

## 컨베이어 벨트 시스템에서의 부품 처리를 위한 로봇과 시각 시스템의 접속

박태형 박충수 이범희 이상욱 고명삼  
서울대학교 제어계측공학과

Robot and Vision System Interface for Material Handling on Conveyor Belt System

T.H.Park C.S.Park B.H.Lee S.U.Lee M.S.Ko

Dept. of Control & Instrumentation Eng.

Seoul National University

### ABSTRACT

The robot system which can handle a stream of randomly positioned parts on a conveyor belt system, is developed. It is composed of a PUMA 560 robot, a conveyor belt system and a vision system. The performance of the overall system is mainly dependent upon the robot and vision system interface technique. A vision algorithm is developed to determine the position, orientation and type of the part. Calibration procedure and the vision-to-robot transformation are also proposed. Experimental results are then presented and discussed.

### 1. 서 론

컨베이어 벨트로 흐르는 부품을 로봇이 처리하는 것은 로봇의 중요한 활용 분야 중의 하나이다. 이때 로봇은 다양한 형태와 이송 자세를 갖는 부품을 능동적으로 처리할 수 있는 능력이 요구되며, 이를 위하여 시각 시스템 (vision system) 의 도입이 필수적이다. 그러나 시각 시스템은 작업 조도의 외란, 낮은 처리 속도 등으로 인하여, 그 응용이 제한된다. 컨베이어 벨트를 포함한 로봇 작업 시스템에 응용되기 위하여, 시각 시스템은 빠른 처리 속도를 갖고 조도의란에 강인하여야 하며, 로봇과 결합시키는 적절한 방법이 개발되어야 한다.

UNIVISION 시스템 [8] 은 Unimation 사의 PUMA 로봇과 Machine Intelligence 사의 VS-100 시스템을 결합하여 개발되었다. GM 사의 CONSIGHT-1 [7] 은 로봇과 컨베이어 벨트, 리니어 어레이 카메라 (linear array camera) 로 구성된 부품 처리용 작업 시스템이다.

본 논문은 새로 개발된 로봇 작업 시스템을 소개한다. 작업 시스템은 PUMA 560 매니퓰레이터와 컨베이어 벨트, 시각 시스템, 주변 센서로 구성되며, 전체 시스템은 주 컴퓨터인 PC/AT 에 의하여 통합 제어 된다. 개발된 작업 시스템은 컨베이어 벨트로 흐르는 다양한 형태와 자세의 부품을 분류 처리할 수 있는 능력을 갖고있다. 작업 시스템의 성능은 시각 시스템의 인식 정확도와 인식 소요 시간 및 시각 정보의 정확한 활용에 따라 평가될 수 있다. 본 논문에서는 시각 시스템의 부품 인식 방법과 시각 정보의 변환 방법에 관해 논한다. 전체 작업 시스템의 구성과 부품 인식 알고리즘이 제안되며, 시각 정보를 로봇에 활용하는 방법, 작업 시스템의 캘리브레이션 방법을 제시한다. 실험 결과를 보이고, 발생된 오차의 원인을 분석한다.

### 2. 작업 시스템의 구성 및 문제의 제기

컨베이어 벨트로 이송되는 부품을 처리하기 위하여 구현된 작업 시스템은 그림 1 과 같다. 작업 시스템은 컨베이어 벨트 시스템, PUMA 560 로봇 매니퓰레이터, 시각 시스템으로 구성된다. 각 구성 요소는 계층적으로 구성된 제어기에 의하여

제어된다. 컨베이어 벨트는 1 칩 마이크로 컨트롤러인 8031 프로세서 보드에 의하여 이송 속도와 이송 위치가 제어된다. 이 마이크로 컨트롤러는 주 컴퓨터인 PC/AT 에 의하여 제어된다. 컨베이어 벨트위에 장착된 광센서는 카메라의 촬영 영역 내에 부품이 이송되었는지의 여부를 감지한다. 시각 시스템은 CCTV 카메라와 조명 장치를 포함하며, 카메라에서 포착된 부품의 아날로그 영상 신호는 디지털이제에 의하여 512 x 512 x 8 비트의 디지털 정보로 변환된다. 시각 프로세서인 PC/XT 는 이 디지털 정보를 처리하여 부품의 형태, 위치, 오리엔테이션을 인식한다. PUMA 560 매니퓰레이터의 서보 제어기는 주 컴퓨터로부터 입력된 관절 변위 값에 의하여 각 관절의 모터를 구동시킨다. [11]

시각 프로세서에 의하여 인식된 부품의 위치와 오리엔테이션은 카메라 영상 평면에서의 좌표값이며, 로봇에 이 값을 직접 입력 시킬 수 없다. 따라서 주 컴퓨터는 시각 프로세서로부터 전송된 값을 작업 공간에서의 로봇의 관절 변위 값으로 변환시켜야 한다. 또한 이상적이 아닌 실제 작업 환경에서의 부품의 시각 처리는 조명 상태, 그림자 등의 영향을 심하게 받으며, 이를 극복할 수 있는 부품 인식 알고리즘이 필요하다. 본 논문은 이러한 문제를 해결하고, 실제 작업 환경에의 적용이 가능한 하나의 방법을 제시한다.

### 3. 부품 검색 알고리즘

#### 3.1 잡는 위치 (Gripping Position) 의 결정

이 절에서는 시각 장치를 이용하여 부품의 잡는 위치 (gripping position)을 결정하는 방법에 대하여 설명한다. 먼저 카메라를 이용하여 영상 데이터를 취득한 후, 로우 패스 필터링 (low pass filtering)을 거친 입력 영상의 그레이 레벨 히스토그램의 밸리 (valley)를 찾는 간단한 방법으로 영상 영역화를 수행하여 부품을 배경으로부터 구분해낸다. [1,2] 이 경우, 부품과 배경이 뚜렷이 구분될 수 있도록 조명 상태를 미리 조정해 두어야 좋은 영상 영역화 결과를 얻을 수 있다.

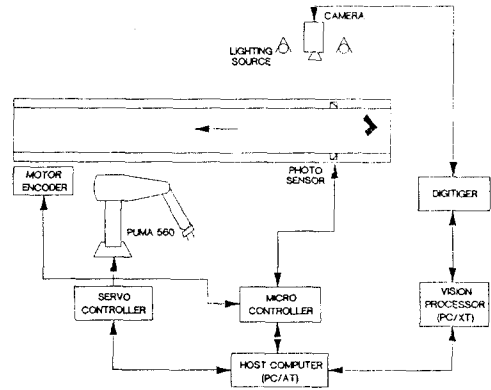


그림 1 작업 시스템의 구성도

윤곽선 추출 알고리즘을 이용하여 부품의 윤곽을 찾아낸 후, 둘레를 나타내는 윤곽선을

$$\{P(i) \mid P(i) = (x(i), y(i)), i=1, \dots, N\}$$

으로 표현한다. 여기서 N은 부품의 둘레를 구성하는 점들의 수를 나타낸다. 이 때 다음의 식을 이용하여 검색하는 부품의 중심점 (Cx, Cy)를 계산한다.

$$C_x = \left( \sum_{i=1}^N x(i) \right) / N \quad (1)$$

$$C_y = \left( \sum_{i=1}^N y(i) \right) / N$$

또한 부품의 오리엔테이션은 다음의 식에 의하여 구한다.

$$\theta = (1/2) \operatorname{atan}\{2\mu(1,1)/(\mu(2,0)-\mu(0,2))\} \quad (2)$$

단,

$$\mu(p,q) = \sum_{i=1}^N \{(x(i)-C_x)^p (y(i)-C_y)^q\}$$

다음으로 주어진 윤곽선에 대하여, Wall [4]의 알고리즘을 이용하여 다각형 근사화를 수행한다. 다각형 근사화시킨 결과를 특징으로 사용하는 이유는 물체의 서로 다른 부분의 독립적인 인식이 가능하고, 물체의 변위와 회전 변화를 정확히 알아낼 수 있다는 장점이 있기 때문이다.

마지막으로 다각형 근사화 과정을 통해 얻어진 선분들 중 길이, 주위 선분과 이루는 각도, 부품의 방향과 이루는 각도의 차이를 이용하여 두개의 주선분을 선택하고, 이 주선분 위에서 중심점(center of mass)과의 직교점을 구하여 이 점을 부품의 잡는 위치(gripping position)로 결정한다. 그림 2는 컨베이어 벨트를 흐르는 부품의 입력 영상으로부터, 부품의 방향, 중심점 그리고 잡는 위치를 구한 예이다.

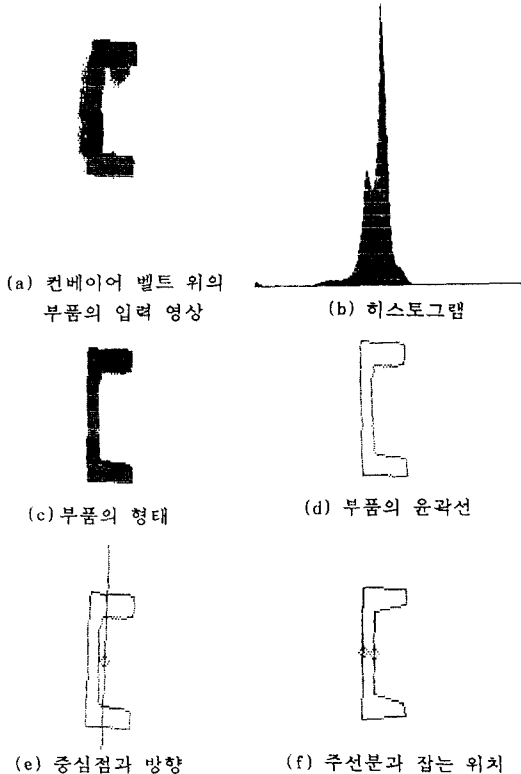


그림 2 부품 검색 과정

### 3.2 부품의 종류 결정

본 논문에서 대상으로 하는 부품의 입력 영상은 2차원 물체로 간주할 수 있다. 2차원 물체를 인식하기 위하여 제안된 여러가지 알고리즘 중, FD (Fourier Descriptor) [5,6]를 이용하여 부품의 종류를 판정한다.

부품의 둘레를  $\{(x(n), y(n)), n=0, \dots, N-1\}$ 으로 나타낼 때, 함수  $u(n)$ 을 다음과 같이 정의한다.

$$u(n) = x(n) + j y(n) \quad (3)$$

$u(n)$ 의 DFT (Discrete Fourier Transform) 표현은 다음과 같다.

$$u(j) = (1/N) \sum a(k) \exp(j2\pi kn/N) \quad 0 \leq n \leq N-1$$

$$a(k) = \sum u(n) \exp(-j2\pi kn/N) \quad 0 \leq k \leq N-1 \quad (4)$$

식 (4)에서 복소수 값을 가지는 계수  $a(k)$ 를,  $u(n)$ 으로 표현되는 물체의 둘레의 FD 라고 한다. 이 FD 는 같은 물체에 대해서도 여러 변화에 의해 서로 다른 값들을 나타낸다. 즉, 물체에 병진 이동, 크기 변화, 시작점 변화, 회전 이동이 일어났을 경우에는 FD 값이 변한다.

이러한 성질을 가진 FD는 크기나 방향 변화에 관계 없이 비슷한 형태를 가진 두 개의 형상을 정합시키는데 사용될 수 있다. 두 개의 물체의 둘레를 나타내는  $u(n)$ ,  $v(n)$ 의 FD를 각각  $a(k)$ ,  $b(k)$ 라 했을 때, 두 물체의 형태는 다음의 식에 의하여 정의된 거리  $d$ 가 작을수록 유사하다.

$$d(u_0, \beta, \theta_0, n_0) = \operatorname{Min}_{u_0, \beta, \theta_0, n_0} \{ \sum |u(n) - u_0 - \beta v(n+n_0) \exp(j\theta_0)|^2 \} \quad (5)$$

식 (5)에서  $u_0, \beta, \theta_0, n_0$ 는 각각 병진 이동, 크기 변화, 시작점 변화 및 회전 이동에 의한 영향을 최소화 시키도록 선택된다. 만약  $u(n)$ ,  $v(n)$ 을  $\sum u(n) = \sum v(n) = 0$ 이 되도록 정규화시키면, 식 (5)에 의해 정의된 거리는, 다음 식과 같은  $u_0, \beta, \theta_0$  값을 가질 경우에 최소화 된다.

$$u_0 = 0$$

$$\beta = \frac{\sum c(k) \cos(\phi(k) + k\varphi + \theta_0)}{\sum |b(k)|^2} \quad (6)$$

$$\tan \theta_0 = \frac{\sum c(k) \sin(\phi(k) + k\varphi)}{\sum c(k) \cos(\phi(k) + k\varphi)}$$

위의 식에서  $c(k) \exp(j\phi(k)) = a(k)b(k)$ ,  $\varphi = -2\pi n_0/N$ 이며,  $c(k)$ 는 실수 값을 가진다. 식(6)에 의해 얻어진  $u_0, \beta, \theta_0$ 에 의하여 FD  $a(k), b(k)$  간의 최소 거리  $d$ 는 다음 식과 같이 주어진다.

$$d = \operatorname{Min}_{\varphi} [d(\varphi)]$$

$$= \operatorname{Min}_{\varphi} \{ \sum |a(k) - \beta b(k) \exp(j(k\varphi + \theta_0))|^2 \} \quad (7)$$

식(7)에 의하여 주어지는 최소 거리  $d$ 에 의하여  $a(k), b(k)$ 로 나타내어지는 두 둘레  $u(n), v(n)$ 의 형태의 차이를 측정할 수 있다. 이를 이용하여 컨베이어 벨트 위에 흐르는 부품의 FD 값들과 저장해 놓은 모델들의 FD 값들의 거리를 계산한다. 이 값이 최소가 되는 모델을 부품의 형태로 판단한다.

### 4. 부품 처리 알고리즘

#### 4.1 월드 모델링 (World Modeling)

시각 시스템에서 얻어진 정보를 이용하여 로봇트가 부품을 처리하기 위하여, 영상 평면의 정보를 로봇트 핸드의 위치와 오리엔테이션으로 변환시켜야 한다. 이를 위하여 전체 작업 시스템의 월드 모델링이 필요하다.

작업 시스템의 구성 요소인 PUMA 560 매니플레이터와 카메라, 부품의 좌표계를 그림 3과 같이 설정한다. (Rx, Ry, Rz)는 PUMA 560 매니플레이터의 기준 좌표계로서 전체 작업 시스

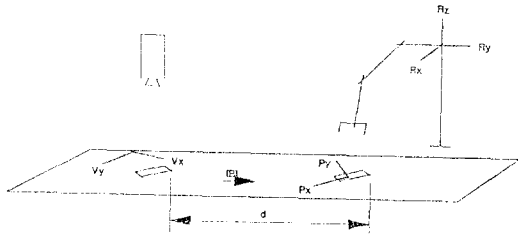


그림 3 작업 시스템의 월드 모델링

템의 기준 좌표계로 사용된다.  $(V_x, V_y, V_z)$  는 기준 좌표계에 대한 카메라의 좌표계로서,  $V_x$  는 영상 평면의  $x$  축,  $V_y$  는 영상 평면의  $y$  축으로 설정한다. 이때  $V_z$  가  $R_z$  와 평행하도록 카메라를 설치한다.  $(P_x, P_y, P_z)$  는 처리점에서의 기준 좌표계에 대한 부품의 좌표계로서, 부품을 처리 할 때의 로봇트 핸드의 좌표계와 같다.  $P_x$  는 로봇트 핸드의 normal 벡터 방향,  $P_y$  는 sliding 벡터 방향,  $P_z$  는 approach 벡터 방향과 일치되어야 한다.  $b$  는 컨베이어의 이동 방향을 나타내는 벡터이며,  $d$  는 부품의 인식점과 처리점 사이의 거리이다.

#### 4.2 로봇트와 시각 시스템의 접속

3 절에서 기술된 부품 인식 알고리즘은, 카메라로부터의 영상 신호를 처리하여 부품의 중심 위치와 오리엔테이션 및 부품의 형태 정보를 얻는다. 그러나 부품의 중심 위치와 오리엔테이션은 카메라 좌표계에서의 값이다. 이 값들은, 로봇트가 부품을 처리하기 위하여, 기준 좌표계에 대한 로봇트 핸드의 위치와 오리엔테이션으로 변환되어야 한다. 즉 시각 시스템과 로봇트를 접속시키는 알고리즘이 필요하다.

앞서 정의된 작업 시스템의 월드 모델링에 의하여 다음의 식이 성립한다.

$$[P] = [T_v] [V] + s d [B] \quad (8)$$

단,  $[P]$  는 기준 좌표계에 대한 처리점에서의 부품의 좌표계를 나타내는 제차 변환 행렬 이고,  $[V]$  는 카메라 좌표계에 대한 인식점에서의 부품의 좌표계를 나타내는 제차 변환 행렬 이며,  $[T_v]$  는 기준 좌표계에 대한 카메라 좌표계를 나타내는 제차 변환 행렬이다.  $[B]$  는 컨베이어 벨트의 이동 방향을 나타내는 제차 변환 행렬,  $d$  는 부품의 인식점과 처리점 사이의 거리이고,  $s$  는 로봇트 거리와 실제거리의 비율이다.

식(8) 은 시각 시스템에서 인식된 부품의 중심 위치와 오리엔테이션  $[V]$  에 의하여, 부품의 처리를 위한 로봇트 핸드의 오리엔테이션  $[P]$  가 결정됨을 의미한다. 이 때 각 제차 변환 행렬의 모든 원소는 로봇트의 거리 개념으로 표현되나  $d$  는 실제 거리이다. 따라서 로봇트의 거리로 변환하기 위한 비율  $s$  가 정의되어야 한다. 부품 인식후의 이송거리  $d$  를 명확히 하며 부품을 처리할 수 있으며, CONSLIGHT-1 [7] 은 위의 식에 의하여 부품을 처리한다. 그러나 이 식을 구현하기 위해서는  $[T_v]$ ,  $[B]$ ,  $s$  가 작업 시작 전에 캘리브레이션되어야 한다. 모든 캘리브레이션 과정에는 오차가 발생되며, 캘리브레이션 대상이 많으므로 인하여 오차가 누적될 수 있다. 또한 캘리브레이션에 많은 시간이 소요된다.

본 논문에서는 이러한 문제점을 개선하기 위하여 변환된 식을 사용한다. 식(8)에서  $[B]$  는 병진 항만을 갖고 있는 행렬이므로 다음과 같이 변환시킬 수 있다.

$$[P] = s d [B] [T_v] [V] \quad (9)$$

이송거리  $d$  가 일정한 경우 식(9)는 다음의 식과 같다.

$$[P] = [T] [V] \quad (10)$$

단,  $[T] = s d [B] [T_v]$  이며 변환 행렬이라 정의한다. 식

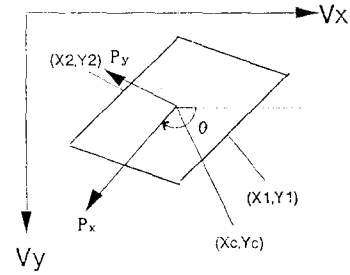


그림 4 시각 시스템의 정보와 부품 좌표계의 관계

(10)의 구현은  $[T]$  만의 캘리브레이션에 의하여 가능하다. 즉 카메라 좌표계에 대한 부품의 좌표계는 변환 행렬에 의하여 직접 부품의 처리를 위한 로봇트 핸드의 좌표계로 변환된다.

그림 4 는 시각 시스템에 의하여 전송된 데이터를 카메라 좌표계 상에서 표시한 것이다. 부품의 중심 위치  $(x_c, y_c)$  와 잡는 위치  $(x_1, y_1)$ ,  $(x_2, y_2)$  의 6개 데이터가 전송되며, 이들은 영상 평면 상에서의 좌표값이다. 그러나 이들 데이터의 크기는 화면 상의 거리를 의미한다. 즉 중심 위치  $(x_c, y_c)$  는 영상 평면 상의  $x$  축 ( $V_x$ )으로  $x_c$  pixel,  $y$  축 ( $V_y$ )으로  $y_c$  pixel 의 위치를 나타낸다. 따라서 화면 1 pixel 간의 간격을 실제 거리로 환산시키기 위한 스케일 인자(scale factor) 가 필요하다. 실제로 카메라 배율의 조절, 카메라 고정 높이의 변화에 따라 스케일 인자는 변화한다.  $x$  축의 스케일 인자를  $a_x$ ,  $y$  축의 스케일 인자를  $a_y$  라 정의한다.

카메라 좌표계에서의 부품의 중심 위치와 오리엔테이션을 나타내는 제차 변환 행렬  $[V]$  는, 하나의 병진 행렬과 하나의 회전 행렬에 의하여 구할 수 있다. 즉,

$$[V] = \text{Trans}(a_x x_c, a_y y_c, 0) \text{Rot}(2, \theta) \quad (11)$$

이 때 카메라 좌표계의  $x$  축 ( $V_x$ ) 과 인식 위치에서의 부품 좌표계의  $x$  축 ( $P_x$ ) 사이의 회전각  $\theta$  는 다음과 같이 구한다.

$$\theta = \begin{cases} \pi/2 + \text{atan}((y_2 - y_1)/(x_2 - x_1)) & , (x_2 - x_1)(y_2 - y_1) \geq 0 \\ 3\pi/4 + \text{atan}((y_2 - y_1)/(x_2 - x_1)) & , (x_2 - x_1)(y_2 - y_1) < 0 \\ \pi/2 & , x_2 = x_1 \end{cases} \quad (12)$$

변환 행렬  $[T]$  와 스케일 인자  $a_x, a_y$  를 작업 시작 전에 캘리브레이션한 후, 식(11)-(12)에 의하여  $[V]$  를 구하고, 식(10)에 의하여 부품의 처리를 위한  $[P]$  를 계산한다. 컨베이어 벨트로 이송되는 부품의  $[V]$  는 인식점에서 시각 시스템에 의하여 얻어진다.  $[P]$  는 이 인식점으로부터 미리 지정된 거리  $d$  만큼 컨베이어 벨트로 이송시킨 처리점에서의 부품의 위치와 오리엔테이션이며, 이는 부품처리를 위한 로봇트 핸드의 목표위치가 된다.  $[P]$  는 역기구학 방정식(inverse kinematics)에 의하여 각 관절의 변위로 변환되어 로봇트의 서보 제어기에 입력된다.

#### 4.3 작업 시스템의 캘리브레이션 방법

작업 시스템에서의 컨베이어 벨트의 위치, 로봇트의 위치, 카메라의 위치 및 배율은 변경 될 수 있다. 로봇트와 시각 시스템의 접속을 위하여, 카메라의 스케일 인자  $a_x, a_y$  와 변환 행렬  $[T]$ 를 작업 환경의 변화에 따라 캘리브레이션하여야 한다. 캘리브레이션의 정밀도는 작업 정밀도에 크게 영향을 미치므로 그 오차가 최소화 되도록 하여야 한다.

##### 1) 스케일 인자의 캘리브레이션

영상 평면에서의 스케일 인자  $a_x, a_y$  의 값을 최소 자승 추정법 (Least Squares Estimation Method) 에 의하여 캘리브레이션 한다. 카메라 좌표계에서의 두 점  $(x_1(i), y_1(i))$ ,  $(x_2(i), y_2(i))$  사이의 실제 거리와 계산 거리 사이에는 다음

의 관계식이 성립된다.

$$d(i)^2 = a_x^2 (x_1(i) - x_2(i))^2 + a_y^2 (y_1(i) - y_2(i))^2 \quad (13)$$

$d(i)$ 는 측정된 두 점 사이의 실제 거리이다. 따라서 스케일 인자는 두 가지 데이터에 의하여 계산이 가능하다. 그러나 실제로는 카메라 평면 상의 점의 위치에 따라  $a_x$ ,  $a_y$ 는 다르며, 값의 정확도를 높이기 위하여 가능한 여러 점에 대한 측정이 바람직하다.

$n$  개의 데이터에 의한  $a_x$ ,  $a_y$  값의 계산은 다음의 행렬식에 의하여 이루어진다.

$$D = \Phi \Omega \quad (14)$$

단,

$$D = \begin{bmatrix} d(1)^2 \\ \vdots \\ d(n)^2 \end{bmatrix} \quad \Phi = \begin{bmatrix} (x_1(1) - x_2(1))^2 & (y_1(1) - y_2(1))^2 \\ \vdots & \vdots \\ (x_1(n) - x_2(n))^2 & (y_1(n) - y_2(n))^2 \end{bmatrix}$$

$$\Omega = \begin{bmatrix} a_x^2 \\ a_y^2 \end{bmatrix}$$

이 때 측정치  $D$  와 계산치  $[\Phi \Omega]$  사이의 오차, 즉

$$E = (D - \Phi \Omega)^T (D - \Phi \Omega) \quad (15)$$

를 최소화 시키는 추정 파라미터  $\Omega$  는 다음의 식과 같이 결정된다.

$$\Omega = (\Phi^T \Phi)^{-1} \Phi^T D \quad (16)$$

위의 식은 행렬  $\Phi$ 가 nonsingular 인 경우, 식(15)의 오차를 최소화하며 유일함이 증명된다. [9] 추정된 카메라의 스케일 인자는 카메라의 배율과 설치 높이가 고정된 경우 불변이다.

## 2) 변환 행렬의 캘리브레이션

식(10)에서 정의된 변환 행렬  $[T]$  는, 시각 시스템의 정보에 의하여 얻어진 부품의  $[V]$  를 부품의 처리를 위한 로봇트 핸드의  $[P]$  로 변환시킨다. 이 때 다음의 관계가 성립된다.

$$[T] = [P_i] [V_i]^{-1} \quad (17)$$

$[V_i]$  은 임의의 표본 부품에 대하여, 식(11)-(12)의 결과 얻어진 카메라 좌표계에서의 중심 위치와 오리엔테이션을 나타내는 제차 변환 행렬이다.  $[P_i]$  은  $[V_i]$  으로 인식된 표본 부품을 컨베이어 벨트로  $d$  만큼 이송시킨 후, 그 부품을 잡을 때의 로봇트 핸드의 제차 변환 행렬이다. 이 때 로봇트 핸드의 중심 위치와 부품의 중심 위치가 일치하여야 하며, 로봇트 핸드의 normal 벡터 방향과 부품 좌표계의  $x$  축, sliding 벡터 방향과 부품 좌표계의  $y$  축이 일치되도록 하여야 한다. 구동시의 로봇트의 관절 변위를, 기구학 방정식(kinematics)에 의하여 제차 변환 행렬로 변환시킨다.

따라서 하나의 표본 부품에 대한 1 회의 측정으로, 식(17)에 의하여 변환 행렬  $[T]$  를 구할 수 있다. 카메라의 설치 위치와 배율, 컨베이어 벨트의 설치 위치와 부품의 이송거리가 고정되면,  $[T]$  는 불변이다. 이송 거리  $d$  는 컨베이어 벨트의 구동 모터에 부착된 엔코더의 펄스값으로 저장된다.

## 5. 시스템 통합 및 실험

PUMA 560 로봇트 매니플레이터, 컨베이어 벨트 시스템, 광센서, 시각 시스템으로 구성된 전체 작업 시스템의 동작은, 주 컴퓨터의 작업용 언어 시스템에 의하여 프로그래밍되어 수행된다. 작업용 언어 시스템은 PUMA 560 매니플레이터를 동작시키는 명령어와 컨베이어 벨트의 동작을 위한 명령어로 구성되며 [11], 시각 시스템과의 정보 전송을 위한 명령어가 추가된다. 작업용 언어 시스템의 개발에 의하여 전체 작업 시스템은 통합되며, 작업용 명령어를 사용한 프로그래밍에 의하여 다양한 작업이 수행된다.

연속적으로 공급되는 여러가지 형태의 부품을 그 형태에

따라 로봇트가 분류시키는 작업을 프로그래밍하고 실행한다. 작업 시스템의 동작 개요는 다음과 같다. 여러가지 형태의 부품은 임의의 자세로 놓여져 컨베이어에 연속적으로 공급된다. 컨베이어 벨트는 부품을 로봇트의 작업공간 내의 처리점으로 이송시킨다. 컨베이어 벨트 위에 장착된 광센서는, 카메라의 영상 획득 영역 내에 부품이 진입하였음을 감지한다. 광센서의 감지 신호에 의하여 카메라는 부품의 영상을 획득하고 디지털라이저에 영상 신호를 전송한다. 영상 획득 시에도 컨베이어 벨트는 계속 구동되며, 광센서에서 감지된 이후 미리 지정된 거리 만큼 부품을 이송한 후 정지한다. 컨베이어 벨트 구동 중 로봇트는 초기 위치에서 작업 준비점(P1)으로 이동한다. 시각 시스템에 의하여 처리된 부품의 정보에 의하여, 부품의 처리를 위한 로봇트의 접근점(P48)과 처리점 (P49)을 계산한다. 접근점은 처리점과 오리엔테이션은 같으나, 수직 상방(기준 좌표계의 Z 축 방향)으로 100 (mm) 위치의 점으로 한다. 접근점을 정하는 이유는, 로봇트가 준비점에서 처리점으로 직접 동작할 경우 발생할 수 있는 부품과의 충돌을 방지하기 위함이다. 로봇트는 접근점과 처리점 사이에서만 직선 동작을 하며, 나머지 동작은 PTP(Point To Point)로 동작한다. 처리점에서 부품을 잡고 수직으로 들어올린 후, 부품의 형태에 따라 지정된 지점으로 부품을 옮긴다. 이러한 로봇트의 동작 중 컨베이어 벨트는 다시 구동되고, 다음 부품에 대한 검색 작업이 수행된다. 위의 과정을 반복하여 임의의 위치와 오리엔테이션을 갖는 여러가지 형태의 부품을, 부품의 형태에 따라 처리, 분류한다.

위의 작업을 수행하기 위하여, 작업용 언어 시스템에서 작성된 프로그램은 그림 5 같으며, 그림 6 은 실험중인 로봇트 작업 시스템의 모습이다. 실험 결과 로봇트 가 부품을 잡는 순간, 부품과 동작된 로봇트 핸드 사이에 오차가 발생하였다. 10 회 측정 결과 평균 위치 오차는 1.5(mm), 평균 오리엔테이션 오차는 0.9(deg) 이었다. 발생된 오차는 다음의 네가지 원

CONV ON	: 컨베이어 벨트 구동
WAIT IMPORTS = 255	: 광센서가 부품을 감지할 때까지 대기
PICTURE	: 카메라로 영상 획득, 비전처리 시각
TILL OFF	: 컨베이어 벨트의 엔코더가 펄스치로부터 1450 펄스 증가하면, 인터럽트에 의하여 컨베이어 벨트 정지
INT ENCODER ON	: 준비점으로 로봇트 이동 (PTP 동작)
TILL ENCODER = 1450	: 비전 처리과정 확인
REDO:PMOVE P1	: 0 (제리) 이전 작업 중지
CHK :IF RECOGN = 0 GOTO FIN	: 1 (처리완료) 이전 다음동작
:IF RECOGN = 1 GOTO GET	: 비전정보 전송, 접근점과 처리점 계산
GOTO CHK	: 부품 형태의 화면 표시
GET :GETFEAT	: 접근점으로 로봇트 이동
SHOW VTYPE	: 처리점으로 로봇트 이동 (직선 운동)
PMOVE P48	: 로봇트의 그리퍼 닫음
CMOVE P49	: 접근점으로 로봇트 이동 (직선 운동)
GRASP	: - 새로운 부품의 비전처리 시각 -
CMOVE P48	
CONV ON	
WAIT IMPORTS = 255	
PICTURE	
TILL ENCODER = 1450	
IF VTYPE = 1 GOSUB V1	: - 부품의 형태에 따른 분류 작업 -
IF VTYPE = 2 GOSUB V2	부품 1:서브루틴 V1, 부품 2:서브루틴 V2
IF VTYPE = 3 GOSUB V3	부품 3:서브루틴 V3
UNGRASP	: 로봇트의 그리퍼 오픈
GOTO REDO	: 작업 반복
FIN :STOP	: 작업 중지
V1 :PMOVE P2	: - 부품 1 의 처리를 위한 서브루틴 V1 -
CMOVE P3	
CMOVE P2	
RETURN	
V2 :PMOVE P4	: - 부품 2 의 처리를 위한 서브루틴 V2 -
CMOVE P5	
CMOVE P4	
RETURN	
V3 :PMOVE P6	: - 부품 3 의 처리를 위한 서브루틴 V3 -
CMOVE P7	
CMOVE P6	
RETURN	

그림 5 작업용 프로그램

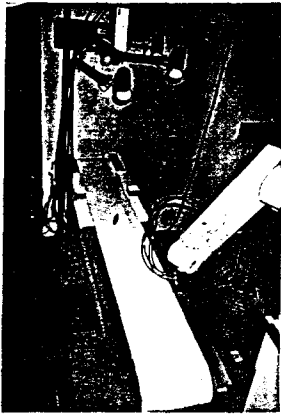


그림 6 부품 분류 실험

인으로 분석될 수 있다.

1) 이산화 오차 : 시각 시스템은 카메라의 아날로그 신호를 256 x 256 의 8 비트 데이터로 이산화하여 처리한다. 이때의 양자화(quantization) 과정은 위치 정밀도를 저하시킨다.

2) 그림자의 영향 : 입체 물체인 부품을 2차원 영상으로 처리하는 경우, 부품의 그림자에 의하여 부품의 윤곽선이 변화할 수 있다. 부품의 그림자는 조명 장치의 조도와 컨베이어 벨트 상의 위치에 따라 그 생성 정도가 다르다.

3) 컨베이어의 이송오차 : 컨베이어 벨트는 구동 모터의 축에 부착된 브레이크를 작동하여 정지시킨다. 브레이크 작동 시간이 일정한 경우, 컨베이어 벨트의 속도가 다르면, 이송 거리에 오차가 발생한다. 즉, 컨베이어 벨트의 정속 운동이 이루어지지 않는 경우, 부품은 예상 위치와 다른 위치에 정지하여 처리된다.

4) 작업 시스템의 캘리브레이션 오차 : 카메라의 스케일 인자와 변환 행렬의 캘리브레이션 과정 중 오차가 발생할 수 있다.

## 6. 결 론

컨베이어 벨트에 임의로 놓여져 흐르는, 다양한 형태의 부품을 처리하기 위한, 시각 시스템과 로봇의 접속 방법을 제시하였다. 2차원의 영상으로부터 부품의 위치와 오리엔테이션 및 형태를 찾기 위하여, 영상 영역화, 다각형 근사화 및 인식 알고리즘이 이용되었고, 이는 고속으로 (PC/386 머신에서 2 초 이내) 처리되었다. 또한 작업 시스템의 월드 모델링 및 캘리브레이션 방법이 개발되었고, 이에 의하여 시각 정보는 실시간에 로봇에 변환되어 전송되었다. 전체 시스템은 주 컴퓨터의 작업용 언어 시스템으로 통합되었고, 부품의 분류 처리 실험이 수행되었다.

실험 결과 작업 오차는 시각 시스템과의 접속 과정에서 주로 발생되었다. 이는 전체 작업 시스템의 성능이 로봇과 시각 시스템의 접속 방법에 크게 좌우됨을 증명하는 것이다. 따라서 정확도의 향상과 소요 시간의 단축을 위한, 로봇과 시각 시스템의 접속 기법에 관한 연구가 계속 진행될 것이다.

## - 참 고 문 헌 -

[1] D. H. Ballard, C. M. Brown, Computer Vision, Prentice-Hall, 1982  
 [2] T. Pavlidis, Algorithms For Graphics And Image

Processing, Computer Science Press, 1982

[3] A. K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, 1989

[4] K. Wall and P. Danielsson, "A fast sequential method for polygonal approximation of digitized curves", Computer Vision, Graphics and Image Processing, vol. 28, pp. 220-227, 1984

[5] C. T. Zhan and R. S. Rokies, "Fourier descriptors for plane close curves", IEEE Trans. Comput., vol. C-21, pp. 269-281, 1972

[6] E. Persoon and K. S. Fu, "Shape discrimination using Fourier descriptors", IEEE Trans. Syst., Man, Cybern., vol. SMC-7, pp. 170-179, 1977

[7] S. W. Holland, L. Rossal and M. R. Ward, "Consight-1: A vision-controlled robot system for transferring parts from belt conveyors", Computer Vision and Sensor Based Robots, Plenum press, 1979

[8] B. Carlisle and S. Roth, "The PUMA/VS-100 robot vision system", 1st International Conf. on Robot Vision and Sensory Control, Vol. 1, 1981

[9] K. Johan and A. B. Wittenmark, Adaptive Control, Addison-Wesley, 1989

[10] K. S. Fu, R. G. Gonzalez and C. S. G. Lee, Robotics: control, sensing, vision and intelligence, McGrawhill, 1987

[11] 이원식, "컨베이어 벨트를 포함한 로봇 작업 시스템의 구성에 관한 연구", 서울대학교 공학석사학위 논문, 1989