

## 영상처리 시스템을 이용한 그림조각 맞추기에 관한 연구

## (A Vision-Based Jig-Saw Puzzle Matching Method)

李 東 柱\*, 徐 一 弘\*, 吳 尚 錄\*\*

(Dong Joo Lee, Il Hong Suh, and Sang Rok Oh)

## 要 約

본 논문에서는 흩어진 그림조각을 맞추기 위하여 조합이 완료된 그림조각에 대한 사전정보를 이용하지 않고 기존의 윤곽선 조합 알고리즘(boundary matching algorithm)을 이용하여 내부에서 외부로 그림조각을 맞추어 나가는 알고리즘을 제안하였다. 구체적으로 그림 조각들에 대한 묘사를 위하여 저해상도의 영상처리 시스템에서 얻은 데이터를 윤곽선 추적 알고리즘(boundary tracking algorithm)을 이용하여 각 그림조각들의 기하학적인 모양을 추출하고, 추출된 각 그림조각들을 특징지워주는 구석점(corner point)사이의 각도, 거리 및 구석점의 볼록성(convexity)과 오목성(concavity)등의 특징들은 그림조각의 윤곽선에서 꼭짓의 불연속성을 이용하여 구한 구석점들로부터 계산하였다. 제안된 알고리즘을 여러종류의 그림조각에 적용한 결과 만족할 만한 결과를 얻었다.

## Abstract

In this paper, a novel method of jig-saw puzzle matching is proposed using a modified boundary matching algorithm without a priori knowledge for the matched puzzle. Specifically, a boundary tracking algorithm is utilised to segment each puzzle from low-resolution image data. Segmented puzzle is described via corner point, angle and distance between two adjacent coner point, and convexity and/or concavity of corner point. Proposed algorithm is implemented and tested in IBM PC and PC version vision system, and applied successfully to real jig-saw puzzles.

## I. 서 론

단순 반복 작업에 주로 응용 되어온 산업용 로봇은 최근 컴퓨터 기술의 발달에 힘입어 기계부품의 조립과 같은 복잡하고 다양한 일을 포함하여 점차 지능을 요구하는 작업에까지 응용이 확대되고 있다.

이러한 로봇 응용분야의 확장은 로봇에게 다양한 센서 기능을 부여함으로써 가능하게 되었는데, 그 대표적인 것이 카메라를 이용한 시각 센서 시스템을 로봇 시스템과 결합하여 물체 인식을 통한 조립작업 등을 수행하게 하는 것이다.<sup>(1-4)</sup>

본 논문에서는 이러한 연구의 일환으로 시각시스템을 로봇 시스템과 결합하고 또한 로봇에게 초보적인 지능을 부여하기 위하여 시각시스템으로 부터 들어온 그림조각에 대한 영상정보를 이용하여 그림조각을 맞추는 작업을 예로들어 연구하였다. 사람이 그림조각 맞추기에 사용하는 정보 및 그 조합은 상당히 복잡하며 직관적이라고 할 수 있는데 그 모

\*正會員, 漢陽大學校 電子工學科  
(Dept. of Elec. Eng., Hanyang Univ.)

\*\*正會員, 韓國科學技術完 電氣制御 研究室  
(Power Controls Lab., KAIST)

接受日字: 1989年 1月 14日

든 메카니즘을 컴퓨터 프로그래밍하기란 거의 불가능하다. 따라서 컴퓨터를 이용한 2차원 그림조각 맞추기에 대해 제시된 여러방법에서는<sup>14-17</sup> 각 그림조각들의 윤곽선으로부터 특정 정보만을 추출하여 그 정보를 이용하여 그림조각 맞추기를 수행하는 윤곽선 조합(boundary matching) 방법을 이용하고 있다. 이들 윤곽선 조합 방법에서 사용하는 대표적인 세가지 특징 정보는 첫째, 두 곡선이 맞추어질때 대응하는 점인 조합점, 둘째, 두 곡선이 맞추어지기 위하여 한 곡선이 다른 곡선에 대하여 회전하는 방향인 조합각 및 셋째, 두 곡선이 맞는 범위인 조합범위등으로 구분할 수 있다.

Freeman<sup>15)</sup>은 물체의 윤곽선을 체인코드(chain code)로 표시하여 조합범위(match range)를 먼저 결정하고 조합점(match point)과 조합각(match angle)을 구하였는바, 그림조각이 어떻게 맞추어지는가에 무관하게 그림조각의 윤곽선들을 분리시킨다는 불합리한 점을 갖고 있다.

한편 [4, 6]에서는 그림조각의 윤곽선의 곡률이 최대 혹은 최소인 점을 중심으로 극좌표를 구성하여 우선 조합점을 구하고 조합각 및 조합범위등을 구하는 방법을 제시하고 있고[4, 7]에서는 알고리즘의 효율성을 위하여 맞추어진 전체의 모양에 제한조건을 두고 있다.

본 논문에서는 체인코드로 표시한 윤곽선에서 간단한 특징들을 구하고 조합점, 조합각 그리고 조합범위등을 차례로 구하여 조합을 수행하는 윤곽선 조합 방법과 몇가지 직관적인 방법을 도입하여 CCD 카메라로부터 실제 영상을 받아들여서 그림조각을 맞추는 알고리즘을 제시하기로 한다. 특히 사전정보(Pre-information)없이 그림조각들을 조합하기 위하여 한 조각에 다른 조각이 맞는다고 인식될때까지 계속 맞추어 나가는 방법을 사용하였다. 이 방법은 실제로 오차가 많은 경우에는 사전정보를 알고 수행하는 경우보다는 덜 유용하지만 그림조각의 모양에 가해지는 제약이 적어진다는 점에서 더 효과적이라고 할 수 있다.

실험대상이 되는 그림조각들은 Back Tracking 없이 맞추어질 수 있는 유일성(Uniqueness)을 갖도록 제한하였다. 이 유일성은 그림조각들이 한 가지 방법만으로 알맞게 맞추어진다는 것을 의미하는데, 그림조각의 기하학적 특성 및 조명과 카메라 및 비전 시스템의 한정된 해상도(finiteresolution)에 의한 잡음에 관련된다. 물론 그림조각 들이 유일성을 지니고 있다고 하여도 Back Tracking을 요구하는 경우가 있는데 이것은 그림조각을 맞추는 기준이 부정확

하거나 또는 오차가 커서 생기는 것으로 판단된다.

본 논문에서는 우선 2장에서 입력된 영상정보에 대한 전 처리 과정을 설명하고 3장에서 전처리된 그림조각들을 짜맞추는 알고리즘을 제시하기로 한다.

끝으로 4장에서는 제시된 알고리즘의 타당성을 보이기 위하여 실험한 결과를 보이고 5장은 결론의 장으로 한다.

## II. 전 처리 과정(pre-processing)

본 장은 서론에서 언급한바와 같이 컴퓨터에 의한 2차원 그림조각 맞추기를 수행하기 위한 전 단계로서, 알고리즘의 수행을 위한 그림조각의 표현 방법 및 특징을 구하는 방법을 기술하기로 한다.

### 1. 그림조각의 표현 방법

그림조각을 맞추는 알고리즘의 수행을 위하여는 먼저 각각의 그림조각에 대한 특징을 구하여야 하는데, 본 논문에서는 기본적으로 윤곽선 조합 방법을 이용하기 때문에 각 그림조각들을 윤곽선만으로 표시하여야 한다. 입력영상은 여러개의 그림조각을 포함하고 있으므로 주어진 영상상에서 각 그림조각을 추출하고, 추출된 그림조각의 경계점을 추적하여 윤곽선을 알아내는 알고리즘이 필요하다.

주어진 이진영상에서 각 그림조각을 추출하기 위하여 Raster Scan 방식<sup>18)</sup>을 이용하였고, 그림조각의 윤곽선 추적(boundary tracking)을 위하여는 Freeman<sup>19)</sup>이 제시한 시계방향의 8-방향 체인코드(clockwise 8-direction chain code scheme)을 사용하였다.

### 2. 구석점(corner point)의 계산 및 특징 추출

그림조각의 놓여진 방향과 크기에 무관한 특징들을 구하기 위하여 그림조각의 윤곽선을 나타내는 체인코드로부터 곡률반경의 불연속성을 계산하여 구한 구석점을 이용하였다.<sup>10)</sup> 이러한 구석점들은 체인을 구성하는 s개의 선소(line segment)를 이동시키면서 찾는다. 이를 구체적으로 기술하면 다음과 같다.

체인코드의 방향벡터를  $a_i$ 로 표시하면, 주어진 n-link chain  $A^n$ 을  $A^n = C_{i=1}^n a_i = a_1 a_2 a_3 \dots a_n$ 으로 나타낼 수 있고 또한, j점을 끝점으로 하며 s개의 링크(link)를 갖는 선소  $L_j^s$ 는 다음과 같이 표시된다.

$$L_j^s = C_{i=j-s+1}^j a_i \quad j = 1, 2, 3, \dots, n, n > s \quad (1)$$

또한  $L_j^s$ 의 X와 Y성분  $X_j^s$ 와  $Y_j^s$ 는 다음과 같다.

$$X_j^s = \sum_{i=j-s+1}^j a_{ix}, \quad Y_j^s = \sum_{i=j-s+1}^j a_{iy} \quad (2)$$

여기서  $a_{ix}$ 와  $a_{iy}$ 는 각각 체인링크(chain link)  $a_i$ 의

X와 Y성분이며  $a_{ix}, a_{iy} \in \{-1, 0, 1\}$ 이다.

한편, X축에 대한  $L_j^s$ 의 각도  $\theta_j^s$ 는

$$\theta_j^s = \begin{cases} \tan^{-1} Y_j^s/X_j^s & \text{if } |X_j^s| \geq |Y_j^s| \\ \cot^{-1} X_j^s/Y_j^s & \text{if } |X_j^s| < |Y_j^s| \end{cases} \quad (3)$$

로 주어진다.

이  $\theta_j^s$ 로부터  $j$ 가 1부터  $n$ 까지 변할때의 체인의 곡률반경을 구하는데, 잡음의 영향을 감소시키기 위하여 Incremental Curvature를 다음과 같이 정의하자.

$$\delta_j^s = 2 \cdot \frac{(\theta_{j+1}^s - \theta_j^s) + (\theta_j^s - \theta_{j-1}^s)}{2} = \theta_{j+1}^s - \theta_{j-1}^s \quad (4)$$

적당한 한계값 이내의  $\delta_j^s$  값을 갖는 연속적인 체인노드(chain node)들을 discontinuity free 영역이라고 하며 다음과 같이 표시하는데, 여기서  $t1$ 과  $t2$ 는 어떤점  $j$ 의 좌측과 우측에 대한 각각의 discontinuity free 영역의 길이이다.

$$t1 = \max \{t \mid \delta_{j-v}^s \in (-\Delta, \Delta), \forall 1 < v < t\} \quad (5)$$

$$t2 = \max \{t \mid \delta_{j+v}^s \in (-\Delta, \Delta), \forall 1 < v < t\}$$

$\Delta$  = Discontinuity-Free 영역을 구하기 위한  $\delta_j^s$ 의 한계값

따라서 어떠한 점에서의 구석(corner)의 정도는 아래의 두값에 의해서 결정된다.

- incremental curvature  $\delta_j^s$ 의 불연속성
- 양변의 discontinuity free 영역의 길이

즉 어떤점  $j$ 의 구석의 정도를 나타내는 값을 그 점의 구석량(cornerity)  $K_j$ 라고하면  $K_j$ 는 다음과 같이 표시된다.

$$K_j = \sqrt{t1} \cdot \sqrt{t2} \cdot \sum_{i=1}^{j-1} \delta_i^s \quad (6)$$

따라서 물체의 구석점은 각 점의 구석량  $K_j$ 와 그 점 주위의 적당한 범위내의 다른 구석량 값을 비교하여 최대값을 갖는 점을 그 물체의 구석점 이라고 인식하였다.

위에서 구한 구석점을 이용하여 그림조각 맞추기 알고리즘에 사용하는 특징으로서 다음의 세 가지를 사용하였다.

- 1) 구석점의 볼록성(convexity) 또는 오목성(concavity)의 판별

한 구석점의 좌표값을 그 점에 좌로 인접한 두 점을 시계방향으로 연결하는 직선의 방정식에 대입한 후 부호로써 결정한다.  $i$ 번째 그림조각의  $j$ 번째 구석

점의 볼록성 혹은 오목성을  $CONi(j)$ 로 표시하면 다음과 같다.

$$CONi(j) = \begin{cases} 1: \text{볼록 구석점 (convex corner point)} \\ 0: \text{오목 구석점 (concave corner point)} \end{cases} \quad (7)$$

- 2) 구석각(corner angle)

하나의 구석점이 좌우로 이웃한 두 구석점과 이루는 각으로서  $i$ 번째 그림조각의  $j$ 번째 구석점의 구석각은 다음과 같다.

$$ANGi(j) = \begin{cases} \theta & : \text{볼록구석점} \\ 360^\circ - \theta & : \text{오목 구석점} \end{cases} \quad (8)$$

$\theta$ ; 2개의 직선이 만나서 이루는 사이각,  $0^\circ \leq \theta \leq 180^\circ$

- 3) 구석점의 거리(corner distance)

하나의 구석점에서 시계방향으로 이웃한 다음의 구석점까지의 거리를 말하며  $i$ 번째 그림조각의  $j$ 번째 구석점의 거리를  $DSTi(j)$ 로 표현하기로 한다.

### III. Jig-Saw Puzzle의 조합

본 장에서는 전 장에서 구한 특징들을 이용한 유효한 조합 방법에 의하여 그림조각을 조합하는 알고리즘을 설명하기로 한다.

서론에서 언급한 바와같이 모든 경우에 대하여 알고리즘을 구현하기란 거의 불가능하므로 Freeman<sup>[5]</sup>과 Rodack<sup>[6]</sup>은 사용하는 특징을 구하는 순서를 다르게 하여 알고리즘을 제시하였고 고순섭<sup>[7]</sup>은 그림1에서와 같이 맞춰진 조각의 구석점을 두 개 이상의 조각이 공유하는 경우(그림1.a)와 맞춰진 퍼즐의 한 변을 한 조각이 두 번 이상 공유하는 경우(그림1.b)로 나누어 그림조각의 모양에 제한을 두고 알고리즘을 제시하였다.

본 논문에서는 그림 1과 같은 모양을 갖는 조각에 대해서도 그림조각 맞추기를 수행하기 위하여 조합이 완료된 모양의 그림조각에 대한 정보를 이용하지

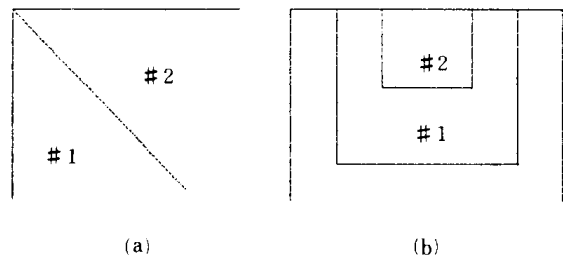


그림 1. 그림모양에 가해지는 제한  
Fig. 1. The constraints on the puzzle.

않고, 안쪽에서 시작하여 바깥쪽으로 그림조각의 조합을 수행하는 알고리즘을 제안하기로 한다. 그러나 본 알고리즘 역시 완전한 것이 아니므로 서로 다른 두 개의 그림조각에 속한 한 쌍의 구석점의 구석각의 합이  $360^\circ$  또는  $180^\circ$ 인 구석점들이 있으면 좋은 결과를 얻지만, 이러한 구석점들이 없으면 알고리즘의 수행 결과가 좋지 않다. 따라서 아래의 설명은 위의 조건을 만족하는 그림조각에 대한 것으로 제한한다. 우선 본 논문에서 사용하는 몇 가지 용어에 대한 설명을 하면 다음과 같다.

- 주조각(master piece) : 두 개의 조각이 조합될 때에 기준이 되는 조각; #m
- 종속조각(slave piece) : 두 개의 조각이 조합될 때에 주조각에 붙여지는 조각; #s
- $P_m(i)$  : m 번째 조각에 속한 i 번째 구석점
- $NUM(m)$  : m 번째 조각에 속한 모든 구석점의 수
- $\epsilon_1$  : 각도에 대한 허용 한계 오차
- $\epsilon_2$  : 거리에 대한 허용 한계 오차

1. 후보조합조각(candidate match piece)의 선택 조합을 수행하기 위해서는 우선 기준이 되는 조각을 정하여야 한다. 기준이 되는 조각의 선택은 조각의 추출과정에서 찾아진 순서대로 조각조각에 지정된 번호에 따라 조합조건을 적용하여 두 개의 조각을 선택한다. 알고리즘의 첫 부분에서는 여러 조각중에서 2개를 후보조합 조각으로 선택하고 뒤의 3.5절에서 설명하고 있는 부분조합조각을 구성한 다음 이 조각을 주조각으로 정하고 남은 조각중에서 하나를 선택하여 종속조각으로 하였다.

2. 조합조건(match condition) 및 조합범위의 설정 조합조건은 서로 다른 두 조각에서 구한 후보조합점(candidate match point)의 조합 여부를 결정하는 조건을 뜻한다.

본 논문에서는 전술한 바 두가지 경우로 나누어 각 경우에 대하여 조합조건 및 조합범위의 설정방법을 제시하였다.

1) 서로 다른 두 조각에 속한 두개의 구석점이 내부점을 구성하는 경우

$$\text{즉 } ANG_m(i) + ANG_s(j) = 360^\circ, m \neq s$$

인 경우 각 조각의 구석점에 시계방향으로 붙여진 번호를 이용하여 조합조건을 정한다. i와 j가 두 조각의 후보조합점이라면 조합조건은 다음과 같다.

- (a)  $CON_m(i) + CON_s(j) \neq 0$
- (b)  $|ANG_m(i) + ANG(j) - 360^\circ| \leq \epsilon_1$
- (c)  $|DST_m(i-1) - DST_s(j)| \leq \epsilon_2$  and  $|DST_m(i) - DST_s(j-1)| \leq \epsilon_2$

$$(d) CON_m(i-1) + CON_s(j+1) \neq 0 \text{ and } CON_m(i+1) + CON_s(j-1) \neq 0$$

위의 조건 중에서 두 개의 볼록점은  $360^\circ$ 를 구성하지 못하므로 우선 a)조건을 검사한다.

이 경우 조합범위의 설정은 다음과 같다.

조합조건을 검사하여 두 구석점  $P_m(i)$ 와  $P_s(j)$  ( $m \neq s$ )가 조합점으로 판정되면 이들과 이웃한 점들 역시 조합조건을 만족하는 경우가 있으므로 두 조각이 만나는 범위를 정하여야 한다. 이를 그림2에 알고리즘 형식으로 설명하였다. 그림2의 알고리즘에서 range\_1과 range\_2는 조각 #m의 조합범위이고 range\_3와 range\_4는 조각 #s의 조합범위이다.

```

1° Set match point ; Pm(i) Ps(j)
2° direction=1, counterdirection= -1
   ii=i, jj=j
3° ii=ii+counterdirection
   jj=jj+direction
4° Check match condition ; Pm(ii) Ps(jj)
   IF pass GO TO 3°
   ELSE range-1=ii, range-2=jj
        ii=i, jj=j
5° ii=ii+direction jj=jj+counterdirection
6° Check match condition ; Pm(ii) Ps(jj)
   IF pass GO TO 5°
   ELSE range-3=ii, range-4=jj
7° END

```

그림 2. 조합범위의 설정을 위한 알고리즘  
Fig. 2. Match range setting algorithm.

2) 서로 다른 두 조각에 속한 두 개의 구석점이 변상의 점(Side point)을 구성하는 경우 즉

$$ANG_m(i) + ANG_s(j) = 180^\circ, m \neq s$$

인 경우로써 1)의 경우의 조합조건을 만족하는 구석점이 없을 때에 적용한다. 주조각 #m의 구석점  $P_m(i)$ 와 종속조각의 구석점  $P_m(j)$ 가 후보조합점인 조합조건은 다음과 같다(그림3.a)

- (a)  $CON_m(i) + CON_s(j) \neq 0$
- (b)  $|ANG_m(i) + ANG_s(j) - 180^\circ| \leq \epsilon_1$
- (c)  $|DST_m(i) - DST_s(j-1)| \leq \epsilon_2$

위의 조건이 그림 3. b에서는 성립하지 않는데 조합점을 중심으로 한쪽 방향만 비교하므로 (a), (b) 조

건을 만족하는 점의 위치에 영향을 받기 때문이다. 이러한 경우에는 단지 주조각과 종속조각을 맞바꾼 다음에 위의 조건을 적용한다(그림3. c). 이 경우 조합범위의 설정은 위의 조합조건을 만족하면서 서로 맞닿는 한 변이 조합범위가 된다.

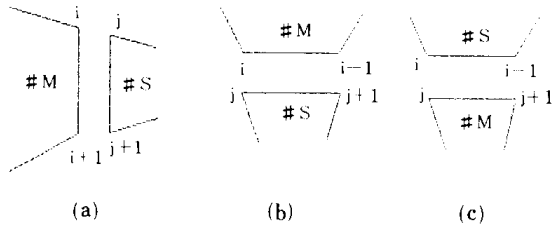


그림 3. 경우 2)에 대한 설명  
Fig. 3. Match condition for case 2).

3. 합병(merging)

두 개의 조각 #m과 #s가 조합된다고 판정되면 주조각에 종속조각을 일치시키기 위하여 이동벡터와 회전각을 구하여야 한다. i와 j, k와 l을 각각 조각 #m과 #s의 조합범위라고 하자.

1) 이동벡터

두 개의 조각 #m과 #s의 서로 맞닿는 구석점 Pm(i)와 Ps(k) 사이의 거리를 계산하여 구한다.

2) 회전각

각 조각의 조합범위를 나타내는 두 구석점을 연결하는 벡터가 X축과 이루는 각을 구한 후 이들 두 각의 차이를 회전각으로 정한다.

4. Fit-or-Not의 결정

전 과정까지 수행하여 두 조각을 일치시킨 다음에 이들 두 조각이 조합범위 밖에서 서로 겹치는지의 여부를 검사하여야 한다. 다음의 알고리즘을 두 조각의 조합범위 밖에 있는 모든 변에 대하여 적용하였다.

- 1° 종속조각의 조합범위 밖에 있는 한 변을 선택한다.
- 2° 주조각의 조합범위 밖에 있는 한 변에 대한 직선의 방정식 F(x, y)를 구한다.
- 3° 1°에서 선택한 변의 양끝점(Pi, Pe)을 2°에서 구한 방정식에 대입한다.
- 4° 만일 F(Pi)·F(Pe) > 0이면 1°로 가고 아니면 두개의 조각 #m과 #s의 조합범위를 다시 찾는다.

5. 부분조합조각(partial matched piece)의 구성

두 조각이 서로 조합된다고 판정하여 합병과정을 수행한 다음에는 남아있는 조각에 대하여 지금까지의 과정을 반복하여야 한다. 이 경우에 기준이 되는 새로운 주조각을 만들어 주어야 하는데 합병과정에서 새로이 만들어진 조각을 그대로 사용하기 보다는 다음과 같은 보정작업을 수행한 다음에 사용한다.

1) 사라지는 점(hidden point)의 처리

두 조각이 맞추어지면 서로 맞닿는 부분의 구석점들은 사라지는 점이므로 새로이 구성되는 조각의 구석점이 될 수 없다.

2) 변상의 점의 처리

두 조각이 맞추어질때 두 조각의 조합범위의 끝부분의 구석점들이 만나서 180°를 이루면 이들 점은 새 조각의 구석점이 아니다.

3) 형태의 조정(form adjustment)

두 조각이 조합범위에 따라 맞추어지고 나면 두 조각의 조합범위에 바로 이웃한 구석점들이 이루는 틈새가 생긴다. 이 틈새는 종속조각의 회전과정에서 생기거나 혹은 구석점의 부정확한 설정에 의한 것으로서 합병과정 전에서는 알아내기 힘들다. 따라서 두 조각이 맞추어진 다음에 이 틈새들을 보정하여 주는 형태의 조정과정이 꼭 필요하다. 이를 그림4에 보았다. 즉 그림4. a의 조각 #m과 #s에 대해 설명하면 이 경우 조합범위는 다음과 같다.

조각 #m; range-1=i, range-3=i+1

조각 #s; range-2=j, range-4=j-1

(a) 두 구석점 Pm(i)와 Ps(j)가 다음조건을 만족하면 틈새를 형성한다고 판단한다.

- $|360^\circ - \text{ANG}_m(i) - \text{ANG}_s(j)| \leq \epsilon 1$
- $|\text{DST}_m(i-1) - \text{DST}_s(j)| \leq \epsilon 2$

(b) 위의 두 조각에 대하여 조합범위를 확장시킨다

조각 #m; range-1=i-1

조각 #s; range-2=j+1

(c) 그림4. a의  $\theta$ 는 벌어진 틈새의 각도로서 이 각에 대하여 (b)에서 새로이 구한 조합범위에 합병과정을 수행한다. 한편 점 Pm(i)와 Ps(j)는 사라지는 점으로 처리한다(그림4. b).

(d) 새로이 구성된 조각에 대하여 위의 과정을 다시 실행한다.

위의 방법을 그림4. c에 적용하면 성립하지 않는다. 이 경우는 각 조각의 조합범위의 확장이 달라지게 된다. 즉

조각 #m; range-1=i-1

조각 #s; range-3=k

이 되므로 이 조합범위에 따라서 다시 합병과정을

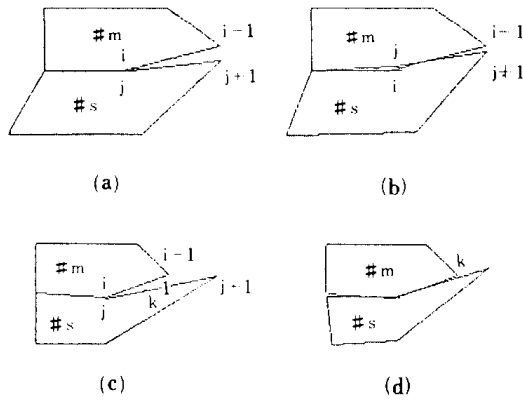


그림 4. 형태 조정이 필요한 조합된 그림조각  
 Fig. 4. Two examples for from adjustment.

수행한다(그림 4. d).

4) 위의 과정을 거쳐서 만들어지는 조각에 대하여 각 구석점에 지정되는 번호와 특징들을 재조정한다.

위의 방법을 적용하다보면 조합범위 내에게 두 개의 조각이 겹치는 경우가 종종 생긴다. 그러나 내부의 겹치는 정도가 심하지 않은 경우에는 전체적인 외형의 모습이 중요하다고 볼 수 있으며, 또한 계속되는 조합과정에서 정확성을 얻을 수 있다.

3장에서 설명한 전체조합 알고리즘의 flow chart를 그림5에 보았다.

#### IV. 실험결과

본 논문에서 제안한 알고리즘을 여러 종류의 그림 조각에 대해 적용하여 실험하였다.

실험시 그림조각은 흰색으로하고 그림조각이 놓여지는 배경은 검은색으로 하여 이진영상을 구성하였다. 입력영상을 얻기위하여 그림 6 과 같이 고정시킨 카메라를 통하여 출력된 신호는 디지털라이저(PCVISION plus Frame Grabber)의 프레임 메모리(frame memory)에 256 명암도로 저장되며, D/A 컨버터를 통하여 모니터에 표시된다. 이와 동시에 parallel Port를 통하여 컴퓨터에 전송된다.

실험결과를 그림7-그림9에 나타내었다. 각 그림에서 (a)는 입력영상을 얻기 전에 그림조각이 놓여진 상태를 나타내며, (b)는 입력영상에서 11-2. 절의 구석점을 이용하여 그림조각을 직선으로 근사화시킨 사람이다. (c)는 제안된 알고리즘을 적용하여 그림조각을 조합한 결과이다.

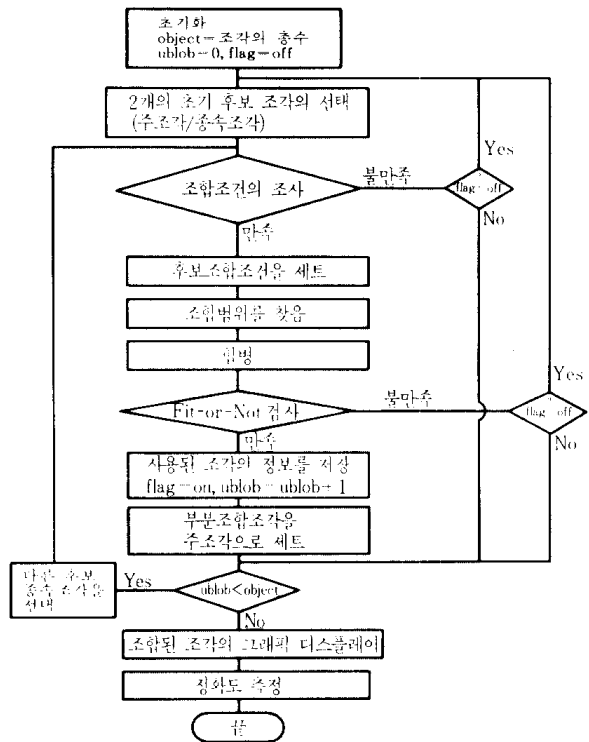


그림 5. 제안된 조합 알고리즘의 흐름도  
 Fig. 5. A flowchart of the proposed puzzle matching algorithm.

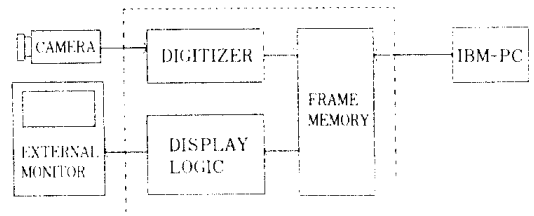
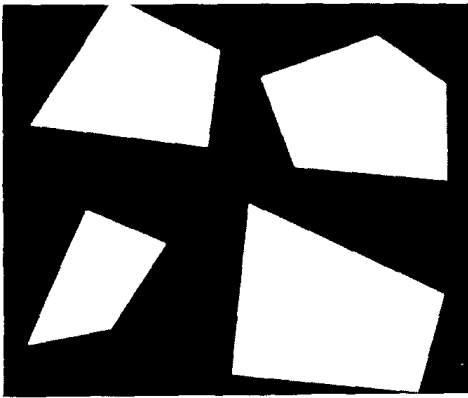


그림 6. 디지털 영상처리 시스템  
 Fig. 6. Digital image processing system.

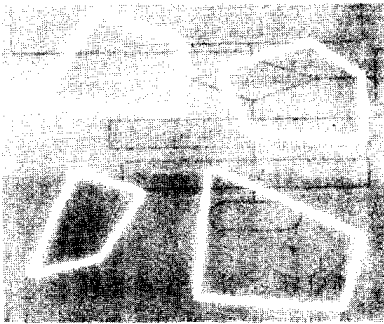
본 실험을 수행하는 과정에서 그림조각이 놓여지는 방향에 따라서 알고리즘의 적용 결과가 좋지 못한 경우가 발생하였다. 이는 한정된 해상도로 인하여 정확한 구석점을 구하는 것이 곤란하기 때문이라고 사료된다.

#### V. 결 론

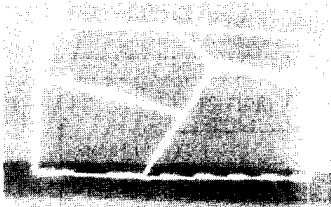
본 논문에서는 카메라로부터 그림조각의 영상을 받아서 그림조각 맞추기를 수행하였다. 전 처리 단계



(a)



(b)



(c)

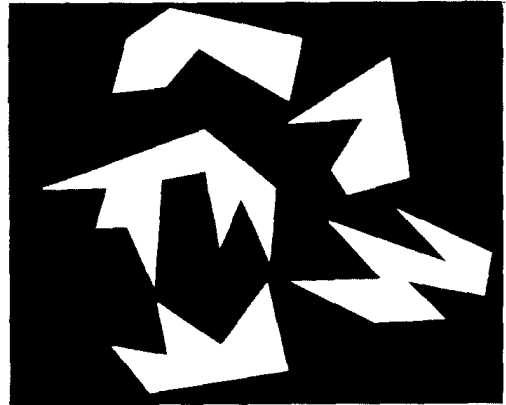
그림 7. 예제 (1)

- (a) 실제 그림 조각들
- (b) 전처리 과정에 의해 표현된 그림 조각들
- (c) 맞추어진 그림 조각들

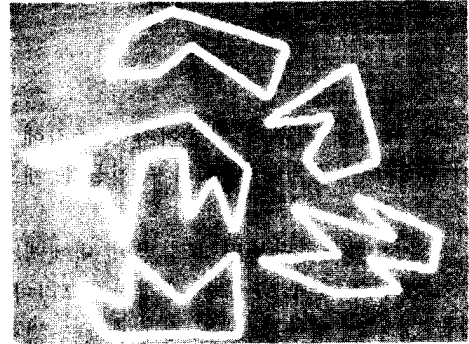
Fig. 7. Example (1).

- (a) Real jig-saw puzzles.
- (b) Pre-processed jig-saw puzzles.
- (c) Matched puzzles.

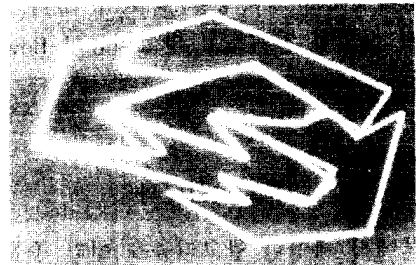
에서는 실험 대상이 되는 물체의 윤곽선 추적(boundary tracking)을 수행하였고 이를 이용하여 그림 조각을 근사화 시키기 위한 구석점들을 구하였다. 또한 조합과정에서 직접적으로 필요한 각 구석점들의 특징으로 볼록성(convexity), 오목성(concavity), 구석



(a)



(b)



(c)

그림 8. 예제 (2)

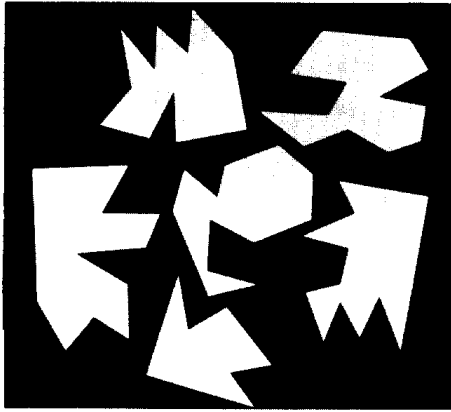
- (a) 실제 그림 조각들
- (b) 전처리 과정에 의해 표현된 그림 조각들
- (c) 맞추어진 그림 조각들

Fig. 8. Example (2).

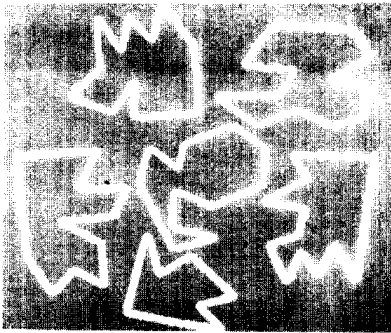
- (a) Real jig-saw puzzles.
- (b) Pre-processed jig-saw puzzles.
- (c) Matched puzzles.

각 및 구석점사이의 거리등을 계산하였으며, 이들을 테이블 형태로 컴퓨터에 저장하였다.

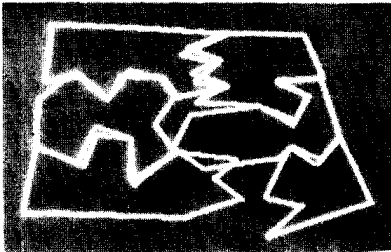
그림조각들을 실제로 조합하는 과정에서는 그림에 대한 사전정보 없이 그림조각 맞추기를 수행하였고 이를 IBM PC용 비전 시스템과 CCD 카메라를 이용



(a)



(b)



(c)

그림 9. 예제 (3)

- (a) 실제 그림 조각들  
 (b) 전처리 과정에 의해 표현된 그림 조각들  
 (c) 맞추어진 그림 조각들

Fig. 9. Example (3).

- (a) Real jig-saw puzzles.  
 (b) Pre-processed jig-saw puzzles.  
 (c) Matched puzzles.

하여 실제 실험하여 결과를 분석하였다.

본 논문에서 다룬 그림맞추기는 그래픽 데이터를 광범위하게 다루고 그 결과를 인식하는 문제들의 대표적인 보기로서 이러한 문제와 관련되어 앞으로 더욱 개발해야 할 분야는 다음과 같다.

- 카메라 렌즈에 의한 외각의 찌그러짐의 방지.
- 윤곽선 조합의 근간이 되는 구석점을 보다 정확하게 찾는 알고리즘의 개발
- 잘못 맞추어진 그림조각을 다시 맞추는 Back Tracking 알고리즘의 개발.

#### 參 考 文 獻

- [1] S. Ando, "Current status and future of intelligent industrial robots," *IEEE Trans. IE*, no. 3, pp. 291-298, 1983.
- [2] W.A. Perkins, "A model-based vision system for industrial parts," *IEEE Trans. on Comput*, vol. 27, no. 2, pp. 126-141, 1978.
- [3] M. Yadhida S. Tsuji, "A versatile machine vision system for complex industrial parts," *IEEE Trans. on Comput.* vol. 26, no. 9 pp. 882-894, 1977.
- [4] S.-R. Oh, J.H. Lee, K.J. Kim and Z. Bien, "An intelligent robot system with jigsaw-puzzle matching capability," Proceedings of 15th ISIR, Tokyo, Japan, September 1985.
- [5] H. Freeman L. Garder, "A pictorial jigsaw puzzles," *IEEE Trans. on Electric Comput*, vol. EC-13, pp. 118-127, 1964.
- [6] G.M. Rodack N.I. Badler, "Jigsaw puzzle matching using boundary centered polar encoding," *C.G.I.P*, no. 19, pp. 1-17, 1982.
- [7] 고순섭, "로봇 시각 센서를 이용한 그림조각의 조합에 관한 연구," KAIST 석사학위논문, 1986.
- [8] R.L.T. Cederberg, "Chain-link coding and segmentation for raster scan devices," *C.G. I.P*, vol. 10, pp. 224-234, 1979.
- [9] H. Freeman, "Computer processing of line-drawing images," *computing surveys*, vol. 8, no. 1, pp. 57-97, 1979.
- [10] H. Freeman L.S. Davis, "A corner-finding algorithm for chain-coded curves," *IEEE Trans. Comput*, vol. 26, pp. 205-216, 1977.



---

 著 者 紹 介
 

---



李 東 柱 (正會員)

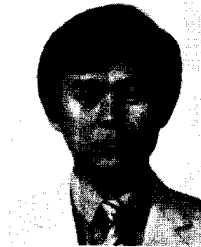
1963年 4月 9日生. 1986年 2月  
 한양대학교 전자공학과 졸업. 19  
 88年 8月 한양대학교대학원 전자  
 공학과 졸업 공학석사학위 취득.  
 1989年 7月~현재 삼성종합기술  
 원 정보시스템연구소 연구2실 근

무. 주관심분야는 robot vision과 neural network임.



徐 一 弘 (正會員) 第26卷 第6號 參照

현재 한양대학교 공과대학 전  
 자공학과 부교수



吳 尚 錄 (正會員)

1958年 6月 7日生. 1980年 2月  
 서울대학교 공과대학 전자공학과  
 공학사 취득. 1982年 2月 한국과  
 학기술원 전기및 전자공학과 공  
 학석사 취득. 1987年 2月 한국과  
 학기술원 전기및 전자공학과 공

학박사학위 취득. 1987年 3月~1988年 1月 한국과학  
 기술원 시스템제어 연구실 Post Doc. 1988年 1月~현  
 재 한국과학기술연구원 제어 시스템 연구실 선임연  
 구원. 주관심분야는 공장자동화, 로보틱스, 로봇트비  
 전 시스템, 기계학습제어 등임.