자동검사 시스템을 위한 컴퓨터 비젼의 연구†

장 동 식*

An Automatic Inspection System Using Computer Vision

Dong-sik Jang*

Abstract

A line search method is developed to locate all the conerpoints of 2-dimensional polygon images for inspection purposes. This optimization-based method is used to approximate a 2-D curved object by a polygon. This scheme is also developed for inspection of objects in industrial environment. The inspection includes dimensional verification and pattern matching which compares a 2-D image of an object to a pattern image. The method proves to be computationally efficient and accurate for real time application.

1. 서 론

컴퓨터 비젼은 로보트 컨트롤 (robot contril)과 부품검사(part inspection) 등에 폭넓은 적용을 하여 왔다. 만약 한 부품이 컴퓨터 비젼 시스템을 이용하여 검사된다고 가정할 때 무엇보다도 한 부품의 영상(image)이 그 부품의 CAD 정보 또는 그와 연관된 데이터 베이스 파일과 관련이 맺어지게 하기 위하여서는 각 부품의 형태 또는 부품번호가 각 영상으로부터 판명 되어져야 한다. 이렇게 얻어진 영상과 부품의 관계가 정립되면 각 부품의 중요한 특징의 숫치적인 정보가 비전시스템을 통하여 얻어지고 이 정보가 CAD 시스템 또는 데이터 베이스의 정보와 비교되는 것이다.

한 2-D 형상의 외각선(boundary) 정보가 패턴 인식 (pattern recognition)이나 부품검사(parts inspection)을 할 때에 만약 주어진 외각선이 직선으로 이루어졌다면 그 외각선상의 극치점 (cornerpoints)만의 인식으로도 시각검사 하는데 있어서 일반적으로 충분할 것이다. 이 경우 한 영상의 외각선이 결정되면 그 외각선을 이루는 직선들의 길이와 각 직선들이 만나는 점-극치점의 내분각들은 쉽게 구하여진다. 하지만 한 영상의 외각선이 직선이 아닌 곡선으로 이루어졌다면 그 곡선부분과 근사한 모양이 되게 다직선(polyline)화 하는 작업이 이루어져야 할 것이다. 이런 곡선을 대치한 polyline은

† 이 논문은 1990년도 한국학술진흥재단의 자유공모과제 학술연구조성비의 연구비 지원으로 이루어졌음.

* 고려대학교 공과대학 산업공학과

부품검사 뿐만 아니라 겹쳐진 부품인식 등에도 이용될 수가 있을 것이다.

한 부품이 일반적으로 기계부품일 경우는 3-D의 정보가 필요하게 될 것이다. 이러한 때에는 정면, 측면, 그리고 윗면의 2-D영상을 종합 이용하여 3-D물체를 나타낼 수 있을 것이다. 한 좋은 예로 원통모양의 물체는 정면인 경우 직사각형으로 나타날 것이고 윗면인 경우 원형으로 나타나는 두개의 orthogonal 방향으로 부터 3-D물체 정보를 얻을 수 있는 것이며 이러한 정보로써 부품이 놓인 방향 등의 정보는 로보트 컨트롤 등에 이용될 수 있을 것이다.

2-D geometric 외각선의 구조를 나타내는 방법은 수많은 방법이 연구되어져 왔다. 그중 polyline 방법이 실현에 쉽기 때문에 많이 이용되어져 왔다. polyline 방법의 최대 관심사는 어떻게 가장 정확한 극치점을 나타내느냐에 달려있다. 기본적인 방법은 merging과 splitting 방법에 근본을 둔다. Merging 기술은 Shirai [5]에 의해 개발되었는데 두 선분(line segments)로 나뉘어지고 한 segment는 현재의 시작 pixel 점과 외각선상의 임의의 한 점으로 부터 몇 pixel전 까지를, 그리고 다른 segment는 현재의 pixel과 그 외각선 상의 임의의 한 점으로 부터 몇 pixel 후를 잇는 선분을 뜻한다. 이렇게 하여 외각선상의 각 pixel에 대하여 이 두 선분의 각을 연속적으로 구한다. 만약 두 선분의 각이 어떤 주어진 한 점에서 임의의 threshold를 넘으면 그 점을 극치점라고 정해 나가는 방법을 말한다.

Tomek [7]는 tolerance-band 방법을 개발하였다. 두개의 정해놓은 tolerance level 만큼 떨어진 평행한 band가 외각선상의 시작 pixel점에 놓여지게 되고 이 두개의 평행한 pixel 사이에 들어갈 수 있는 최대의 긴 곡선 분절을 찾아 나가는 방법이다. 이 외각 곡선은 이 주어진 band사이에서 직선으로 가정을 둘 수 있는 것이다. Roberts [4]는 Last-squared-error best-fit 방법을 각 pixel에 대하여

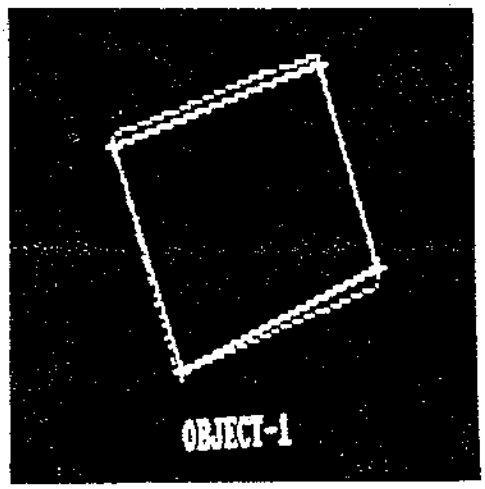

그림 1. Merging 방법을 이용한 직사각형의 극치점

적용함으로써 각 segment를 병합하는데 이용하였다.

Ballard와 Brown [1]은 위의 방법에다가 one-pass merging 기술을 제안했다. 하지만 이 방법은 정확한 극치점을 나타내는데는 별로 성공적이지 못하였다. 그림 1에서 보는 바와 같이 부품 검사에 충분할 정도의 정확도를 나타내지 못하는 것을 볼 수 있다. 이 그림은 직사각형 영상의 극치점을 merging 기술을 이용하여 나타낸 것으로써 십자표시한 곳을 극치점으로 선언했다.

Splitting 방법에 있어서는 한 영상에 있는 물체의 외각선은 초기에 두개의 section으로 나누어지고 각 section을 직선으로 간주할 수 있는지를 조사한다. 만약 주어진 외각선 segment가 직선으로 간주 될때 극치점을 선언하는 방법을 취하고 그리고 그 외각선 segment는 두개의 subsection으로 다시 나누어 지고 계속 조사해 나가는 방법을 취한다. 이러한 절차는 모든 subsection이 모두 조사 될 때까지 반복적으로 진행된다. 자세한 설명은 Duda와 Hart [3] 또는 turner [8]의 논문을 참고할 수 있다.

본 실험에서는 splitting 방법이 merge 방법보다

다각형 (polygon)으로 영상의 외곽선을 나타내는데 더욱 효과적임을 입증할 수 있었다. 특히 영상의 노이즈가 많이 존재할 때는 더욱 큰 차이를 보였다. 그러나 splitting 방법은 많은 계산을 요구하므로 극치점 모두를 구하는데 시간이 많이 소모되는 결점이 발견 되었다.

2. 극치점(cornerpoint) 산출

한 영상에 있는 물체의 외곽선은 외곽선 추적 알고리즘을 이용하여 취할 수 있다[3]. 외곽선 상의 pixel점은 임의의 pixel점을 시작점으로 하여 순차적으로 나열할 수 있다. 그러므로 외곽선 상에서 마지막 pixel로 할당된 pixel 수는 외곽선 상의 전체 pixel점 숫자와 같게 될 것이다. 그러므로 지금부터는 각 pixel 점은 2 pixel의 고유번호를 나타낼 수 있다. 각 pixel 고유번호는 CRT 상의 X-Y좌표의 위치와 연관되어 있고 그 정보는 데이터 파일로 저장될 것이다.

2-1. Splitting 방법

아래에서는 splitting 방법을 수행하는데 대한 절차를 설명한다. Array M()은 극치점에 해당하는 pixel의 고유번호를 저장하기 위해 만들어진다. 초기에는 3개의 pixel의 고유번호-첫번째 pixel점, 마지막 pixel점 그리고 임의의 외곽선 상의 pixel점-를 첫번째 pixel점, 임의의 중간의 pixel점, 그리고 마지막 pixel점의 순서로 array에 저장된다.

처음에는 첫번째와 중간 pixel의 고유 번호와 중간 pixel의 고유 번호를 "a"와 "b"로 주어지고 함수 distance(a, b, c)를 부르게 된다. 그 함수는 "a"와 "b"를 잇는 선분으로부터 가장 멀리 떨어진 점의 고유번호 "k"를 반환하게 된다. 당연히 pixel k에서 ab선분까지의 직선거리가 이미 정해진 threshold 보다는 크다는 조건을 만족해야 한다. 그렇지 않으면 k의 값은 0로 대치되고 "a"와 "b"사이에는 극치점이 존재하지 않는다고 가정된다. 만약 극치점 k가 반환되면 그 값을 array M()에 있는 pixel의 고유번호 a, b 사이에 삽입된다. 그러면 두개의 end pixel점은 "a"와 "k"가 되고 함수 distance()를 다시 부르게 된다. 만약 극치점이 존재하지 않을때는 다음 array "k"와 "b"에 있는 2개의 pixel 고유번호가 함수의 인수로 적용된다. 이러한 절차가 더 이상의 극치점을 발견 할 수 없을때까지 계속 재귀적으로 불려진다.

위의 전체 절차를 나타내는 알고리즘은 다음과 같다.

```
/* 초기 segment의 결정 */
M(1) : =1
M(2) : =Int(0.5 * 외곽선상에 놓인 pixel들의
     수)
M(3) : =마지막 pixel의 고유번호
v    : =    3(=array M()에 들어있는 전체
     숫자
/* 물체 외곽선에 놓은 극치점 발견 */
for i=1 to v-1
   a : =M(i)
   b : =M(i+1)
/* 두 pixel의 고유번호 "a"와 "b"를 함수 dista-
     nce(a, b, k)에 적용 */
     distance(a, b, k)
     while(k>0) do
           for j=v downto i+1
           do M(j+1) : =M(j)
           (Mi+1) : =k
           v : =v+1
           a : =M(i)
           b : =M(i+1)
           distance(a, b, k)
     end {while}
end {for}
```

Merging 방법은 각 pixel을 단 한번의 평가로 충분한 반면에 splitting 방법에 있어서는 각 pixel은 적어도 한번이상은 평가되어져야 한다. 그러므로 splitting 방법의 계산횟수가 자연히 많게 되므로 계산량을 줄이는 것은 매우 중요한 의미를 갖는다.

그림 2에서와 같이 직사각형의 극치점들이 모두 결정되었다 가정을 두자. 맨처음 물체 외곽선의 임의의 2개의 pixel점을 직선으로 연결되는데 그 임의의 점을 그림 2에서 보는바와 같이 y축 최저와 최소값을 각각 "a", "b"로 가정할 수 잇다. 그림 2에서 극치점 "a"부터 "b"까지의 외각 pixel점에 있는 극치점들을 발견하기 위하여서는 "a"와 "b" 사이에 있는 모든 pixel점들이 distacne() 함수가 이용되는 polyline approximation에 적용되어져야 한다. 결국 이 과정은 선분 ac 상에 새로운 극치점 여부를 판단하는 절차가 되는 것이다. ac 선분상에 극치점 조사가 끝나면 선분 cd 상에서의 pixel점들에 대한 거리계산이 계속 될 것이다. 그러므로 극치점 a와 b를 잇는 왼쪽 외각 segment 전체에 대한 평가를 위하여서는 4개의 선분(line segment)이 거리계산을 위한 평가에 이용 되어져야 한다. 또한 오른쪽 외곽선 segment에 대한 평가절차는 왼쪽 외곽선 segment에 대한 평가량 만큼 될 것이다. 전체적으로 8개의 선분이 평가되어 4개의 선분을 가진 전체 외곽선 상에 있는 전체 pixel점 숫자보다도 2배가 많은 수가 된다.

Splting 방법에 의한 계산량은 외곽선 상에 같은 수의 pixel점을 가지고 있다 해도 그 외곽선 상에 있는 극치점의 수에 따라 다양하게 나타남을 알 수 있다. 하지만 만약 외각선에 극치점들이 무작위(random)로 나타난다고 가정한다면 주어진 극치점 수에 대하여 평가되어야 할 pixel 수를 예측할 수 있다. 일반적은 splitting 방법 하에서 v개의 극치점을 가진 외각선에 대하여 평가되어 져야 할 선 segment 수, h(v)는 다음의 식으로 나타난다 :

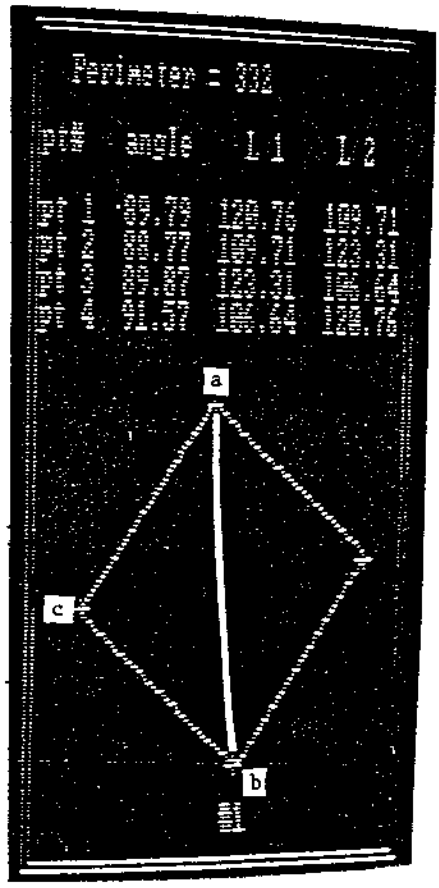

그림 2. 네개의 극치점을 가진 직사각형의 영상

$$H(v)=H(v/2)+H(v/2)+v/2+v/2 \quad \cdots\cdots (1)$$

그리고 H(1)=0이다. 식 (1)이 그림 2에서와 같이 4개의 극치점을 가진 한 polgon에 적용되었을 때 H(v)의 값은 8이 된다. V개의 극치점을 가진 영상에 있는 한 물체 외각선은 v개의 선분을 가지게 되므로 평가 되어져야 할 각 선분의 평균 수 N(v)는 다음의 식으로 나타난다 :

$$N(v)=H(v)/v. \quad \cdots\cdots (2)$$

사실 N(v)는 극치점들을 발견하기 위하여 각 pixel 이 몇번씩 계산에 이용되어 져야 하는 가를 뜻한다고 간주될 수 있다. 예로써 15개의 극치점을

가진 외각선에 대하여 각 pixel은 기존의 splitting 방법이 취할때 대강 4번씩 계산에 사용되어져야 한다.

2-2. Golden Section 방법의 적용

두 외곽선 segment을 만드는 2개의 주어진 외곽선 상의 점 사이에 나타난 appriximating polyline이 있고 또한 이 두 외곽선 segment는 그림 2에서와 같이 하나의 극치점을 가진다고 가정을 하자. 만약 외곽선 상의 모든 pixel 점들이 다각선화하기 위하여 수직으로 투영되었다고 가정을 하면 각 pixel 점들의 거리는 투영된 pixel 점들에 대하여 convex 함수를 형성하게 된다. 각 극치점들은 approximating polyline에 대하여 최대 거리를 가지는 외각선상의 pixel 점들이 될 것이다.

직선화 하는 방법은 효과적으로 극치점들을 찾는데 적용될 수 있을 것이다. 이 문제에 있어서 검토되어야 할 영역은 외곽선 segment로 부터 투영된 불연속 (discrete) pixel 점들로 제한된다. 유도함수를 이용하지 않는 많은 line search 방법 중에는 goden section search 방법과 Fibonacci search 방법이 계산량에 있어서 가장 효과적이다.

Goden section search 방법이 p pixel 점을 가진 외곽선 segment에 적용되었을 때 계산되어져야 할 pixel 수 q는 아래와 같이 주어진다 :

$$q > 4.78 \log p\text{-}1. \quad \cdots\cdots (3)$$

예로써 한 외곽선 segment가 200 pixel 점을 가지고 있다면 한 극치점은 단지 11 pixel 점을 계산함으로써 발견될 수 있다. 감소 비율 (reduction ratio) B(p)는 다음과 같다 :

$$B(p) = (4.78 \log p\text{-}1)/p. \quad \cdots\cdots (4)$$

위의 200 pixel 점을 가진 외곽선 segment에 대하여 B(200)=0.05는 golden section 방법은 기존의 splitting 방법의 단지 5%의 계산량을 요구한다는 것을 의미한다. 이 5%의 계산량은 20배의 검사속도를 의미하며 B(p)의 값은 외곽선 segment의 크기가 증가함으로써 더욱 감소할 것이다. 그림 3은 goden section 방법을 이용한 위의 연구 결과를 이용하여 직선 및 곡선 물체에 대한 polygonal approximation을 보여주는 것이다. 십자 표시한 것이 극치점을 나타내는데 사용되었다.

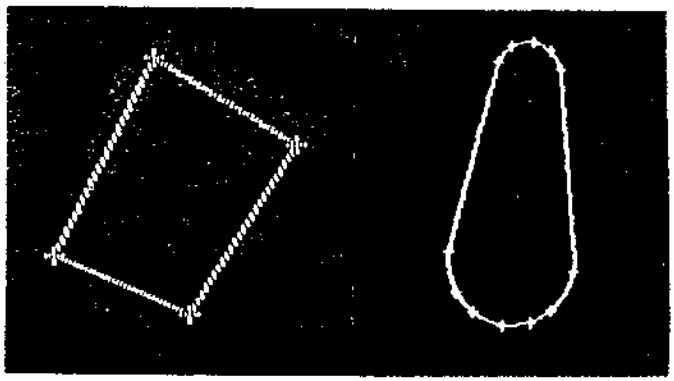

그림 3. 두 물체의 polygonal approximation

3. 자동 검사 시스템을 위한 pattern matching 방법

만약 검사 되어져야할 한 영상 (test image)과 양품의 기준 영상(pattern image)이 있다고 가정을 한다면 두 영상의 중심점(centroids)과 놓인 방향(orientations)이 주어지면 translation과 rotation을 하여 두물체를 겹쳐 비교해 봄으로서 검사를 수행할 수 있을 것이다. 한 영상의 중심점(x_c, y_c)은 아래와 같이 계산 되어진다.

$$x_c = \frac{1}{n} \sum_{i \in P} xi$$

$$y_c = \frac{1}{n} \sum_{i \in P} yi \,, \quad \cdots\cdots (5)$$

P는 한 영상에서의 모든 pixel들의 집합이다. 또한 놓인 방향은 least inertia 축을 이용하면 한 주어진 영상의 독특한 방향을 나타낼 것이다. 이러한 least inertia 축은 또한 다음과 같이 정의 될 수 있다.

$$a = \sum_{i \in P} (x_i - x_c)^2$$

$$b = \sum_{i \in P} 2(x_i - x_c)\ (y_i - y_c)$$

$$c = \sum_{i \in P} (y_i - y_c)^2$$

$$\cos(2\theta) = \frac{(a-c)}{\sqrt{b^2 + (a-c)^2}}$$

$$\theta = \begin{cases} \dfrac{tan^{-1}\left(\frac{b}{a-c}\right)}{2}, & \text{if } \cos(2\theta) \geq 0; \\ \dfrac{tan^{-1}\left(\frac{b}{a-c}\right)}{2} + \pi/2, & \text{otherwise.} \end{cases} \quad \cdots\cdots (6)$$

3-1. 가설 검정

만약 두 영상이 같은 물체로 부터 취하여져도 영상 처리과정에서 나타나는 외부 노이즈와 sampling과 transformation 과정에 나타나는 round-off 에러 등으로 정확하게 동일하게 나타나지는 않는다. 이러한 시스템 에러의 크기는 검사 되어져야 할 영상에 있는 물체가 주어진 허용계치(tolerance level)안에서 양품으로 판정될 것인지를 결정하기 위하여 측정되어져야 한다.

이러한 시스템 에러를 측정하는 한 방법으로서는 한 물체를 여러번 임의로 놓여지게 하여 영상처리 및 transformation 한 후 극치점들의 거리의 차이를 취함으로서 시스템 에러를 측정하는 것을 생각할 수 있다. 이러한 차이는 영상 물체의 모양에 따라 틀릴것이며 각 영상에 나타난 샘플 물체의 극치점들의 오차는 평균값과 편차를 가질것이다.

3-1-1. 데이터 수집 방법

시스템 에러를 측정하기 위하여서는 한 물체의 기준이 되는 영상 정보(또는 CAD 데이터)와 그 물체를 임의의 위치(location)와 방향(orientation)으로 있게한 다음 취한 영상에 나타난 물체를 tranformation을 하여 기준 영상과 포개어 놓았을때 나타나는 에러를 계산해야 하는데 각각을 영상 1과 영상 2라고 하자. 영상 1과 영상 2에는 물체의 극치점들을 위에서 제안한 방법으로 찾아낸 다음 영상 1에 있는 극치점들은 영상 2의 외곽선 상에서 얼마나 떨어졌나 하는 거리 간격 δ_i는 아래와 같이 계산된다.

$$\delta_i = d_{ic} - d_{i'c}, \quad \cdots\cdots (7)$$

여기서 d_{ic}는 영상 1에서의 물체에 대한 극치점 i로 부터 중심점 c까지의 거리를 나타내며 $d_{i'c}$는 영상 2에서의 물체에 나타나는 외곽선 상의 pixel 점 i'로 부터 중심점 c까지의 거리를 나타낸다. 만약 n_i 극치점이 영상 1에 있다면 n_i개의 거리간격 정보가 산출 될 것이다.

또한 djc는 영상 2에서의 물체에 대한 극치점 j로

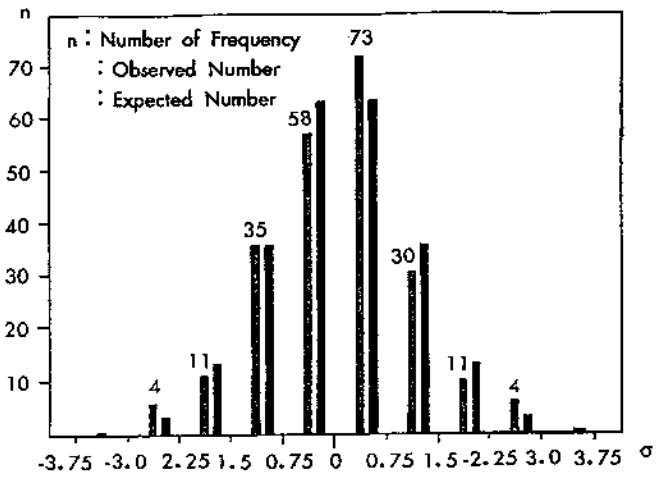

그림 4. 시스템 에러의 분포도

부터 중심점 c까지의 거리를 나타내며 $d_{j'c}$는 영상 1에서의 물체에 나타나는 외곽선 상의 pixel 점 j'로부터 중심점 c까지의 거리를 나타내는 식 δ_j는 아래와 같이 계산된다.

$$\delta_j = d_{jc} - d_{j'c}, \quad \cdots\cdots (8)$$

이런 과정을 통하여 시스템 에러의 평균과 표준편차는 한 샘플에서 (n1+n2) 개의 이격 거리를 나타내는 데이터 점들을 이용하여 다음과 같이 나타나진다.

$$u = \frac{1}{n1+n2}\left(\sum_{i=1}^{n1}\delta_i + \sum_{j=1}^{n1}\delta_j\right)$$

$$\sigma^2 = \frac{1}{n1+n2}\left(\sum_{i=1}^{n1}(\delta_i-u)^2 + \sum_{j=1}^{n1}(\delta_j-u)^2\right) \cdots (9)$$

이러한 시스템 에러의 평균과 표준편차의 신뢰성을 높이기 위해서는 여러 샘플을 취하여 더많은 데이터 점들을 취하는 것이 바람직 할 것이다. 실제로 10개의 그림 2의 물체를 샘플로 취하여 나타난 이격 간격을 나타내는 데이터 점들은 그림 4와 같이 나타났다. 카이 2승 테스트를 이용한 goodness of fit 테스트는 취하여 진 데이터가 정규분포의 성질을 나타냄을 보여준다.

3-1-2. 검정 과정

가설 검정을 위한 통계적 검사 방법은 허용한계를 설정해야 한다. 그러므로써 각각의 이격 간격이 허용한계안에 있는지를 테스트함으로서 검정 과정을 수행할 수 있다. 만약 테스트 이미지에서 n_t의 극치점이 있고 기준 이미지에서 n_p의 극치점이 있다면 (n_p+n_t)개의 이격 간격중에서 어느 하나가 허용오차를 넘으면 그 테스트 이미지에 있는 물체는 허용한계를 넘은 불량으로 판단될 것이다.

이격간격의 상한규격 U와 하한규격 L은 이격간격의 분포도를 이용하여 다음과 같이 나타난다.

$$L = u - k(p)\sigma$$
$$U = u + k(p)\sigma \quad \cdots\cdots (10)$$

여기서 k(p)는 시스템 에러의 이격간격이 두 규격값 안에 p 확률로 있기 위한 요인으로 정의된다. k(p)의 값은 가설검정에서 제시된 신뢰수준 1-α와의 관계는 다음과 같다.

$$(P)^{np+m} \geq 1-\alpha \quad \cdots\cdots (11)$$

같은 신뢰수준에서 데이터 점 np+nt가 많아지면 p의 값은 증가되며 따라서 동시에 k(p)은 큰값이 될 것이다.

4. 실험결과

비젼시스템은 IBM/386 호환기종에 vedio input으로 하나의 CCD 카메라와 Imaging Technolgy Inc의 Frame Grabber 영상처리용 보드가 장치되어져 있다. 물체는 512×512 해상도로 처리되며 흰색의 배경위에 놓여져 있으며 software는 C언어가 이용되었다.

그림 5는 테스트 물체가 기준되는 물체와 비교되는 검사과정을 나타낸다. 그림 5-a에서 기준되는 물체의 영상이 왼쪽에 그리고 테스트 물체는 오른쪽에 위치하며 테스트 물체의 극치점들은 rotation과 trnslation되어 기준 물체에 겹쳐진 상태를 나타낸다. 테스트 물체의 극치점들의 위치를 표시하

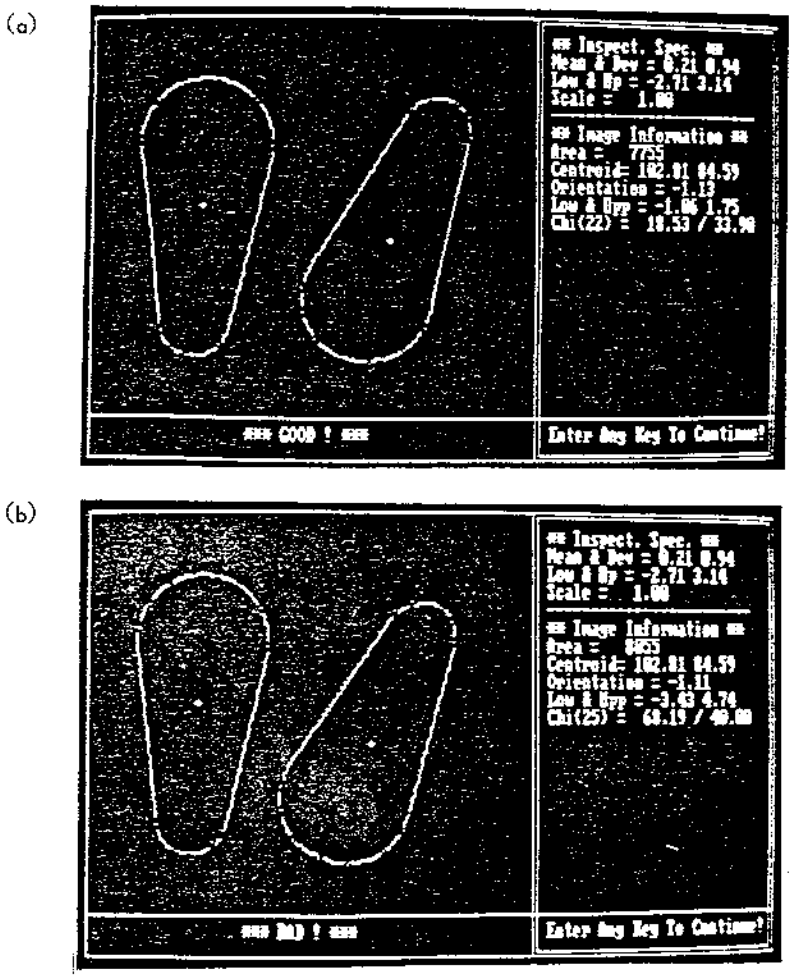

그림 5. 곡선 물체의 검사

기 위하여 기준 물체의 외곽선상의 pixel들의 일부분은 지워진 상태이다. 만약 테스트 물체가 기준 물체와 동일하다면 transformation에 따른 시스템 에러는 앞에서의 연구결과와 같이 정규 분포를 가지게 될 것이다. 그림 5-a의 두 번째 줄에서와 같이 평균 값이 0.21이고 표준 편차가 0.94이다.

식 10와 식 11을 이용하여 95%의 신뢰수준에서 상한 규격과 하한 규격 (−2.71, 3.14)이 세번째 줄에서와 같이 설정되었다. 이격간격을 나타내는 22 data점 (n_p+n_i)이 이러한 통계에 이용되었고 상한과 하한의 최대 이격 간격은 1.75와 −1.06(10번째 줄)이 되며 이 수치는 상한 규격과 하한 규격안에 있으므로 가설 검정의 결과는 "양품"으로 판정 되었다.

그림 5-b에서는 두 최대 이격 간격이 4.74와 −3.43이 되어 상한 및 하한 규격 밖에 있으므로 "불량"으로 판정 되었다.

5. 결 론

극치점을 이용하여 점 패턴 (point pattern)을 대응하는 방법으로 컴퓨터 비젼을 이용한 검사방법이 개발되었다. 이러한 검사방법은 치수 검증 (dimensional verification)외에도 물체의 2-D 형상을 서로 비교하는데에도 이용될 수 있다. 본 연구에서 제안한 방법은 최소한의 계산량으로 치수 검증을 위하여 sub-pixel의 정확도를 보여준다.

3-D 물체의 검사를 위하여 그 물체가 세 직교 방향으로 보여진 세 2-D 영상으로 표현되므로 본 연구에서 제안된 검사 방법을 이용하여 3-D 물체의 검사에 이용할 수 있다. 형상 대응을 위한 본 연구의 검사 방법은 실제 산업환경에서 나타나기 쉬운 겹쳐진 물체의 인식을 위하여서도 이용될 수 있을 것이다.

참고문헌

[1] Ballard and Brown, *Computer Vision*, Prentice-Hall, Inc., New Jerscy, 1982.

[2] Chin and Dyer, Model-Based Recognition in Robot Vision, *Computing Surveys*, Vol. 18, No. 1, pp.68-108, 1986.

[3] Duda and Hart, *Pattern Recognition and Scene Analysis*, John Wiley & Sons, New York, 1973.

[4] Roberts, Machine Percenption of Three-dimensional Solids, *Optical and Electro-optical Information Processing*, J.P. Tippett et al (Eds), MIT Press, Cambridge, MA., 1965.

[5] Shirai, Edge Finding Segmentation of Edges and Recognition of Complex Objcts, In Proceeding of the 4th International Joint Conference on *Artificial Intelligence*, Kaufmann, Los Altors, Calif., pp.674-681, 1975.

[7] Tomek, Two Algorithms for Piecewise Linear Continous Approximation of Funtion of One Variable, *IEEE Trans. Computers*, Vol. 23, No. 4, pp.445-448, 1986.

[8] Tuner, Comper Perecption of Curved Object Using a Television camera, Ph.D. Dissertation, Univ. Edinburgh, 1974.