

머신비전을 이용한 PCB 스크린인쇄기의 정렬오차측정 및 위치보정 (2)

신동원[#]

Measurement and Correction of PCB Alignment Error for Screen Printer Using Machine Vision (2)

Dongwon Shin[#]

ABSTRACT

This paper presents the measurement and correction method of PCB alignment errors for PCB screen printer. Electronic equipment is getting smaller and yet must satisfy high performance standard. Therefore, there is a great demand for PCB with high density. However conventional PCB screen printer doesn't have enough accuracy to accommodate the demand for high-resolution circuit pattern and high-density mounting capacity of electronic chips. It is because the alignment errors of PCB occur when it is loaded to the screen printer. Therefore, this study focuses on the development of the system which is able to measure and correct alignment errors with high-accuracy. An automatic optical inspection part measures the PCB alignment errors using machine vision, and the high-accuracy 3-axis stage makes correction for these errors. This system used two CCD cameras to get images of two fiducial marks of PCB. The centers of fiducial marks are obtained by using moment, gradient method. The first method is calculating the centroid by using first moment of blob, and the latter method is calculating the center of the circle whose equation is obtained by curve-fitting the boundaries of fiducial mark. The operating system used to implement the whole set-up is carried in Window 98 (or NT) environment. Finally we implemented this system to PCB screen printer.

Key Words : Screen Printer (스크린인쇄기), PCB(인쇄회로기판), Alignment Error (정렬오차), Fiducial Mark (기준마크), Machine Vision (머신비전)

1. 서론

최근 전자부품이 집적화하고 초소형화함에 따라, 고밀도의 실장 능력을 갖춘 PCB의 수요는 날이 증가하고 있다^{1,2}. 특히 부품의 표면장착기술과 더불어 PCB의 다층화가 추세로 가고 있는 시점에서 고정밀의 PCB를 제조하는 장비에 대한

수요는 매우 높다고 볼 수 있다³. PCB에 회로패턴을 인쇄하는 스크린인쇄기는 PCB 제조 과정에서 가장 많이 사용되는 장비로서 스크린인쇄기의 고정밀화는 고정밀의 PCB를 생산함에 있어서 불가결하다⁴.

본 논문은 스크린인쇄기에서 영상처리를 이용하여 PCB의 정렬오차를 계산하고 정밀이송부를

^{*)} 2002년 11월 25일 접수
[#] 교신저자, 금오공과대학교 기계공학부
Email : shindw@kumoh.ac.kr, Tel : (054) 467-4380

이용하여 오차를 보정해주는 시스템에 대하여 기술한 것으로, 전편에는 PCB 내에 있는 2 개의 기점 마크(Fiducial mark)의 중심위치로부터 PCB 의 병진, 회전의 정렬오차를 구하는 식을 유도하였고, 미지의 초기 인자를 구하는 방법에 대하여 기술하였다. 본 논문에서는 전편의 결과를 적용한 장비의 전체 시스템 구성 및 운영프로그램 구조에 대하여 기술한다. 전체 운영프로그램의 구조는 초기화단계, 자동실행단계의 2 개의 주요 과정으로 분리하여 기능의 명확성을 도모하였다.

본 논문은 다음과 같이 구성되어 있다. 2 장에는 머신비전부와 정밀이송부로 이루어진 장비의 하드웨어에 대하여 언급한다. 3 장에서는 측정시스템의 정밀도를 좌우하는 기점마크의 중심위치를 구하는 영상처리 방법에 대하여 언급한다. 4 장에서는 시스템 프로그램의 알고리즘적인 면에서 초기화단계, 자동실행단계 등의 과정을 상세히 언급한다. 5 장은 실제로 구현한 운영프로그램의 구성에 대하여 언급한다.

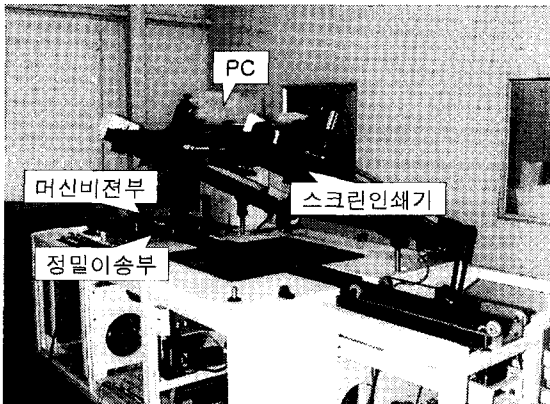


Fig. 1 Picture of screen printer with machine vision part and precision moving part

2. 장비 하드웨어

전체 장비 하드웨어는 Fig. 1 과 같이 영상을 획득하여 정렬오차를 계산하는 머신비전부와 정밀 $xy\theta$ Stage 를 이용하여 정렬오차를 보정해주는 정밀이송부로 크게 분류할 수 있다. 오차측정 및 보정이 완료된 PCB 는 스크린인쇄기 쪽으로 이동하고 스크린 인쇄 후 언로드(Unload)된다.

2.1 머신비전부

머신비전부는 동축 LED 조명, 두대의 카메라 및 Frame Grabber 보드, PCB 의 영상으로부터 정렬 오차를 계산하는 영상처리부로 구성되어 있다.

실험에서 사용한 조명은 평면형태로 배열된 LED 로부터 나온 광선이 Half Mirror 를 통과하여 카메라와 수직으로 조사되는 동축(Coaxial) 조명을 이용하였다(Fig. 2).

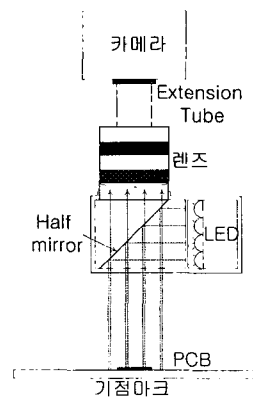


Fig. 2 Coaxial LED illumination

영상획득의 대상이 원형 동박(Copper film)일 경우 Fig. 3 과 같이 일반적인 할로겐조명을 사용하면 난반사가 일어나 배경과 대상물의 경계가 불분명하게 나타나는데 반하여 동축 LED 조명을 사용하면 경계가 확실하게 나타난다. 이러한 조명의 선택은 영상분할(Image Segmentation) 시 정확한 결과 값을 얻게 할 수 있을 뿐만 아니라, 필터링과 같은 영상의 전처리 단계를 줄일 수 있어 경계(Edge) 검출 속도를 증가시킬 수 있다⁵.

두대의 카메라는 기판 위의 두개의 기점 마크의 영상을 각각 획득하여 영상 신호 전송 규격인

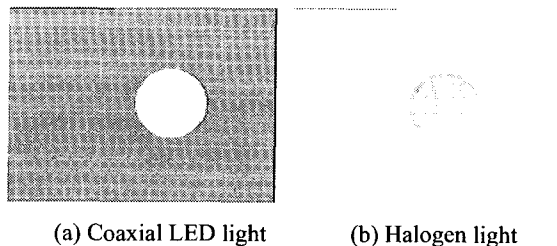


Fig. 3 Images of fiducial mark with different illuminations

RS-170 신호로 Frame grabber 로 보내준다. 카메라가 볼 수 있는 영역인 FOV(Field of View)는 7.3mm * 5.7mm 이다. CCD 카메라의 규격은 촬상 소자의 크기가 6.4*4.8 mm 이며 총 화소수가 36 만 화소이다. 카메라는 50mm 렌즈와 Extension Tube 와 함께 C-Mount 로 연결되어 있다. 두대의 카메라는 마이크로미터에 의하여 xy 방향으로 PCB 의 원하는 위치로 이동할 수 있다.

Frame Grabber 는 카메라에서 들어온 아날로그 신호를 디지털 신호로 변환하여 영상처리부로 보내준다. Frame Grabber 에서 획득된 디지털 영상의 크기는 640(H)*480(V) 픽셀(Pixel)이며 한 픽셀 당 8bit 데이터이므로 명도단계는 256 단계이다.

2.2 정밀이송부

정밀이송부는 xyθ 의 3 축 Stage, 3 개의 400 pulse/rev 의 2 상 스텝모터 및 드라이브, 모션제어기로 구성되어 있다. xyθ Stage 는 xy 이송부가 θ 이송부 위에 올려져 있는 구조로 되어 있다. 각축 양단에 Limit 센서가 부착되어 있고, Ball-Screw 로 작동되며 피치는 1mm 이다. x, y 축은 한 펄스당 2.5μm 를 움직이고, θ축은 0.573 mdeg 회전한다. PCB 의 영상신호로부터 계산된 PCB 의 정렬오차를 이용하여 정밀이송부의 3 축 xyθ Stage 는 PCB 를 정렬오차가 없는 기준위치로 움직인다. 위치보정이 완료된 PCB 는 Vacuum Plate 를 통하여 스크린인쇄기 쪽으로 이동한다.

전체적인 작동흐름을 보면, 상위제어기(PLC)에서 시작신호가 들어오면 측정/보정시스템에서 머신비전부는 PCB 의 정렬위치오차를 측정하고 정밀이송부는 보정을 행한다. 측정 및 보정이 완료되면 상위제어기로 종료신호를 알린다. 종료신호를 받은 상위제어기는 PCB 의 Unload 를 실행하고 Unload 완료신호를 내보낸다. Unload 완료신호를 받은 측정/보정시스템은 xyθ Stage 를 원위치시킨다. 위치측정/보정 시에 에러가 발생하면 시스템은 상위제어기로 에러가 발생했음을 알린다. 이와 같이 측정/보정시스템에는 PLC 와 같은 상위제어기와 신호를 주고 받기 위해 입출력장치가 이용된다.

3. 원의 중심을 구하는 알고리즘

전편에서 기술한 PCB 의 정렬오차를 계산하는 경우 기점마크의 위치정보를 이용하는데, 기점마

크의 위치정보라 함은 기점마크의 중심의 위치를 의미한다. 따라서 기점마크의 중심좌표를 얼마만큼 정확히 구하는가의 여부가 측정시스템의 정밀도를 좌우하므로 중심위치를 구하는 과정은 전체 알고리즘 면에서 매우 중요한 부분이라고 볼 수 있다. 일반적인 기점마크는 원형으로 형성되기 때문에 본 논문에서는 원의 중심을 구하는 방법을 2 가지 측면에서 접근하였다. 첫번째는 원의 1 차 모멘트를 이용하여 원의 무게중심을 구하는 방법이고⁶ 두번째는 원의 경계치를 구한 후 원의 중심을 최소자승법(Least Square Method)를⁷ 이용하여 구하는 방법이다. 원의 경계치를 구하는 경우 1 차 미분법을⁸ 이용한 알고리즘을 적용하여 경계를 구하였다. 카메라로부터 입력된 영상에서 영상처리 대상인 원을 배경과 분할(Segmentation)하기 전에 중간값 필터를 이용하여 획득한 영상의 잡음을 제거하였다⁶. 중간값 필터는 데이터 열을 크기 순으로 정렬하여 중간위치에 오는 값을 해당 픽셀의 밝기 값으로 선정하는 것으로 영상으로부터 충격(Impulse) 잡음을 제거하는데 효과적인 필터이다.

3.1 모멘트법에 의한 원의 중심 구하기

모멘트법을 적용하기 전에 주어진 영상을 이진화(Thresholding)하여 영상의 원 부분과 배경 부분을 분할시킨다. 여기에서 영상을 이진화하기 위해서는 일정한 기준값을 선정하여야 하는데 이 값을 임계값(Threshold)이라 한다. 본 연구에서는 임계값을 자동으로 설정하는 방법 중에서 분산비를 최대로 하는 방법 (otsu's method)을⁹ 이용하여 임계값을 구하였다.

이 방법은 영상을 2 개의 클래스 S_1, S_2 로 분할하는 경우, 다음에 표시된 분리도(separability) $\eta(T)$ 가 최대가 되도록 임계값 T 를 설정하는 것이다. 이 방법은 쌍봉(Bimodal)선이 보이지 않는 경우에도 분할할 수 있는 장점이 있다

$$\eta(T) = \left[\frac{\sigma_B^2(T)}{\sigma_W^2(T)} \right] \quad (1)$$

여기서 클래스 S_1, S_2 에 대하여, $\sigma_B^2(T)$ 는 클래스간 분산 (between-class variance)이고 $\sigma_W^2(T)$ 는 클래스내 분산 (within-class variance)으로 식(2), (3) 와 같다. 영상전체의 분산은 $\sigma_T^2 = \sigma_W^2 + \sigma_B^2$ 과 같

다.

$$\sigma_w^2 = \omega_1 \sigma_1^2 + \omega_2 \sigma_2^2$$

$$= \frac{1}{N} \left\{ \sum_{i \in S_1} (i - \mu_1)^2 n_i + \sum_{i \in S_2} (i - \mu_2)^2 n_i \right\} \quad (2)$$

$$\sigma_B^2 = \omega_1 (\mu_1 - \mu_T)^2 + \omega_2 (\mu_2 - \mu_T)^2$$

$$= \frac{1}{N} \left\{ \sum_{i \in S_1} (\mu_1 - \mu_T)^2 n_i + \sum_{i \in S_2} (\mu_2 - \mu_T)^2 n_i \right\} \quad (3)$$

여기서,

- ω_1, ω_2 : 클래스 S_1, S_2 의 발생 확률
- μ_1, μ_2 : 클래스 S_1, S_2 의 픽셀 명도의 평균값
- σ_1, σ_2 : 클래스 S_1, S_2 의 픽셀 명도의 분산
- i : 픽셀의 명도
- n_i : i 의 명도를 가지는 픽셀의 개수
- N : 전체 픽셀의 개수

분산비가 최대가 되는 임계값을 이용하여 영상을 이진화하면 8-연결로 연결된 블럽(blob)을 찾게 된다. 영상 중에서 가장 많은 픽셀을 가진 블럽을 기점마크의 블럽으로 선정한다. 기점마크의 중심점은 식(4)과 같이 블럽의 1 차 모멘트를 이용한 무게 중심으로부터 구한다.

$$\bar{x} = \frac{\sum_i^N x_i}{N}, \bar{y} = \frac{\sum_i^N y_i}{N} \quad (4)$$

여기서, x_i 는 픽셀의 x 좌표, y_i 는 픽셀의 y 좌표, N 은 총픽셀수이다.

이와 같이 구한 임계값을 이용하여 영상을 이진화하기 전에 영상 내에 존재하는 임펄스 잡음을 없애는 것이 유리하다. 본 연구에서는 이를 위하여 중간값 필터를 사용하였다. 원래 영상이 Fig. 4와 같을 때, 중간값 필터를 사용하기 전과 후의 한 지점에서(Fig 4 에서 실선) 명도의 변화를 관찰하면 Fig. 5, 6 과 같다. 그림에서와 같이 중간값 필터를 사용하면 잡음 값이 많이 줄어들어 중간값 필터를 사용하지 않은 경우보다 임계값의 설정이 보다 정확해질 수 있어, 영상분할이 정확해진다.

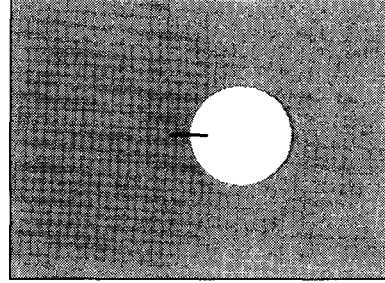


Fig. 4 The original image of fiducial mark of PCB

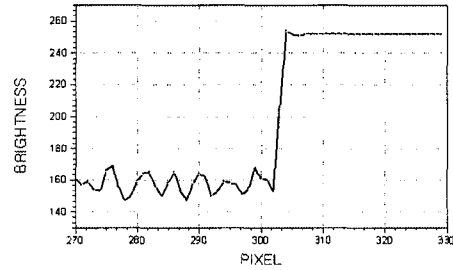


Fig. 5 The change of gray level without median filtering

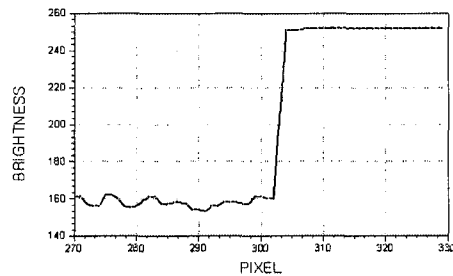


Fig. 6 The change of gray level with median filtering

3.2 일차 미분법에 의한 원의 중심 구하기

기점마크의 경계를 구하기 위해 일차 미분법(Gradient Method)을 사용하였다. 일차 미분필터를 사용하면 밝기의 기울기 값을 구하여 급격히 변화하는 부분을 경계로 검출하는 데 사용할 수 있다. 본 연구에는 일차 미분필터 중에서 다음과 같은 컨볼루션(Convolution) 마스크를 가지는 Sobel 필터를 사용하였다⁶.

$$G_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}; \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (5)$$

경계가 되는 최종 기울기 영상은 G_x 로 필터링 된 영상과 G_y 로 필터링 된 영상을 절대치 화한 후 가산하여 구한다.

원래 영상의 Sobel 필터링 한 영상은 Fig. 7 과 같다. 중간값 필터의 일차미분에 대한 효과를 관찰하기 위하여 중간값 필터를 적용하였을 경우와 적용하지 않았을 경우, Fig 7의 실선부분에서의 명도 변화를 관찰하면, Fig. 8, 9 와 같다. 중간값 필터를 적용하지 않은 경우, 적용한 경우보다 일차미분값이 두 배 정도 커짐을 알 수 있다. 잡음의 일차미분 시 그 값이 커져서 실제 경계가 아님에도 경계인 것처럼 잘못 판단될 수가 있다. 이러한 경계정보를 가지고 최소자승법에 의해 원의 중심을 구하게 되면 잘못된 중심을 찾을 수 있다. 따라서 일차미분을 하기 전에 중간값 필터를 적용하는 것이 정확한 결과를 가져 올 수 있다.

실제로 중간값 필터를 하였더라도 영상의 모든 잡음을 필터링할 수 없다. 중간값 필터링을 한 후 일차미분을 한 경우, 경계영역이 아닌 잡음의 영역에서 일차미분값이 크게 나올 수 있다. 따라서, 일차미분된 영상을 Otsu's method 를 사용하여 임계값을 선정하고 임계값보다 낮은 픽셀은 경계 영역에서 제외시킨다. 이렇게 구한 경계영역의 영

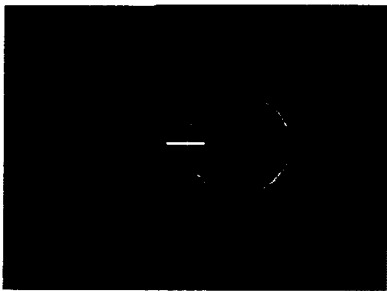


Fig. 7 The edge image of fiducial mark after sobel filtering

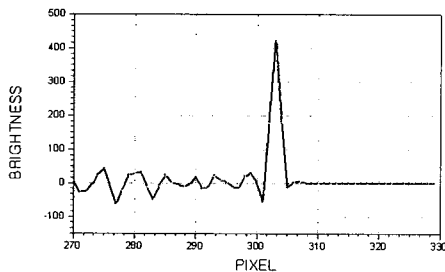


Fig. 8 The change of gray level without median filtering

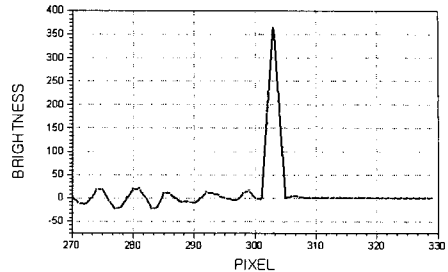


Fig. 8 The change of gray level with median filtering

상에 서브픽셀(Subpixel) 알고리즘을 ¹⁰ 적용하여 정확한 경계를 선정한다. 이러한 경계값을 가지고 최소자승법을 이용한 곡선맞춤(Curve-fitting)을 함으로써 원의 중심을 구할 수 있다.

3.3 최소자승법을 이용한 원의 중심 찾기

일차 미분법을 통하여 구한 경계를 가지고 원의 반지름 r 및 중심 (x_0, y_0) 을 구하기 위하여 다음과 같은 원의 방정식을 가정한다.

$$(x_i - x_0)^2 + (y_i - y_0)^2 = r^2, i = 1, \dots, n \quad (6)$$

여기서 n 은 경계를 구성하는 Pixel 의 개수이고, (x_i, y_i) 은 경계픽셀의 x, y 좌표값이다.

식(6)의 우변을 좌변으로 넘기고 오차의 자승을 다음과 같이 정의한다 ⁵.

$$E = \sum_{i=1}^n [(x_i - x_0)^2 + (y_i - y_0)^2 - r^2]^2 \quad (7)$$

$$= \sum_i (x_i^2 - 2x_i x_0 + y_i^2 - 2y_i y_0 + z)^2$$

where, $z = x_0^2 + y_0^2 - r^2$

이어서 오차를 최소화하는 최소자승법을 사용하여 식(8)을 만들 수 있고, 이로부터 원의 반지름 r 및 중심 (x_0, y_0) 을 구할 수 있다.

$$\frac{\partial E}{\partial x_0} = \sum_{i=1}^n (x_i^2 - 2x_i x_0 + y_i^2 - 2y_i y_0 + z)(-2x_i) = 0$$

$$\frac{\partial E}{\partial y_0} = \sum_{i=1}^n (x_i^2 - 2x_i x_0 + y_i^2 - 2y_i y_0 + z)(-2y_i) = 0 \quad (8)$$

$$\frac{\partial E}{\partial z_0} = \sum_{i=1}^n (x_i^2 - 2x_i x_0 + y_i^2 - 2y_i y_0 + z) = 0$$

이와 같이 경계값을 이용한 최소자승법을 통하여 원의 중심을 구하는 경우를 모멘트법을 이용하여 원의 중심을 구하는 경우와 비교해 본다.

모멘트법은 무게중심을 구할 경우 원의 내부에 해당하는 다수의 픽셀들을 대상으로 계산하므로 잡음에 그다지 민감하지 않으면서도 비교적 정확한 결과를 얻을 수 있다. 한편, 경계값을 이용한 최소자승법은 영상에 잡음이 많은 경우(중간값 필터를 사용해도 모든 잡음이 없어지지 않는) 실제 경계가 아님에도 불구하고 경계로 인식되어 잘못된 중심을 구할 수 있는 단점이 있다. 반면 서브픽셀 알고리즘을 이용하여 경계값을 구하므로 경계의 정확한 값을 구할 수 있는 장점이 있다. 결론적으로, 잡음이 많은 영상에 대하여는 모멘트법이 좋은 결과를, 잡음이 없는 깨끗한 영상에 대하여는 경계값을 이용한 최소자승법이 좋은 결과를 가져올 수 있다.

4. 시스템 프로그램 Flow Chart

시스템 운용프로그램은 크게 3 가지로 구분된다. 초기인자를 구하는 초기화단계과 측정/보정을 자동으로 행하는 자동실행단계와 자동복구과 같은 부가적 단계이다. 초기화단계는 Fig. 10 과 같이 인쇄할 PCB 를 입력한 후 자동실행과정에 필요한 여러 초기 인자 값을 구하기 위한 일련의 과정이다. 그리고 자동 실행 단계는 Fig. 11 와 같이 초기화단계에서 계산된 값을 토대로 오차를 측정 및 보정하여 지속적인 검사를 수행하게 하는 단계이다. 부가적단계는 자동복구라던지 파일관리와 같은 일을 수행하는 단계이다.

4.1 초기화 단계

전편에 기술한 바와 같이 의도적으로 $xy\theta$ stage 에 병진변위 $\Delta_s=[\Delta x_s, \Delta y_s]^T$ 를 가하는 경우, 기점마크 #1 의 변위량이 $[\Delta x_1^d, \Delta y_1^d]^T$, 기점마크 #2 의 변위량이 $[\Delta x_2^d, \Delta y_2^d]^T$ 으로 측정된다면, 카메라의 기울어진 각도 δ_1, δ_2 는 다음과 같이 구할 수 있다(Part 1 논문 참조).

$$\delta_1 = \tan^{-1}(\Delta y_s / \Delta x_s) - \tan^{-1}(\Delta y_1^d / \Delta x_1^d) \quad (9)$$

$$\delta_2 = \tan^{-1}(\Delta y_s / \Delta x_s) - \tan^{-1}(\Delta y_2^d / \Delta x_2^d)$$

또한 의도적으로 $xy\theta$ stage 에 병진변위 $\Delta\theta_s$ 를 가하는 경우, 기점마크 #1 의 변위량이 $[\Delta x_1^d, \Delta y_1^d]^T$, 기점마크 #2 의 변위량이 $[\Delta x_2^d, \Delta y_2^d]^T$ 으로 측정된다면, 기준위치에 있는 기점마크 #2 의 위치 $P_2=[x_2, y_2]^T$ 와 $xy\theta$ Stage 의 회전중심 $C=[C_x, C_y]^T$ 을 다음과 같이 구할 수 있다 (Part 1 논문 참조).

$$\begin{bmatrix} C_x \\ C_y \end{bmatrix} = \rho \cdot \begin{bmatrix} 1 - \cos \Delta\theta_s & \sin \Delta\theta_s \\ -\sin \Delta\theta_s & 1 - \cos \Delta\theta_s \end{bmatrix} Rot(\delta_1) \begin{bmatrix} \Delta x_1^d \\ \Delta y_1^d \end{bmatrix}$$

$$\begin{bmatrix} x_2 \\ y_2 \end{bmatrix} = \begin{bmatrix} C_x \\ C_y \end{bmatrix} - \begin{bmatrix} 1 - \cos \Delta\theta_s & \sin \Delta\theta_s \\ -\sin \Delta\theta_s & 1 - \cos \Delta\theta_s \end{bmatrix} Rot(\delta_2) \begin{bmatrix} \Delta x_2^d \\ \Delta y_2^d \end{bmatrix}$$

where, $\rho = \frac{1}{2 - 2 \cos \Delta\theta_s}$

(10)

초기화 단계는 자동검사를 위해 시스템의 미지의 초기 인자 값을 구하고 시스템을 자동실행하기 위한 일련의 준비과정으로 이를 자세히 기술하면 다음과 같다.

1. PCB 를 입력한다.
2. 2 대의 카메라를 PCB 의 기점마크가 있는 곳으로 이동시키고 두개의 기점마크의 중심 위치를 기점마크의 기준위치로 설정하고 이를 저장한다. 또한 이 때의 $xy\theta$ Stage 의 위치를 $xy\theta$ Stage 의 기준위치로 설정한다.
3. 카메라에서 나오는 영상이 실제로 크기가 얼마인지 알아내기 위해서 카메라 교정 (Calibration)을 행한다. 즉, 카메라 2 대에 대한 x 축, y 축 각각의 교정 데이터를 얻는다.
4. PCB 가 올려진 $xy\theta$ Stage 를 $\Delta x_s, \Delta y_s$ 이동하고 식(9)를 이용하여 기준좌표계에 대한 카메라의 기울어진 각도를 알아낸다.
5. $xy\theta$ Stage 를 $\Delta\theta_s$ 만큼 회전시키고 식(10)을 이용하여 $xy\theta$ Stage 의 회전중심 위치 (C_x, C_y) 과 2 번 카메라좌표계의 기점마크의 기준위치 (x_2, y_2) 를 알아낸다.
6. PCB 를 $xy\theta$ Stage 에서 출력시키고 인쇄기 쪽으로 입력하여 스크린 인쇄를 행하고 회로패턴이 정확한 위치에 인쇄되었는지를 확인한다. 인쇄가 제대로 되었다면 다음 과정으로 넘어가고, 그렇지 않다면 회로패턴의 벗어난

양을 바탕으로 Screen Stage 의 미세 위치조정 마이크로미터를 움직인다. 다시 PCB 를 입력하여 오차를 측정하여 오차만큼의 위치를 보정하여 스크린 인쇄를 행한다. Screen Stage 를 이미 조정 한 후이므로 전보다 인쇄정확도가 높아진다. 인쇄정확도가 만족될 때까지 상기의 순서를 반복한다. 유의할 점은 Screen Stage 를 움직이는 대신 xyθ Stage 를 이동해도 같은 효과가 있다는 점이다. 단, 대변위의 이동시엔 Screen Stage 를 이용하는 편이 좋다.

7. 상기의 과정을 거치면 초기화가 완료되고 PCB 를 출력시킨다.

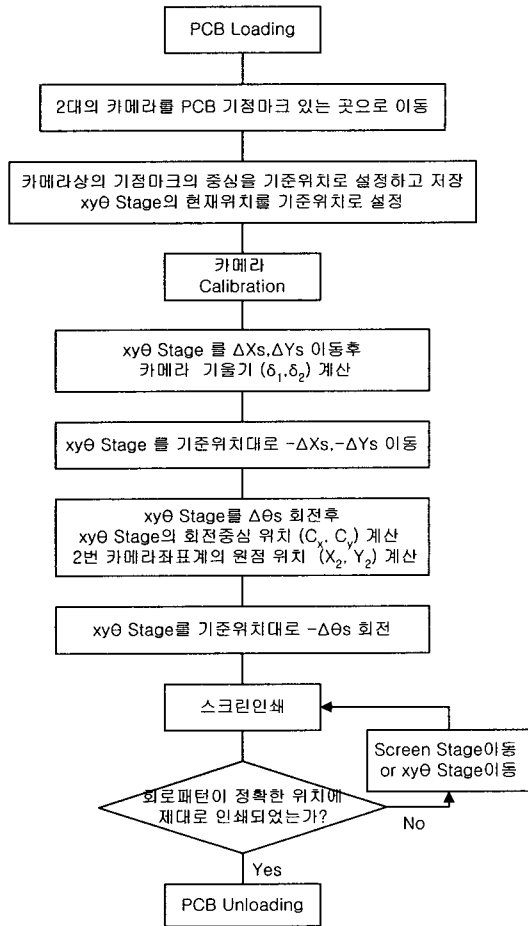


Fig. 10 Initializing stage of operating system

4.2 자동실행 단계

초기화과정이 완료되면 장비가 자동실행을 할 수 있는 여건이 마련된 것이다. 전편에 기술한 바와 같이, 영상신호로 들어온 기점마크의 이동량으로부터 PCB 의 정렬오차인 회전변위와 병진변위를 계산할 수 있다. 자동실행과정은 PCB 가 입력되면 정렬오차를 자동으로 계산하고 이를 기구적으로 보정해주는 일련의 과정이다. 이를 자세히 기술하면 다음과 같다.

1. 스크린 인쇄할 PCB 가 새로이 입력되면 상위 제어기는 자동실행시작을 알리는 START 신호를 주고 측정/보정 시스템은 자동실행을 하게 된다.
2. PCB 의 두개의 기점마크를 영상처리하여 PCB 의 정렬오차($\Delta x, \Delta y, \Delta \theta$)를 계산한다.
3. 계산된 정렬오차값을 이용하여 xyθ Stage 를 움직이는 위치보정을 행한다.

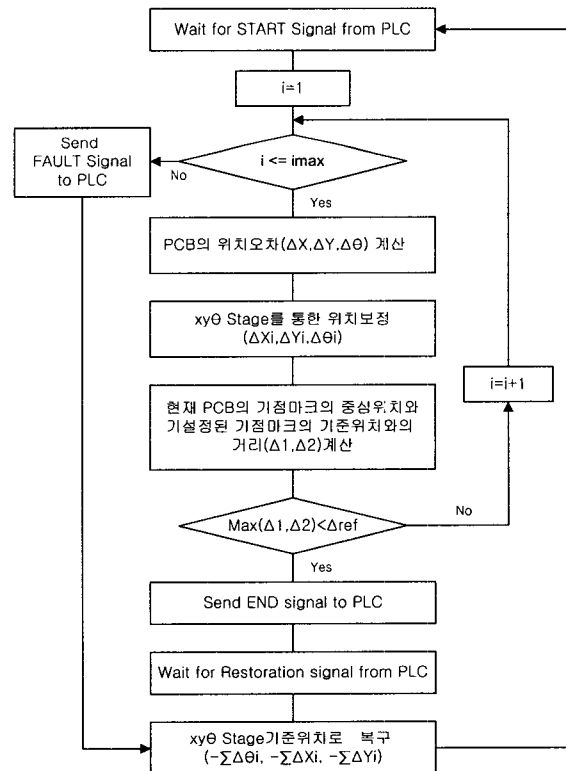


Fig. 11 Automatic execution stage of operating system

4. 보정된 PCB 의 두개의 기점마크의 영상을 다시 읽어 들여 중심위치를 구한 후 초기화단계에서 기설정된 기점마크의 기준위치와의 거리오차를 계산한다. 기점마크는 2 개이므로 거리도 $\Delta 1, \Delta 2$ 로 2 개가 나온다.
5. $\Delta 1, \Delta 2$ 중 큰 값이 사용자가 정한 거리오차한계값(Δ_{ref})보다 작으면 상위제어기에 위치보정이 성공적으로 완료됐다는 END 신호를 보내고 조건이 성립하지 않으면 다시 PCB 의 정렬오차값을 계산한다. 이러한 반복과정이 사용자가 정한 최대반복수(i_{max}) 커지면 위치보정실패로 인식하고 상위제어기쪽으로 FAULT 신호를 준다. 실제로 위치측정/보정의 여러 번의 반복실행으로 안되는 경우는 PCB 기점마크 자체가 불량제작되었거나 파손에 의해서 PCB 에 변형이 갔을 경우이다.
6. 위치보정이 성공하면 상위제어기가 PCB 기판을 스크린 인쇄기로 옮기고 xy θ Stage 에서 기판이 제거되었다고 신호를 보내게 되고 기판이 제거가 되면 xy θ Stage 를 기준위치까지 복구시킨다.
7. FAULT 일 경우 사용자가 PCB 를 제거하고 확인 버튼을 누르면 xy θ Stage 가 기준 위치로 복귀한다.
8. 한 사이클이 끝나면 본 시스템은 다시 상위제어기로부터 START 신호가 올 때까지 기다린다.

4.3 부가적 단계

전체 운영시스템에는 초기화 단계 및 자동실행 단계 외에 자료 관리 과정과 xy θ Stage 기준위치 재 복구과정 등의 부가적인 과정이 포함되어야 한다.

자료 관리과정에서는 초기화 단계에서 계산된 카메라의 교정 데이터를 저장하는 파일(Calibration.dat), 기준이 되는 피두셜 마크의 중심 위치와 검사 방법을 저장하는 파일(Fid_Position.dat), 하루에 실행한 PCB 에 대한 Log 파일을 생성한다. Calibration.dat 와 Fid_Position.dat 에 저장된 정보는 컴퓨터를 재부팅하거나 모터가 제대로 움직이지 않았을 경우 사용된다. Log 파일은 기준마크의 타입이 일반적인 마크인지 혹은 홀인지의 여부, 중심찾는 방법(Moment, Gradient), 위치보정에 성공한 PCB 의 최종 거리 오차값, 반복 실행수, 위치 보정에 실패한

PCB 의 최종거리 오차값 등의 PCB 검사방법 및 결과의 특성값들이 저장된다. Log 파일을 분석하면 하루 검사량과 불량 횟수, 오차 정도 등을 분석할 수 있다.

xy θ Stage 기준위치 재복구 과정은 자동실행시 정전과 같은 불의의 사고를 당했을 경우나 스텝모터가 가끔 펄스를 손실함으로 xy θ Stage 가 기준위치로 복구하지 못하는 경우에 사용한다. xy θ Stage 기준위치 재복구는 저장된 기점마크의 기준위치로 새로 들어온 기판의 기점마크를 돌린 후, 기점마크가 저장된 위치에 오면 그 점을 기준위치로 다시 선정하고, 그 위치를 xy θ Stage 기준위치로 정한다. 기준 위치 재복구 과정은 새로운 종류의 기판을 검사 할 때를 제외하고 컴퓨터를 부팅할 때마다 한번씩 행해주는 것이 좋은 방법이라고 할 수 있겠다.

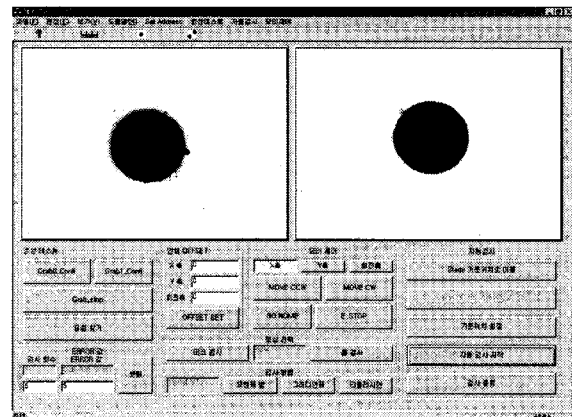


Fig. 12 The GUI interface of operating system

5. 운영 프로그램

본 시스템을 운영하는 프로그램은 Visual C++ 6.0 으로 작성되었다. Fig. 12 와 같이 사용자 인터페이스의 상단부분에 2 개의 카메라에서 들어오는 구멍형태의 기점 마크의 형상을 보여주고 있으며, 기점마크 중심의 십자표시는 기점 마크의 중심의 위치를 나타낸다. 프로그램기능 시험부분이 있어, 영상획득 및 중심 찾기 버튼을 눌러 기점 마크의 형상이 제대로 영상으로 획득되는지, 기점마크의 중심을 제대로 찾는지를 확인 할 수 있다. 전장에 기술한 바와 같이 기점마크의 중심 찾기 방법에 모멘트법과 일차미분법을 선택할 수 있으며, 기점

마크도 구멍과 마크 중 하나를 선택할 수 있다 (Fig. 12 는 구멍의 경우). 프로그램의 중앙 부분은 3 개의 모터를 움직일 수 있는 모터제어 부분이 있다. 우측 부분은 초기화 및 자동검사 기능을 할 수 있는 버튼으로 구성되어 있다. 카메라 교정, 기점마크의 영상저장, 기타 설치된 보드의 주소 설정 등 기타기능은 풀다운 메뉴로 제작하였다.

6. 결론

본 연구는 스크린인쇄기에서 영상처리를 이용하여 PCB 의 정렬오차를 계산하고 정밀이송부를 이용하여 오차를 보정해주는 시스템에 대하여 연구하였고, 이를 실제 시스템으로 구축하여 현장에서 적용하였다.

머신비전부에 동축 LED 조명을 사용하여 기점마크의 잡음을 줄여 영상분할을 원활히 하였다. 측정시스템의 정밀도를 좌우하는 기점마크의 중심 위치를 구하는 방법으로 모멘트법과 일차미분을 이용한 경계검출방법을 이용하였다. 모멘트법 시에는 자동입계방법을 이용하여 영상을 이진화한 후 원의 중심을 구하였고, 경계검출방법에서는 서브픽셀 단위로 구한 경계를 가지고 최소자승법을 이용하여 원의 중심을 구하였다. 전체 운영프로그램을 초기화 단계, 자동실행단계의 중요한 과정으로 분리하여 기능의 명확성을 도모하였다. 마지막으로 C++로 운영프로그램을 구축하여 실제 생산 라인에 적용하였다.

본 연구는 PCB 의 다른 생산 및 검사공정에 적용가능하며 향후에는 노광기와 같은 고정밀 PCB 생산장비에 적용할 예정이다.

참고문헌

1. Teoh, E. K., Mital, D. P., Lee, B. W., Wee, L. K., "Automated visual inspection of surface mount PCBs," Proceeding of IECON'90(16th Annual Conference of IEEE Industrial Electronics Society), pp. 576-580, 1990.
2. Ahmadi, R. H., Kouvelis, P., "Design of electronic assembly lines: An analytical framework and its application," European Journal of Operational Research, Vol. 115, No. 1, pp. 113-136, 1999.
3. Alvi, Mohammed I., Giachetti, Ronald E., "An object-oriented information model for manufacturability analysis of printed circuit board," COMPUTERS IN INDUSTRY, Vol. 45, No. 2, pp. 177-196, 2001.
4. "Applied Technology: Screen Printer Helps Improve Production Consistency and Quality Goals," Electronic packaging and production, Vol. 39, No. 7, pp. 55-56, 1999.
5. Shin, Dongwon, Kim, Chon-Whan, "Measurement and Correction of PCB Alignment Error for Screen Printer Using Machine Vision," Proc. of KSPE Spring Annual Meeting 2000, pp. 347-350, 2000.
6. Gonzalez, R. C., Wood, R. E., Digital Image Processing, Addison Wesley.
7. Park, Jae-Sung, Park, Heui-Jae, Kim, Ku-Young, "Development of Managing System of Vision Probe for CMM," Proc. of KSPE Autumn Annual Meeting 1996, pp. 501-505, 1996.
8. NALWA, VISHVJIT S., BINFORD, THOMAS O., "On Detecting Edges," IEEE Trans. Pattern Anal. Machine Intell., Vol. PAMI-8, No. 6, pp. 699-714, Nov. 1986.
9. Otsu, N., "A Threshold Selection Method from Gray Level Histogram," IEEE Transactions on Syst., Man and Cyber., Vol. SMC-9, No. 1, pp. 62-66, 1979.
10. Parker, J. R., Algorithms for Image Processing and Computer Vision, John Wiley & Sons, 1997.