왼쪽 영상을 보고 장면을 서술해 보자.

- "상자 위에 쌓여있는 형형색색의 파프리카"라고 해석하는 과정에서 상자, 파프리카, 가격표, 글자와 같이 의미 있는 영역으로 분할
- 분할과 인식이 동시에 일어남

■ 컴퓨터 비전

- 현재는 분할 후 인식하는 순차 처리(동시 수행을 추구하는 연구도 있으나 초보 단계)
- 가장 어려운 문제 중 하나

각 절에서 다루는 내용

1. 영상 분할의 원리

→ 영상에서 영역을 분할하는 기본 원리를 먼저 살펴본다.

2. 전통적 방법

→ 초창기 주류 알고리즘을 소개한다. 주로 간단한 규칙을 사용하는 알고리즘이다.

3. 그래프 방법

→ 그래프 알고리즘과 같이 현재 널리 활용되는 전역 최적화에 기반을 둔 접근 방법을 살펴본다.

4. 민시프트

→ 민시프트를 이용한 영역 분할 원리를 알아본다.

5. 워터셰드

→ 워터셰드를 이용한 영역 분할 원리를 알아본다.

6. 대화식 물체 분할

→ 사용자가 대화식으로 관심있는 물체를 분할하는 기능을 구현하는 방법을 공부한다.

7. 알고리즘 선택

→ 지금까지 다룬 알고리즘을 평가하고 상황에 맞는 알고리즘을 알아내는 데 쓸 수 있는 데이터베이스들을 살펴본다.

5.1 영상 분할의 원리

분할의 정의

- 식 (5.1)은 엄밀성보다는 개념적인 정의

$$
\begin{aligned}
&r_i \cap r_j = \varnothing,\ 1 \le i,\ j \le n, i \ne j \\
&\bigcup_{i=1,n} r_i = f \\
&Q(r_i) = \text{참},\ 1 \le i \le n \\
&Q(r_i \cup r_j) = \text{거짓},\ 1 \le i,\ j \le n,\ r_i\text{와 } r_j\text{는 이웃한 영역}
\end{aligned}
\tag{5.1}
$$

Q: 영역의 균일성을 측정하는 어떤 기준

생각해 볼 점

- 적절한 분할이란?
  - 저분할과 과분할
- 사람 vs. 컴퓨터
  - 사람은 선택적 주의집중과 능동 비전 기능을 가지며, 분할 과정에서 고급 지식 사용
    - 물체 모델, 지식, 의도 등
  - 컴퓨터 비전은 그런 수준에 이르지 못함
    - 분할이 끝나야만 고급 지식을 이용하여 인식을 수행

5.1 영상 분할의 원리

- 분할의 어려움
  - 이웃 화소 몇 개를 보고 자신이 영역의 내부인지 경계인지 판단할 수 있을까?

    → 전역 연산 필요성

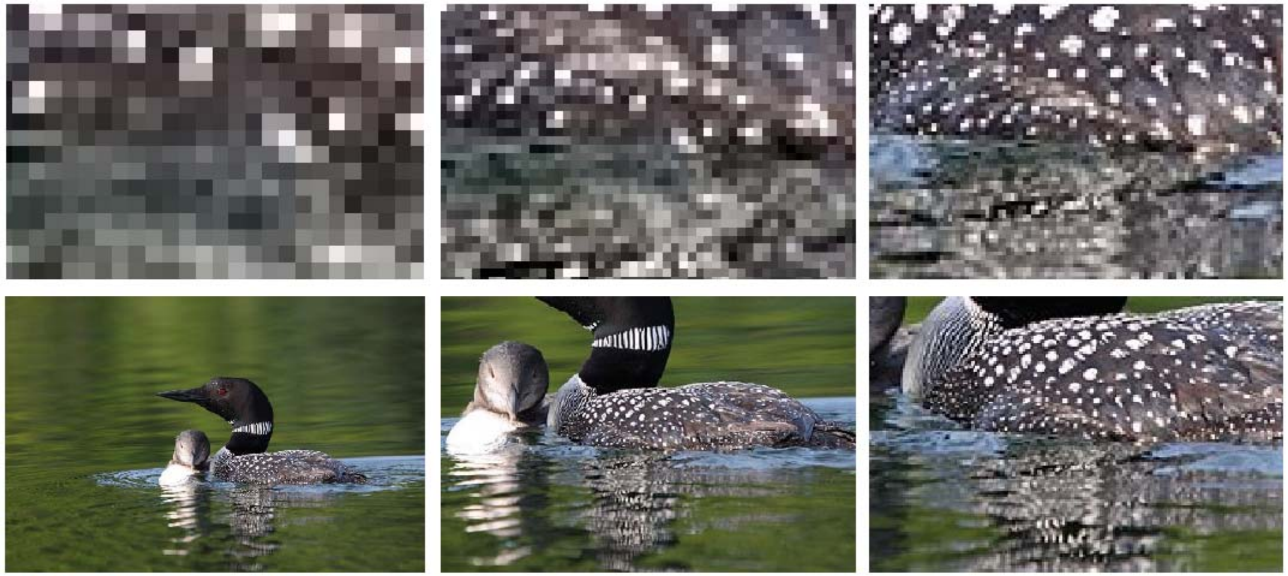


그림 5-2 영상 분할의 근원적인 어려움

5.1 영상 분할의 원리

- 에지(3장) vs. 영역
  - 개념적으로 에지는 영역의 경계에 해당
  - 하지만 에지 검출로는 한계
    - 거짓 긍정, 거짓 부정 → 폐곡선을 이루지 못함

- 영역 vs. 지역 특징(4장)
  - 사람은 지역 특징보다 영역 분할에 훨씬 뛰어남
  - 반면 컴퓨터 비전은 영역보다 지역 특징으로 문제 해결하는 사례 많음
  - 미래는?

5.2 전통적 방법

- 동작 조건
  - 특수 조건 만족하거나 단순한 영상에서만 작동
    - 예) 공장 자동화, 문서 인식 등
    - 문제가 쉽다면 굳이 복잡한 알고리즘 쓸 필요 없음
  - 자연 영상에서 매우 낮은 성능

5.2.1 임계화를 이용한 영역 분할

5.2.2 군집화를 이용한 영역 분할

5.2.3 분할합병

5.2.1 임계화를 이용한 영역 분할

- 이진화를 이용한 영역 분할
  - 문서 영상의 경우 오츄 이진화(2.3.1절)는 훌륭한 영상 분할 알고리즘
  - 하지만 명암 단계가 둘 이상인 경우는 오작동

- 삼진화로 확장
  - 이중 임계값 사용

μ_g: 히스토그램의 전체 평균

부류의 평균값들에 대한 분산으로
세 평균이 다를수록 좋음

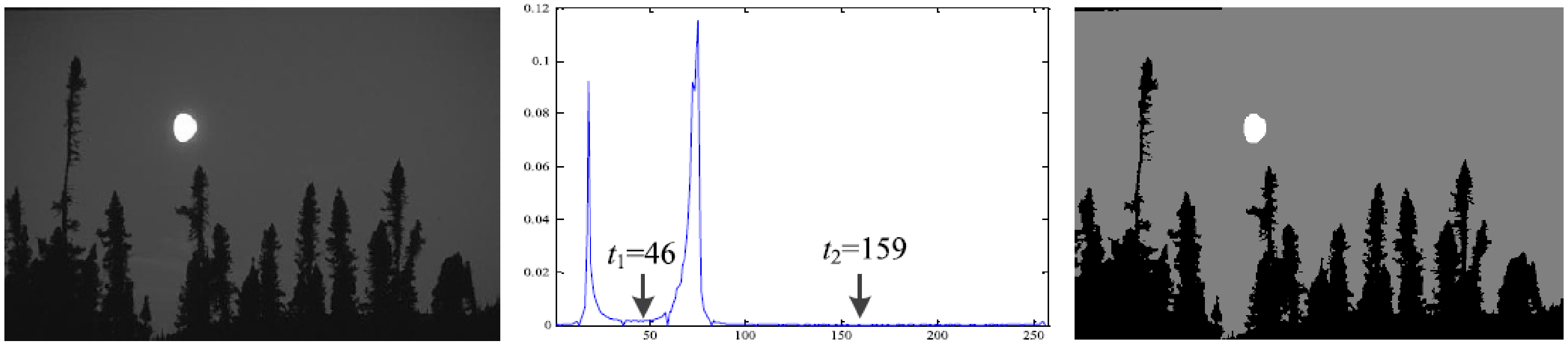
$$v_{between}(t_1,t_2) = w_0(\mu_0-\mu_g)^2 + w_1(\mu_1-\mu_g)^2 + w_2(\mu_2-\mu_g)^2$$

$$\text{이때, } \mu_0 = \frac{1}{w_0}\sum_{i=0}^{t_1} i\hat{h}_i,\ \ \mu_1 = \frac{1}{w_1}\sum_{i=t_1+1}^{t_2} i\hat{h}_i,\ \ \mu_2 = \frac{1}{w_2}\sum_{i=t_2+1}^{L-1} i\hat{h}_i \qquad (5.2)$$

$$w_0 = \sum_{i=0}^{t_1}\hat{h}_i,\ \ w_1 = \sum_{i=t_1+1}^{t_2}\hat{h}_i,\ \ w_2 = \sum_{i=t_2+1}^{L-1}\hat{h}_i$$

$$(\dot{t}_1,\dot{t}_2) = \underset{0<t_1<t_2<L-1}{\operatorname{argmax}}\ v_{between}(t_1,t_2) \qquad (5.3)$$

5.2.1 임계화를 이용한 영역 분할



(a) 원래 영상 (b) 명암 히스토그램 (c) 이중 임계값으로 분할한 영상

그림 5-3 이중 임계값 오츄 알고리즘에 의한 영상 분할

알고리즘 5-1 이중 임계값 오츄 알고리즘

입력 : 영상 $f(j,i)$, $0 \le j \le M-1$, $0 \le i \le N-1$

출력 : 삼진영상 $g(j,i)$, $0 \le j \le M-1$, $0 \le i \le N-1$ // 0, 1, 2 세 가지 값을 가진 영상

```
1  [알고리즘 2-1]을 이용하여 f의 정규 히스토그램 ĥ을 만든다.
2  for(t1=1 to L-3)
3    for(t2=t1+1 to L-2)
4      식 (5.2)를 이용하여 v_between(t1, t2)를 계산한다.
5  2~4행에서 가장 큰 v_between을 생성한 (t1, t2)를 임계값 (t̂1, t̂2)로 취한다.
6  (t̂1, t̂2)로 f를 삼진화하여 g를 만든다.
```

5.2.1 임계화를 이용한 영역 분할

- 적응적 임계화
  - 하나 또는 두 개의 임계값을 영상 전체에 적용하는 전역 방법의 한계
    - 지역적으로 명암 분포가 다른 경우 낮은 분할 품질

(a) 원래 영상 (b) [알고리즘 2-4]를 적용한 결과 (c) [알고리즘 5-1]을 적용한 결과

그림 5-4 효모 영상과 임계화한 영상

  - 적응적 임계화로 해결: 지역에 따라 적응적으로 임계값 결정
    - $t(j,i)$를 어떻게 결정할까?

t(j,i): (j, i) 위치 이웃 값을 보고 결정
(ex.) 평균 & 표준편차 이용

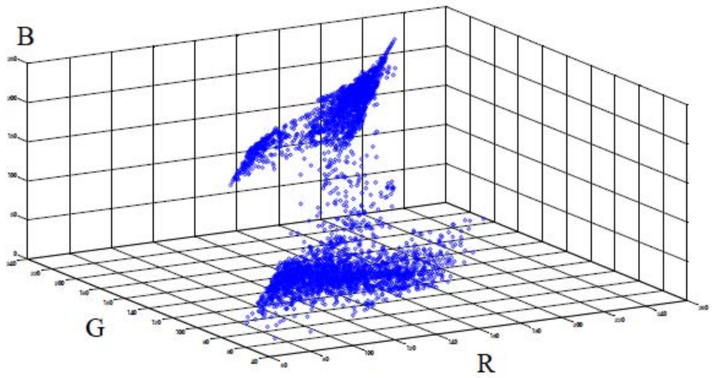
$$b(j,i) = \begin{cases} 1, & f(j,i) \geq t(j,i) \\ 0, & \text{그렇지 않으면} \end{cases} \tag{5.4}$$

5.2.2 군집화를 이용한 영역 분할

군집화

- 화소를 RGB 3차원 컬러 공간으로 매핑 한후, k-means 로 군집화

(a) 원래 영상

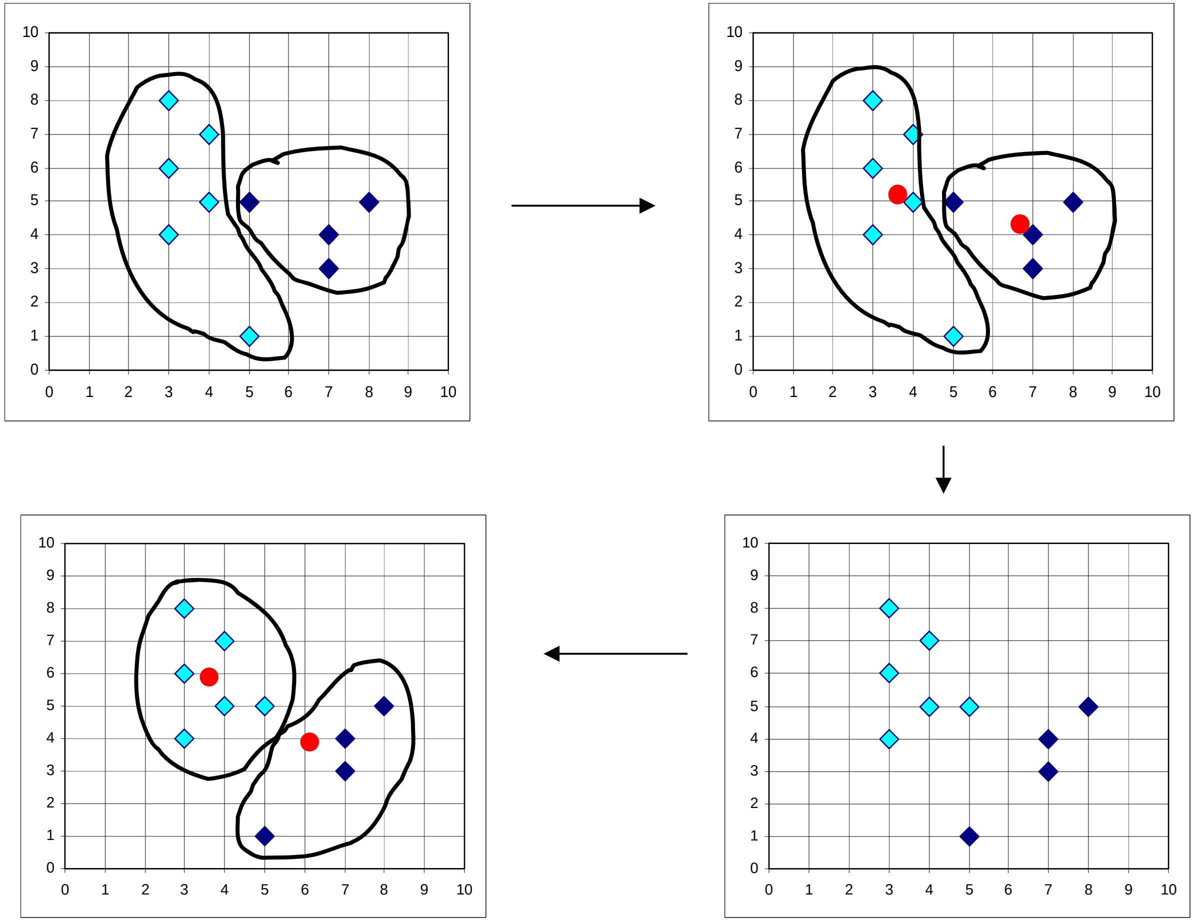


(b) 3차원 공간에 매핑한 결과

그림 5-5 컬러 영상 화소를 RGB 공간에 매핑

5.2.2 군집화를 이용한 영역 분할

- K-means clustering



5.2.2 군집화를 이용한 영역 분할

알고리즘 5-2 *k*-means를 이용한 컬러 영상 분할

입력 : 컬러 영상 $f_r(j,i)$, $f_g(j,i)$, $f_b(j,i)$, $0 \le j \le M-1$, $0 \le i \le N-1$, k(군집의 개수)

출력 : $[1, 2, 3, \cdots, k]$로 번호가 매겨진 영상

1 화소를 $\mathbf{x}_i=(r_i, g_i, b_i)$ 형태로 변환하여 샘플 집합 $X=\{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$을 구성한다. // $n=MN$

2

3 // 4~9행은 k-means 군집화

4 k개의 군집 중심 $Z=\{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_k\}$를 초기화 한다.

5 while(TRUE) {

6   for(i=1 to n) $\mathbf{x}_i$를 가장 가까운 군집 중심에 배정한다.

7   if(이 배정이 첫 배정이 아니고 이전 루프의 배정과 같음) break;

8   for(j=1 to k) $\mathbf{z}_j$에 배정된 샘플의 평균으로 $\mathbf{z}_j$를 대치한다.

9 }

10

11 // 군집화 결과를 이용한 분할

12 for(j=1 to k) $\mathbf{z}_j$에 배정된 샘플(화소)을 j로 번호를 매긴다.

5.2.2 군집화를 이용한 영역 분할

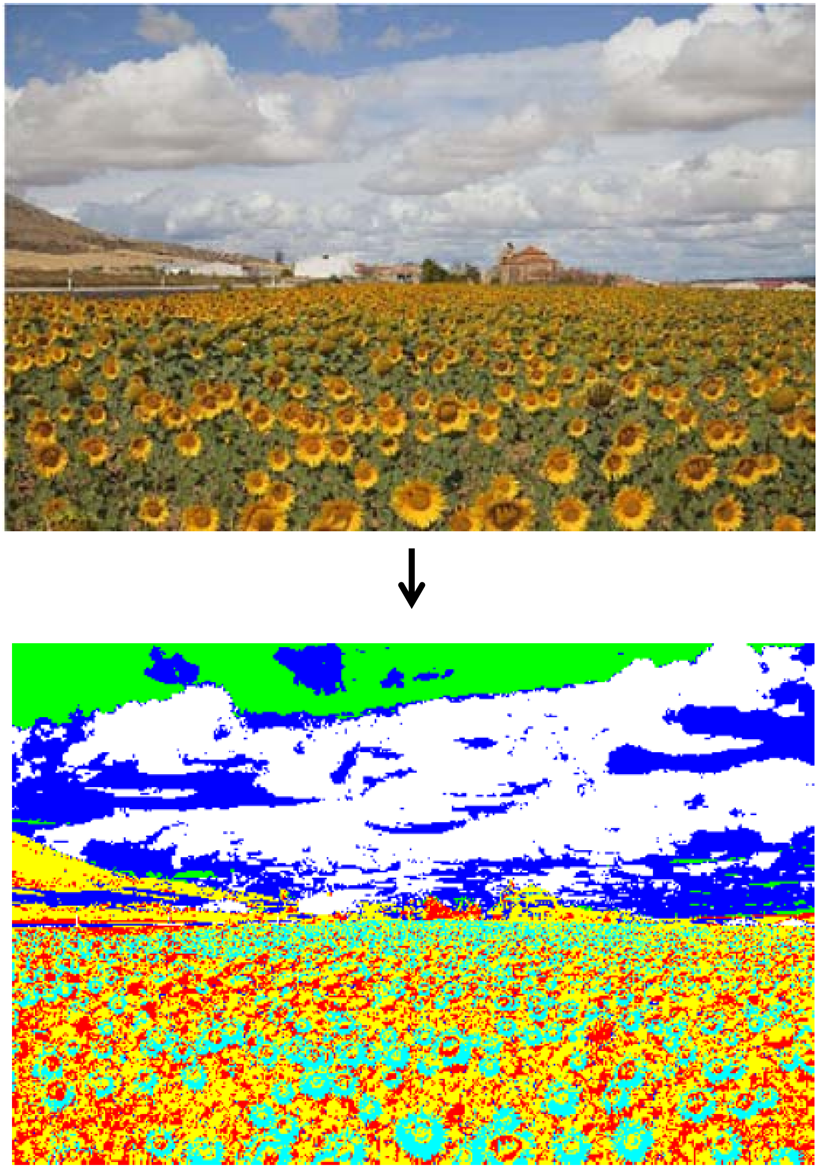


그림 5-6 *k*-means 군집화로 분할한 결과

자연 영상에서 낮은 성능
5.4절은 진보한 군집화 알고리즘인
민시프트 사용

5.2.3 분할합병

원리

- 영역의 균일성을 측정하는 $Q(r_i)$를 이용하여 분할과 합병을 반복
  - $Q(r_i)$이 거짓이면 r_i를 네 개 영역으로 등분하고 재귀 반복
  - $Q(r_i \cup r_j)$가 참이면 r_i와 r_j를 합병
- 분할 결과는 4진 트리로 표현

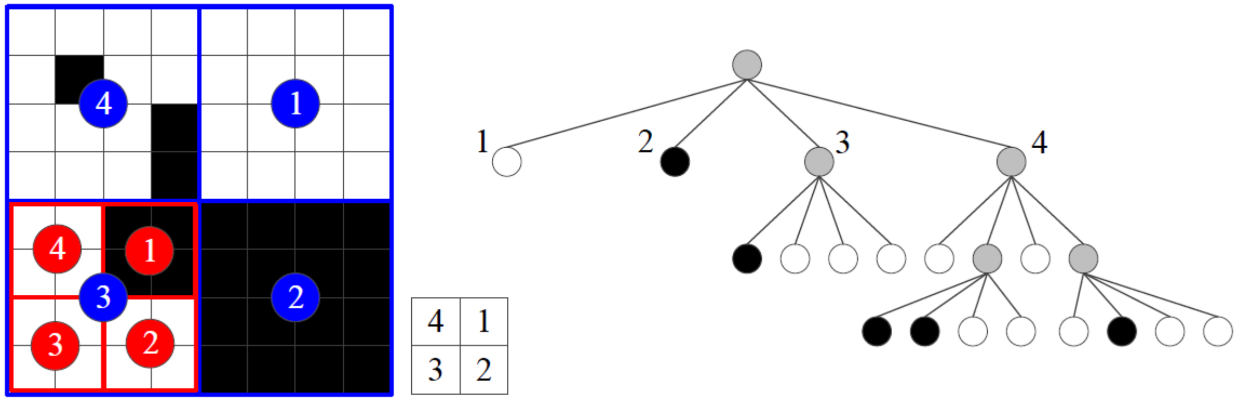


그림 5-7 4진 트리를 통한 영역 분할 결과

5.2.3 분할합병

알고리즘 5-3 분할합병을 적용한 영역 분할

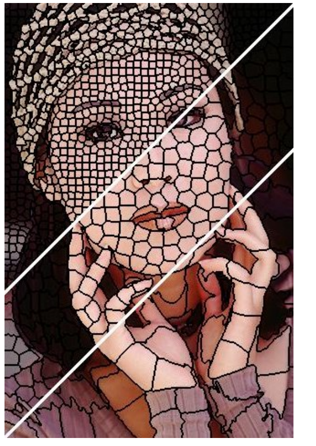
입력 : 영상 $f(j,i)$, $0 \le j \le M-1$, $0 \le i \le N-1$, 임계값 s

출력 : 분할 결과를 표현하는 4진 트리

```
1   split(f);  // 분할
2   while(TRUE) {  // 합병
3     for(모든 이웃한 영역 쌍 r_i와 r_j에 대해)
4       if(Q(r_i∪r_j)) r_i와 r_j를 합병하라.
5     3~4행에서 변화가 없으면 break;
6   }
7
8   function split(r){
9     if((not Q(r)) and (r의 넓이 > s)) {
10      r을 네 개의 4분 영상 r_i, 1≤i≤4로 나눈다.
11      for(i=1 to 4)
12        split(r_i);
13    }
14  }
```

5.3 그래프 방법

- 그래프
  - $G=(V,E)$
  - 노드 집합 $V=\{v_1, v_2, \ldots, v_n\}$
    - 화소 또는 슈퍼 화소가 노드
  - 에지 집합 E
    - 이웃 노드 간에 에지 설정
    - 두 노드 v_p와 v_q를 연결하는 에지는 가중치 w_{pq}를 가짐
    - 가중치는 유사도(같은 정도) 또는 거리(다른 정도)로 측정



superpixel

$$\text{거리 } d_{pq} = \begin{cases} |f(v_p) - f(v_q)|, & v_q \in Neigh(v_p) \\ \infty, & \text{그렇지 않으면} \end{cases}$$

$$\text{유사도 } s_{pq} = \begin{cases} e^{-d_{pq}}, & v_q \in Neigh(v_p) \\ 0, & \text{그렇지 않으면} \end{cases} \tag{5.5}$$

$$\text{이때, } \|\mathbf{x}(v_q) - \mathbf{x}(v_p)\| \le r \text{ 이면 } v_q \in Neigh(v_p)$$

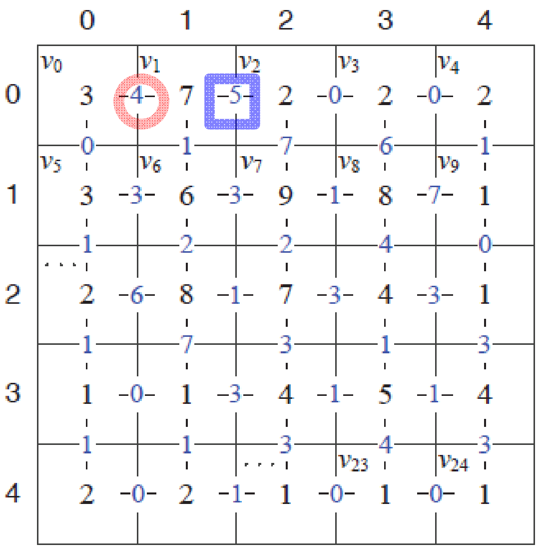
x: 화소의 위치

5.3 그래프 방법

예제 5-1 영상의 그래프 표현

[그림 5-8(a)]는 10단계의 명암을 갖는 5×5 크기의 아주 간단한 영상을 보여준다. 그래프로 표현하기 위해 먼저 25개의 화소를 행 우선 순서에 따라 v_0~v_{24}로 표기한다. 예를 들어 (1,2) 위치에 있는 화소는 노드 v_7이 된다.

식 (5.5)에서 이웃을 규정하는 거리를 r=1로 설정하여 4-연결된 화소만 이웃이라고 가정하고, 에지의 가중치는 식 (5.5)의 맨 위에 있는 거리를 채택한다. 그래프를 인접 행렬로 표현하면 25×25 행렬이 되는데, 에지의 가중치 값을 계산하여 채우면 [그림 5-8(b)]와 같다.



	v_0	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}	v_{11}	v_{12}	…	v_{23}	v_{24}
v_0	0	4	-	-	-	0	-	-	-	-	-	-	-	-	-	-
v_1	4	0	5	-	-	-	1	-	-	-	-	-	-	-	-	-
v_2	-	5	0	0	-	-	-	7	-	-	-	-	-	-	-	-
v_3	-	-	0	0	0	-	-	-	6	-	-	-	-	-	-	-
v_4	-	-	-	0	0	-	-	-	-	1	-	-	-	-	-	-
v_5	0	-	-	-	-	0	3	-	-	-	1	-	-	-	-	-
v_6	-	1	-	-	-	3	0	3	-	-	-	2	-	-	-	-
v_7	-	-	7	-	-	-	3	0	1	-	-	-	2	-	-	-
⋮		…						…							…	
v_{24}	-	-	-	-	-	-	-	-	-	-	-	-	-	-	0	0

(이미지 내의 25개 화소)

(a) 입력 영상(화소를 잇는 값(파란색)은 에지 가중치) (b) 인접 행렬 표현

그림 5-8 명암 영상의 그래프 표현

이 그래프는 대각선 방향으로 세 줄의 띠를 형성하는데, 유효한 값이 드문 희소행렬sparse matrix이다. 이웃의 범위를 규정하는 r을 크게 하면 띠의 폭이 늘어난다. 예를 들어, 8-이웃을 채택하면 1이었던 띠의 폭이 3이 된다.

5.3 그래프 방법

원리

에지 $w_{6,7}$=3, $w_{7,8}$=1. 따라서 $w_{6,7}$이 분할될 가능성이 높음

- 유사도가 높은 노드 쌍은 같은 영역(연결요소), 낮은 노드 쌍은 다른 영역에 배치
- 유사도 낮은 에지가 분할선이 될 가능성 높음
- '가능성이 높다'라는 표현의 중요성
  - 지역적으로 유사도 낮더라도 전역적 판단에서 자르지 말아야 한다면 분할선으로 취하지 않음 → 전역 최적해 추구

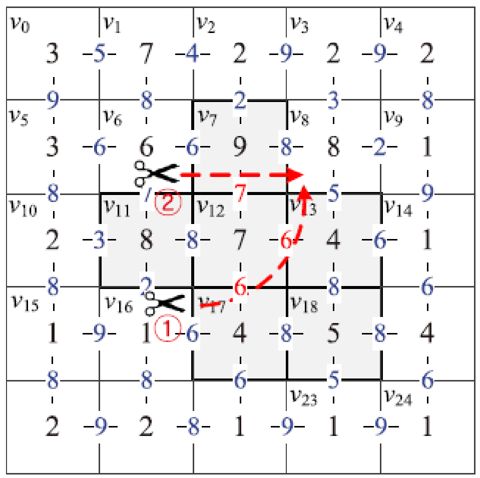
전역 최적화 문제의 구현

- 1. 어떤 분할의 좋은 정도를 측정하는 목적 함수 → 분할 품질 관련
- 2. 목적 함수를 최대화/최소화하는 최적해를 찾는 효율적인 탐색 알고리즘 → 속도 관련

- MST (최소 신장 트리)
- 정규화 절단

5.3.2 정규화 절단

- 유사도 그래프와 Wu의 분할 품질 척도
  - 예) 거리가 *d*일 때 9-*d*로 유사도 계산



```
     0123456789012345678901234

     95---9-------------------   0
     594---8------------------   1
     -499---2-----------------   2
     --999---3----------------   3
     ---99----8---------------   4
     9----96---8--------------   5
     -8---696---7-------------   6
     --2---698---7------------   7
     ---3---892---5-----------   8
     ----8---29----9----------   9
     -----8----93---8---------   0
     ------7---398---2--------   1
W=   -------7---896---6-------   2
     --------5---696---8------   3
     ---------9---69----6-----   4
     ----------8----99---8----   5
     -----------2---996---8---   6
     ------------6---698---6--   7
     -------------8---898---5-   8
     --------------6---89----6   9
     ---------------8----99---   0
     ----------------8---998--   1
     -----------------6---899-   2
     ------------------5---999   3
     -------------------6---99   4
```

그림 5-11 유사도 그래프와 인접 행렬

  - Wu의 척도 *cut* [Wu93]

$$cut(C_1, C_2) = \sum_{p \in C_1, q \in C_2} w_{pq} \qquad (5.11)$$

값이 작을 수록 유리한 분할(유사도가 작으므로)

5.3.2 정규화 절단

- Shi의 정규화 절단 [Shi2000]
  - *cut*의 문제점
    - 영역 크기에 대한 고려 없음

      예) 분할 1의 cut은 12, 분할 2는 7 → 하지만 직관적으로 분할 1이 우수
    - 원래 연결요소를 작은 것과 큰 것으로 분할하는 경향
    - 두 연결요소 간의 에지 개수가 적으므로 *cut*이 작아짐
  - Shi는 정규화 절단 *ncut*으로 확장하여 문제 해결

Normalized cut

$$ncut(C_1, C_2) = \frac{cut(C_1, C_2)}{assoc(C_1, C)} + \frac{cut(C_1, C_2)}{assoc(C_2, C)} \tag{5.12}$$

이때 $assoc(C_i, C) = \sum_{p \in C_i, q \in C} w_{pq}$, $C = C_1 \cup C_2$

assoc(C_i, C): 분할 요소 C_i와 분할전 요소 C사이의 가중치 값(영역 크기 고려)

  - *ncut*이 최소가 되는 분할을 찾는 문제는 NP-complete → 스펙트럴 군집화를 이용하여 근사해 구함

(Ref.) Luxburg'2007

(ex.) 앞의 경우

7/7 + 7/(8+6+6+7+...) = 1.xxx

12/큰값 + 12/큰값 = 0.xxxx

5.3.2 정규화 절단

근사해 탐색 알고리즘

$$(\mathbf{D} - \mathbf{W})\mathbf{y}^{\mathrm{T}} = \lambda \mathbf{D}\mathbf{y}^{\mathrm{T}} \qquad (5.13)$$

$$\mathbf{A}\mathbf{y}^{\mathrm{T}} = \lambda \mathbf{y}^{\mathrm{T}}$$
$$\text{이때} \quad \mathbf{A} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}} \qquad (5.14)$$

- **W**는 그림 5-11의 인접 행렬이고, **D**는 $d_{ii} = \sum_j w_{ij}$인 대각선 행렬

W = 25x25 행렬
D는 W로 구함

두번째 작은 고유값에 해당하는
고유 벡터가 최적 분할 정보 가짐

5.3.2 정규화 절단

영상 분할

- 식 (5.14)의 고유값과 고유 벡터를 이용하여 영상 분할
- 실제로 쓰이는 유사도
  - 화소의 특징값(명암 또는 컬러)뿐 아니라 화소 사이의 거리도 같이 사용
  - 특징값이 비슷할수록 거리가 짧을수록 큼
  - 특징값이 크게 다르더라도 이웃이면 같은 영역이 된다거나 특징값이 비슷하더라도 멀면 다른 영역에 배정할 가능성 확보 ← 전역 최적화

$$s_{pq} = \begin{cases} e^{-\left(\frac{\|\mathbf{f}(v_p)-\mathbf{f}(v_q)\|^2}{\sigma_I}+\frac{\|\mathbf{x}(v_p)-\mathbf{x}(v_q)\|^2}{\sigma_X}\right)}, & \|\mathbf{x}(v_p)-\mathbf{x}(v_q)\| < r \text{ 이면} \\ 0, & \text{그렇지 않으면} \end{cases} \tag{5.15}$$

이때 $\mathbf{f}(v) = f(v)$(명암 영상인 경우)

5.3.2 정규화 절단

■ 영상 분할 알고리즘

알고리즘 5-5 정규화 절단을 이용한 영상 분할

입력 : 영상 $f(j,i)$, $0 \le j \le M-1$, $0 \le i \le N-1$

출력 : 분할 결과 $S=\{C_1, C_2, \cdots, C_i\}$ // C_i는 연결요소

1	전체 노드 집합 V를 하나의 연결요소 C라고 한다.
2	C의 유사도 행렬 $\mathbf{W}$를 계산한다.
3	식 (5.14)를 풀어, 고유값과 고유 벡터를 구한다.
4	두 번째 작은 고유값에 해당하는 고유 벡터를 이용하여 C를 C_1과 C_2로 분할한다. (e.g.) Eigen vector의 부호에 따라 분할
5	C_1과 C_2 각각에 대해 추가 분할이 필요한지 판단하고, 그렇다면 그것을 C로 놓고 2~5행을 재귀적으로 반복한다.

5.3.2 정규화 절단

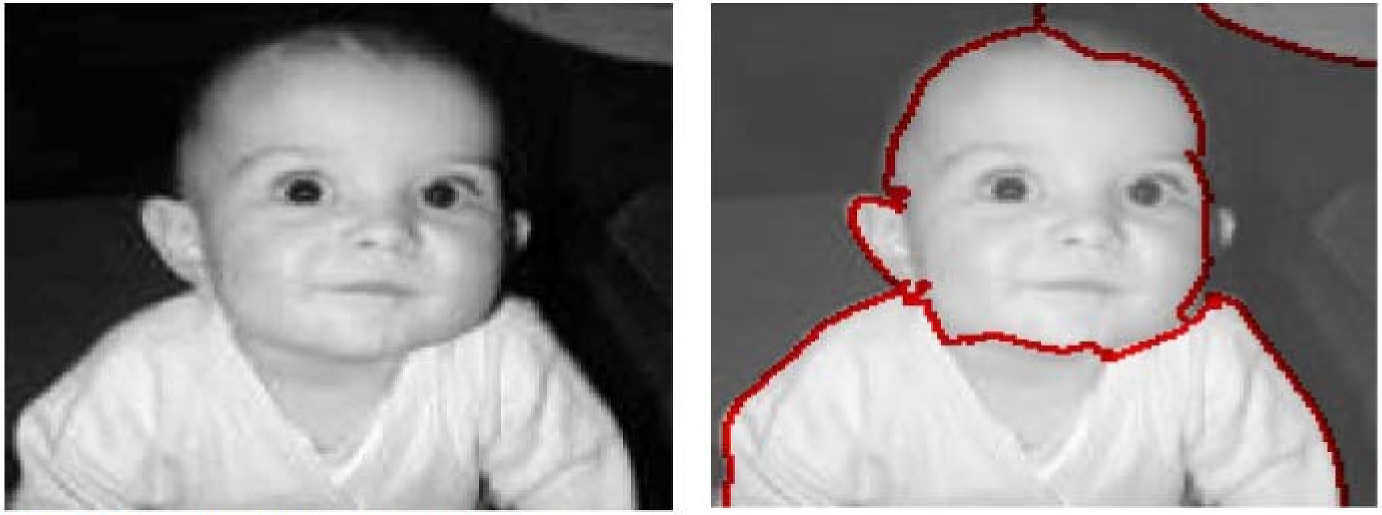


그림 5-12 정규화 절단을 적용한 영역 분할

논문 저자가 제공하는 Matlab 소스 코드를 http://www.cis.upenn.edu/~jshi/software/에서 받을 수 있다.

5.4 민시프트

모드 탐색

- 샘플이 주어진 경우, 어떤 점이 속한 모드(봉우리)를 찾는 문제
- y의 모드는?

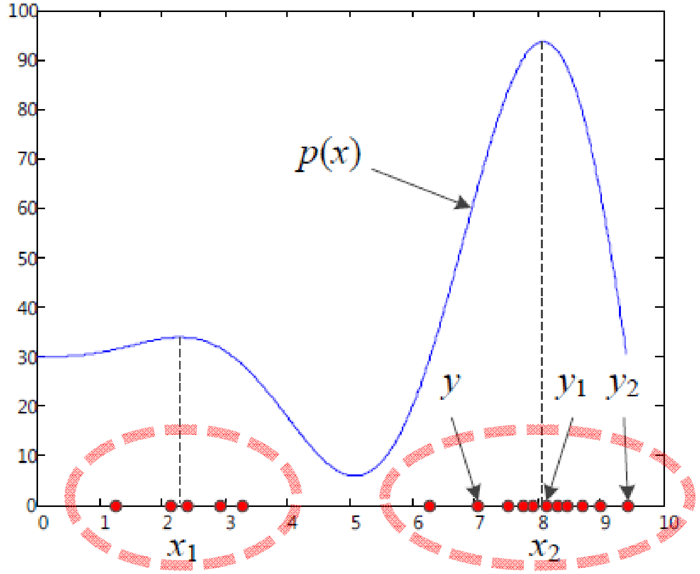


그림 5-13 모드 탐색

주어진 1차원 샘플을 2개의 주요 영역으로 분할 가능

→ 군집을 표현하는 확률을 추정

5.4.1 군집화

- 파젠 창을 이용한 확률분포 추정
  - h는 커널의 폭 (클수록 많은 샘플이 창 안으로 들어와 매끄러운 함수가 생성됨)

$$p(\mathbf{x}) = \frac{1}{nh^d}\sum_{i=1}^{n} k\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right) \quad (5.16)$$

위치 x근방에 들어오는 샘플들의 합으로 p(x) 추정

(Ref.) 패턴인식(오일석)

$$\text{평편한 커널}: k(\mathbf{x}) = \begin{cases} 1, & \|\mathbf{x}\| \leq 1 \\ 0, & \|\mathbf{x}\| > 1 \end{cases}$$

$$\text{가우시안 커널}: k(\mathbf{x}) = \begin{cases} e^{-\|\mathbf{x}\|^2}, & \|\mathbf{x}\| \leq 1 \\ 0, & \|\mathbf{x}\| > 1 \end{cases} \quad (5.17)$$

  - 차원의 저주 발생
    - 차원이 커질수록 기하급수적으로 메모리와 계산 시간 증가
  - 명시적으로 확률분포 추정하려는 시도는 수학적으로 그럴싸하지만 비현실적임

5.4.1 군집화

- 민시프트
  - 우회적으로 소속(모드)을 결정하는 기법
  - **y**를 $\mathbf{y}_0$로 놓고 시작하여, 수렴할 때까지 $\mathbf{y}_0 \rightarrow \mathbf{y}_1 \rightarrow \mathbf{y}_2 \rightarrow \ldots$ 반복

$$\mathbf{y}_{t+1} = \frac{\sum_{i=1}^{n} \mathbf{x}_i k\left(\frac{\mathbf{x}_i - \mathbf{y}_t}{h}\right)}{\sum_{i=1}^{n} k\left(\frac{\mathbf{x}_i - \mathbf{y}_t}{h}\right)} \qquad (5.18)$$

샘플들의 (정규화된) 가중 평균 값
가중치 K(.)는 X_i와 중앙 값 y_t까지의 거리로 결정

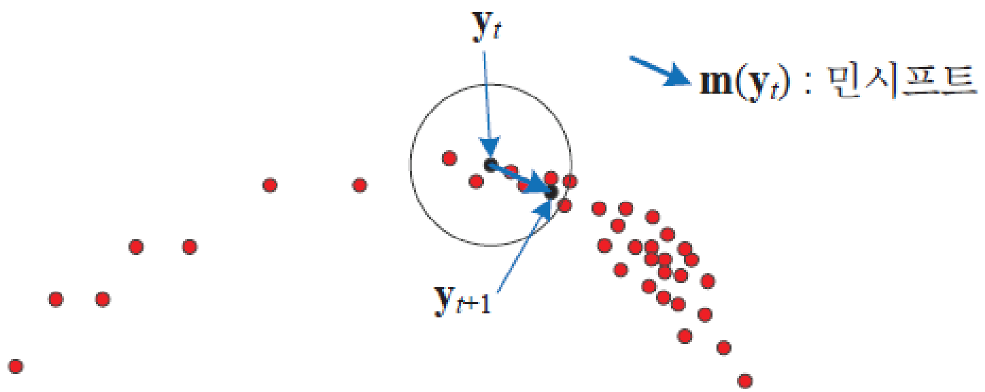


그림 5-14 현재 점 $\mathbf{y}_t$를 민시프트 $\mathbf{m}(\mathbf{y}_t)$가 $\mathbf{y}_{t+1}$로 이동시킴

y_t를 중심으로 weighted average하면
새로운 중심 y_{t+1}이 나옴
(원 내에 속하는 화소들이 한쪽으로 쏠려 있으므로)

5.4.1 군집화

- 바꾸어 쓰면,

$$\mathbf{y}_{t+1} = \mathbf{y}_t + \mathbf{m}(\mathbf{y}_t)$$

$$\text{이때 } \mathbf{m}(\mathbf{y}_t) = \mathbf{y}_{t+1} - \mathbf{y}_t = \frac{\sum_{i=1}^{n} \mathbf{x}_i k\left(\frac{\mathbf{x}_i - \mathbf{y}_t}{h}\right)}{\sum_{i=1}^{n} k\left(\frac{\mathbf{x}_i - \mathbf{y}_t}{h}\right)} - \mathbf{y}_t = \frac{\sum_{i=1}^{n} (\mathbf{x}_i - \mathbf{y}_t) k\left(\frac{\mathbf{x}_i - \mathbf{y}_t}{h}\right)}{\sum_{i=1}^{n} k\left(\frac{\mathbf{x}_i - \mathbf{y}_t}{h}\right)} \tag{5.19}$$

- 민시프트를 이용한 군집화 알고리즘
  - 모든 점에 대해 식 (5.19)로 수렴점 찾은 후, 군집 배정
  - 뛰어난 군집화 알고리즘
  - 군집 개수 자동 결정, 임의 모양 군집, 설정할 매개변수 적음

5.4.1 군집화

알고리즘 5-6 민시프트를 이용한 군집화

입력 : 샘플 집합 $X=\{\mathbf{x}_i \mid i=1,2,\cdots,n\}$, ε(수렴 임계값)

출력 : k개의 모드 $\mathbf{z}_i$, $1 \le i \le k$, $\mathbf{x}_i$의 소속을 나타내는 $\alpha(\mathbf{x}_i)$, $1 \le i \le n$

for(i=1 to n) {
  $\mathbf{y}_0=\mathbf{x}_i$; // 초기점 설정
  t=0;
  while(TRUE) {
    식 (5.19)를 이용하여 $\mathbf{y}_{t+1}$을 계산한다.
    if($\|\mathbf{y}_{t+1}-\mathbf{y}_t\| \le \varepsilon$) break; // 수렴
    t++;
  }
  $\mathbf{v}_i=\mathbf{y}_{t+1}$; // $\mathbf{x}_i$의 수렴점을 저장
}

// 군집화 단계
$\mathbf{v}_i$, $i=1,2,\cdots,n$에서 h 이내에 있는 점들을 모아 군집화하고, 군집 중심을 $\mathbf{z}_j$, $j=1,2,\cdots,k$라 한다.
$\mathbf{x}_i$, $i=1,2,\cdots,n$이 속한 군집 $\mathbf{z}_c$를 찾아 $\alpha(\mathbf{x}_i)=c$라 한다.

5.4.2 영상 분할과 스무딩

- 민시프트를 영상 분할에 어떻게 적용할까? [Comaniciu2002]
  - 영상 f를 샘플 집합 X로 변형
    - 화소가 샘플이 됨

- 몇 차원으로 표현할 것인가?
  - 컬러만 표현하면 2.2절의 k-means와 비슷한 결과 (그림 5-6)
  - 컬러 $\mathbf{x}^r=(r,g,b)$와 화소의 위치 $\mathbf{x}^s=(y,x)$를 같이 표현 → 5차원 벡터 $\mathbf{x}=(r,g,b,y,x)$
  - $\mathbf{x}^r$과 $\mathbf{x}^s$의 스케일 차이를 고려하여 커널을 별도로 표현

$$k(\mathbf{x}) = k\left(\frac{\mathbf{x}^s}{h_s}\right)k\left(\frac{\mathbf{x}^r}{h_r}\right) \tag{5.20}$$

Comaniciu는 256x256크기의 이미지에 대해
$h^s=8$ (17x17), $h^r=7$(15x15x15)로 사용
$X^r=(L, u, v)$로 사용

5.4.2 영상 분할과 스무딩

알고리즘 5-7 민시프트를 이용한 영상 분할

입력 : 컬러 영상 $f_r(j,i)$, $f_g(j,i)$, $f_b(j,i)$, $0 \le j \le M-1$, $0 \le i \le N-1$, min_area(이것보다 작은 영역 제거)

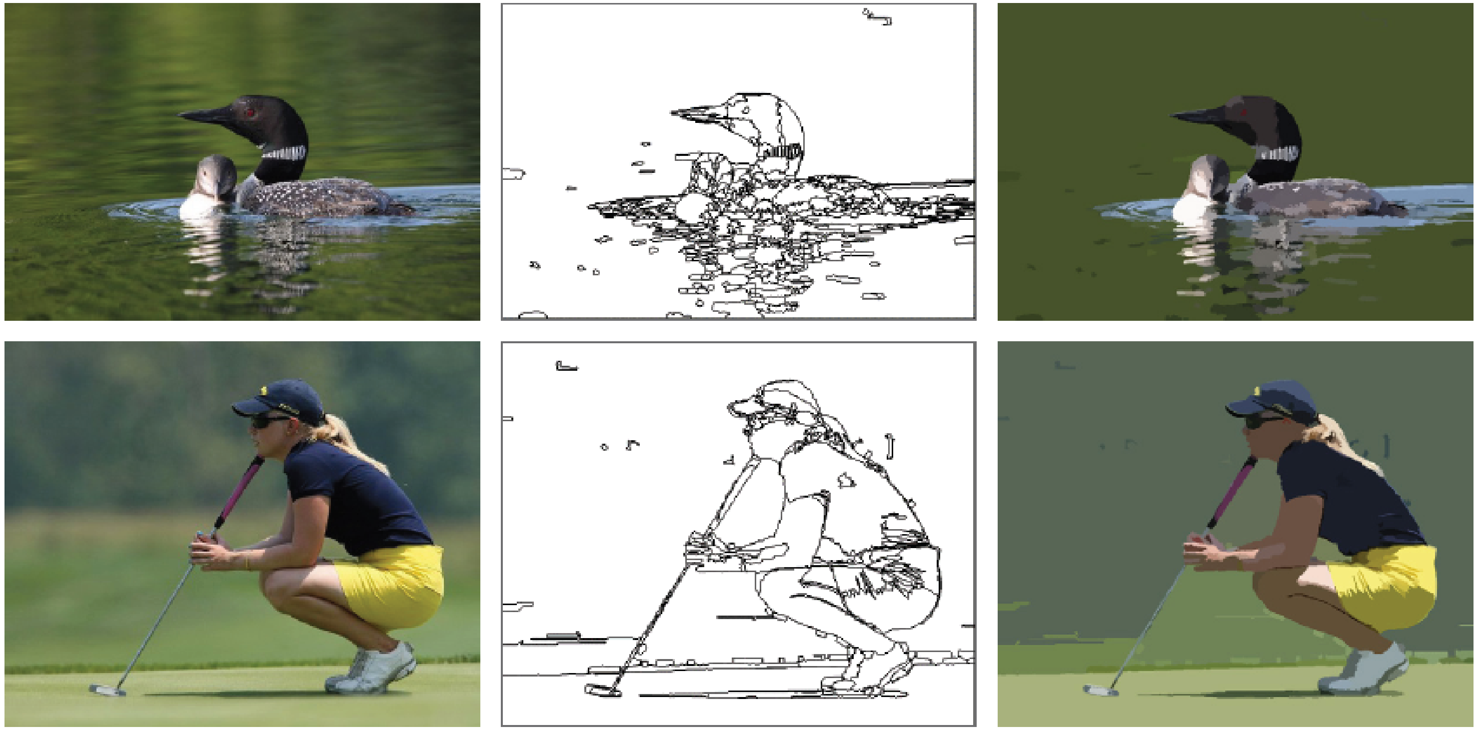
출력 : 분할 결과 $S=\{C_1, C_2, \cdots, C_k\}$ // C_i는 연결요소

```
// 전처리
컬러 공간을 RGB에서 Luv로 변환한다.
모든 화소를 5차원 공간 x_i=(y_i, x_i, L_i, u_i, v_i), i=1,2,…,MN으로 매핑한다.

// 모드 탐색
for(i=1 to MN) {
  y_0=x_i;  // 초기점 설정
  t=0;
  while(TRUE) {
    식 (5.19)를 이용하여 y_{t+1}을 계산한다.
    if(‖y_{t+1} - y_t‖ ≤ ε) break;  // 수렴
    t++;
  }
  v_i=y_{t+1};  // x_i의 수렴점 저장
}

// 후처리(수렴점을 모아 군집(영역) 구성, 작은 영역 제거)
v_i, i=1,2,…,MN에서 공간 좌표는 h_s, 컬러 좌표는 h_r 거리 이내에 있는 점들을 모아 군집화하고, 군집 중심을 z_j,
j=1,2,…,k라 한다.
x_i, i=1,2,…,MN가 속한 군집 z_c를 찾아, x_i를 연결요소 C_c에 배정한다.
크기가 min_area보다 작은 연결요소를 제거한다.
```

5.4.2 영상 분할과 스무딩



(a) 원래 영상 (b) 영역의 경계를 표시 (c) 영역의 평균 컬러로 표시된 분할 영상

그림 5-15 민시프트(EDISON)로 분할한 영상