

인쇄회로기판 검사기의 경로 계획 알고리즘

이수길*, 김화중*, 박태형**
 *충북대학교 대학원 제어계측공학과,
 **충북대학교 전기전자 및 컴퓨터 공학부

A Path Optimization Algorithm of PCB Inspection Machine

Soo Gil Lee*, Hwa Jung Kim*, Tae Hyoung Park**
 *Dept. of Control & Instrumentation Eng. Graduate School, Chungbuk National Univ.,
 **School of Electrical & Computer Eng. Chungbuk National Univ.

Abstract - SMT형 인쇄회로기판 조립라인에서 SMD의 조립상태를 검사하는 검사기를 위한 경로계획 알고리즘을 제안한다. 경로계획 알고리즘은 FOV 생성 최적화와 생성된 FOV의 순서 최적화에 의하여, 검사기의 전체 검사 시간의 단축을 목표로 한다.

본 논문에서는 검사기 경로계획 문제를 수학적으로 모델링하고, 전체 검사 단계를 FOV 생성 단계와 순서결정 단계의 계층적 구조로 구성한다. 각 단계의 알고리즘은 FOV 생성 알고리즘과 TSP 알고리즘을 적용하여 구현한다. 제시된 알고리즘을 실제 검사장비에 적용하여 시뮬레이션 하고, 그 유용성을 검증한다.

1. 서 론

최근 전자제품의 소형화, 다기능화, 그리고 고성능화를 추구함에 따라, SMD(surface mount device) 제작기술과 SMT(surface mount technology) 기술을 이용한 SMT형 인쇄회로기판(PCB; printed circuit board) 조립라인의 생산성이 중요한 요소로 인식되고 있다[1]. 이와 같은 부품제작과 실장기술의 발전으로 PCB가 고밀도화, 고집적화 함에 따라, PCB의 생산성과 함께 생산된 PCB에 신뢰성을 부여하는 검사기술에 관한 연구가 중요시되고 있다. 특히, 그림 1과 같은 비전시스템(vision system)을 이용하는 AOI(automate optical inspection)를 중심으로 연구와 개발되어 사용되는 검사장비가 증가하는 추세이다.

검사기술에 관한 연구 중에서 검사 알고리즘에 관한 연구는 폭넓게 이루어지고 있지만, FOV 생성 알고리즘에 관한 연구는 상대적으로 적다. FOV 생성 알고리즘은 신경 회로망을 이용한 FOV 생성 알고리즘이 나와있다[2]. FOV 생성 알고리즘은 검사영역을 포함하는 FOV의 총수를 최소화하여 검사장비의 전체 검사시간의 단축을 목표로 한다.

본 논문에서는 SMT형 인쇄회로기판 조립라인에서 SMD의 조립상태를 검사하는 AOI형 검사장비의 경로계획 알고리즘을 제안한다. 우선 검사장비 경로계획 문제를 수학적으로 모델링하고, 검사장비의 경로계획 알고리즘은 FOV(field of view) 생성 최적화와 생성된 FOV의 순서 최적화의 계층적 구조로 구성한다. 그리고, 각 단계

를 FOV 생성 알고리즘과 TSP[3][4] 알고리즘을 적용하여 구현한다. 제시된 알고리즘을 실제 AOI 검사장비에 적용하여 시뮬레이션하고, 그 유용성을 검증한다.

2. 문제 구성

AOI 검사장비의 경로계획 문제는 전체 검사시간을 최소화하는 문제로 식 (1)과 같이 나타낸다. 전체 검사시간 $T_{inspection}$ 은 FOV의 수에 따른 영상획득 시간 T_{image} 와 FOV의 검사경로에 따른 이동시간 T_{move} 로 나눌 수 있다. 따라서 검사 장비의 경로계획 문제는 영상획득 시간을 최소화하는 문제와 검사경로 이동시간을 최소화하는 문제로 나누어진다.

$$\min T_{inspection} = \min (T_{image} + T_{move}) \quad (1)$$

하나의 FOV에서 검사영상을 획득하는데 일정한 시간이 소요되므로, 영상획득 시간을 최소화하는 문제는 생성된 총 FOV의 수를 최소화하는 문제로 요약된다. 그리고 총 이동시간을 최소화하는 문제는, 주로 FOV 생성 알고리즘에 의해 생성된 FOV의 수에 의존하지만, 독립적으로 FOV 방문시간을 최소화하는 TSP 문제로 표현될 수 있다. 따라서 식 (1)에서의 검사장비의 총 검사시간을 최소화하는 문제는, 식 (2)와 같이 FOV의 수를 최소화하는 문제와 TSP 문제가 결합되어 표현된다.

$$\min (T_{image} + T_{move}) \leq (\min T_{image}) + (\min T_{move}) \quad (2)$$

검사장비의 카메라에서 하나의 FOV에서 영상을 획득하는 일정한 시간을 T_c 라 하면, 식(1)에서의 총 영상획득 시간은 다음과 같이 표현된다.

$$\min T_{image} = \min \sum_{k=1}^m T_c F_k \quad (3)$$

그리고 FOV의 검사경로에 따른 이동시간 T_{move} 를 수식화하면 식 (4)와 같다[4].

$$\min T_{move} = \min \sum_{i=0}^m \sum_{j=0}^m c_{ij} x_{ij} \quad (4)$$

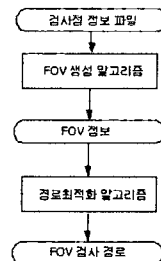


그림 2. 경로계획 문제의 순서도
 Fig. 2. Flowchart of Path Optimization



그림 1. 삼성테크윈 AI400[5]
 Fig. 1. Samsung Techwin AI400[5]

따라서, 식 (3)과 식 (4)를 이용하면 식 (2)는 다음과 같이 표현될 수 있다.

$$(\min T_{image}) + (\min T_{move}) \quad (5)$$

$$= \min \sum_{k=1}^m T_c F_k + \min \sum_{i=0}^m \sum_{j=0}^m c_{ij} x_{ij}$$

본 논문에서는 식 (5), 그림 2와 같이, 경로계획 문제를 FOV 생성 단계와 생성된 FOV의 검사순서를 결정하는 검사 순서결정 단계의 계층적 구조로 구성한다. FOV의 생성 단계에서는 생성된 총 FOV의 수를 최소화하고, FOV 검사순서를 결정하는 단계에서는 검사장비의 이동 시간을 최소화하는 검사순서를 결정한다. 그림 2는 경로 계획 문제의 순서도를 나타낸다.

전체 검사점과 생성된 FOV를 식 (6)과 같이 정의한다. n 은 전체 검사점의 수이고, m 은 생성된 FOV의 수이며 $1 \leq m \leq n$ 의 범위를 갖는다.

$$\text{검사점: } W = \{ w_i \mid w_1, w_2, w_3, \dots, w_n \} \quad (6)$$

$$\text{FOV: } F = \{ f_k \mid f_1, f_2, f_3, \dots, f_m \}$$

FOV를 생성하는 문제는 검사점 W 의 영역에서 FOV F 의 영역으로 매핑하는 할당문제의 일종이다.

$$y_{ik}: w \mapsto f_k \quad (7)$$

$$\sum_k y_{ik} = 1, \forall i \quad (8)$$

$$\sum_k y_{ik} \geq 1, \forall k \quad (9)$$

$$(X_{\min, f_k}) y_{ik} \leq (X_{w_i}) y_{ik} \leq (X_{\max, f_k}) y_{ik} \quad (10)$$

$$(Y_{\min, f_k}) y_{ik} \leq (Y_{w_i}) y_{ik} \leq (Y_{\max, f_k}) y_{ik}$$

식 (8)은 모든 검사점은 단 하나의 FOV를 가지고 있어야 한다는 조건을 나타내며, 식 (9)는 모든 FOV는 하나 이상의 검사점을 포함하고 있어야 한다는 조건을 나타낸다. 그리고 식 (10)은 검사점이 FOV의 영역안에 포함되는지 여부를 나타내는 조건이다.

m 개의 FOV에 대한 검사장비의 작업경로는, 최초의 대기점 $F_0 \in R^2$ 에서 출발하여 모든 FOV를 한변씩 방문하고 다시 대기점으로 돌아오는 해밀토니안 경로(Hamiltonian Path)이다. 검사장비의 총 이동시간을 최소화하는 최적의 작업경로를 생성시키는 문제로 식 (4), (11)과 같이 정수계획 문제로 표현된다[4].

$$\min T_{move} = \min \sum_{i=0}^m \sum_{j=0}^m c_{ij} x_{ij} \quad (4)$$

s. t

$$\sum_{i=0}^m x_{ij} = 1, (i=0, 1, \dots, m)$$

$$\sum_{j=0}^m x_{ij} = 1, (j=0, 1, \dots, m)$$

$$x_{ij} \in \{0, 1\}, (i, j=0, 1, \dots, m) \quad (11)$$

$$N_s = b$$

$$y_{ij} \leq (m-1) x_{ij}, (i, j=0, 1, \dots, m)$$

$$y_{ij} \geq 0, (i, j=0, 1, \dots, m)$$

3. FOV 생성 알고리즘

본 논문에서 제안하는 FOV 생성 알고리즘은 그림 3과 같이, 크게 검사점의 탐색, 4-corner move와 window centering의 세 단계로 나누어 실행한다. 검사점의 탐색 단계에서는 PCB의 중심점에서 가장 먼 검사점을 선택한다. 선택된 검사점이 이미 FOV에 포함되어 있다면 다른 검사점을 선택하고, 그렇지 않다면 4-corner move를 적용한다. 4-corner move란 FOV안의 영역에서 검사점이 놓여있을 때 가장 많은 검사점영역을 포함하는 FOV 초기 셋팅 단계이다. 4-corner move에서 선택된 FOV를 이용해 PCB 중앙에서 먼 쪽으로 이동을 시도한다. 이 동작은 검사점이 홀로 떨어져남아서 후에 다시 하나의 FOV를 갖지 않도록, FOV의 과다성을 줄이기 위한 동작이

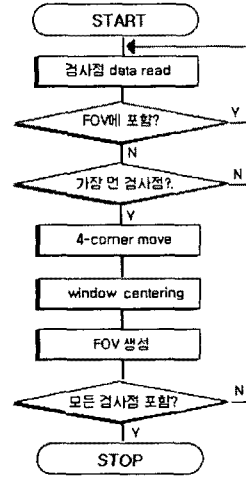


그림 3. Convex hull 알고리즘의 순서도
Fig. 3. Flowchart of Convex Hull Algorithm

다. 하나의 FOV에 많은 윈도우를 포함하는 것도 좋지 않, 선택되지 않은 여분의 검사점으로 인해 생성되는 FOV를 줄이기 위한 방법이다.

제안한 convex hull 알고리즘을 정리하면 다음과 같다.

- S1. (scan) 스캔은 PCB의 중앙점(center point)에서 가장 멀리 떨어진 검사점 순으로 한다. FOV에 포함되지 않은 검사점을 선택한다.
- S2. (FOV 생성) FOV의 생성은 4-corner move와 window centering의 두 과정으로 한다.
 - S2.1. (4-corner move) 선택된 검사점이 window의 1사분면, 2사분면, 3사분면, 4사분면의 구석에 위치하도록 놓고, 각 경우의 gain을 구한다.

$$4\text{-corner move의 gain} = \frac{\text{완전히 포함되는 검사점의 수}}{\text{포함되는 검사점의 총수}}$$
 만약 gain이 1인 경우가 하나면 현재의 사분면을 선택하고, 여러 개면 완전히 포함되는 검사점의 수가 가장 많은 사분면을 선택, 완전히 포함되는 검사점의 수가 같으면 포함되는 검사점의 수가 가장 많은 사분면 선택한다.
 - S2.2. (centering) 다음의 4경우로 나누어 window를 이동하며, 이동되는 사분면 안에 포함되어진 검사점 중에서 PCB 중앙점에서 가장 먼 검사점을 포함하는 방향으로 이동한다.
 - 1사분면 선택된 경우: 2사분면이나 4사분면으로 이동
 - 2사분면 선택된 경우: 1사분면이나 3사분면으로 이동
 - 3사분면 선택된 경우: 2사분면이나 4사분면으로 이동
 - 4사분면 선택된 경우: 1사분면이나 3사분면으로 이동
- S3. 모든 검사점이 FOV에 포함될 때까지 S1.부터 S2.를 반복한다.

4. 시뮬레이션 결과

시뮬레이션은 실제의 AOI 검사장비를 대상으로 수행하였다. 검사장비에 장착된 camera의 FOV 크기는 20×15mm이고, 검사속도는 0.25sec/frame이다. 로봇의 이동속도는 사다리꼴 가감속 프로파일을 X축과 Y축에 함께 적용하였으며, 최고운동속도는 700mm/sec이며, 가속 시간은 0.2sec이다.

위의 검사장비를 대상으로 FOV 생성 최적화와 생성된 FOV의 검사경로 최적화의 두 가지 유형에 대한 시뮬레이션을 하였다. FOV 생성 최적화의 시뮬레이션에서는, 본 논문에서 제안하는 convex hull 알고리즘과의 비

교를 위하여 greedy 알고리즘과 S-shape 알고리즘의 세 가지 FOV 생성 알고리즘에 대한 시뮬레이션을 수행하여, 각 알고리즘별 생성되는 FOV의 수를 비교하였다. 검사경로 최적화의 시뮬레이션에서는 TSP의 heuristic 알고리즘 중에서 널리 사용되는 nearest neighbor 알고리즘과 2-opt 알고리즘을 이용하여 세 가지의 FOV 생성 알고리즘의 총 검사시간을 비교하였다.

시뮬레이션을 위한 PCB 데이터는 표 1과 같다. 검사점은 이미지 검사점과 납땜 검사점, 그리고 브릿지 검사점의 세 가지의 검사점을 포함하고 있다. PCB 데이터는 실제의 데이터(data)를 기반으로 검사점의 수와 분포도에 임의의 변화를 주어 만들었다.

시뮬레이션 결과 FOV 생성 최적화 단계에서는, greedy 알고리즘에 비해 S-shape 알고리즘과 convex hull 알고리즘은 표 2와 같이 약 8.25%~17.36% 개선되었다. 생성된 FOV의 모습은 그림 4에서 알 수 있듯이 FOV간의 불필요한 중첩이 줄어들었음을 알 수 있다. FOV의 수가 줄어들어 따라 검사장비의 총 검사시간도 7.28%~15.54% 향상된 결과를 보였다.

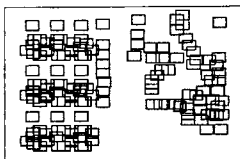
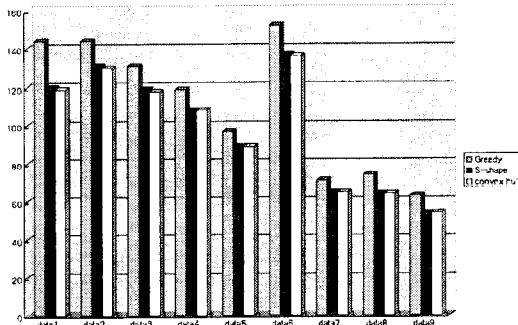
표 1. PCB 데이터의 사양

Table 1. PCB Data

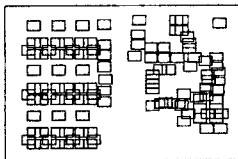
	검사점 수	이미지	납땜	브릿지
data1	3213	161	2868	184
data2	3213	161	2868	184
data3	3057	137	2748	172
data4	2591	119	2322	150
data5	2167	79	1970	118
data6	3213	161	2868	184
data7	1428	56	1294	78
data8	1428	56	1294	78
data9	1199	45	1088	66

표 2. 시뮬레이션 결과 FOV의 총 수

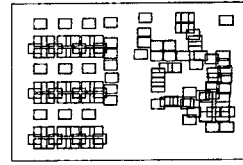
Table 2. Simulation Result Total Number of FOV



(a) Greedy 알고리즘



(b) S shape 알고리즘



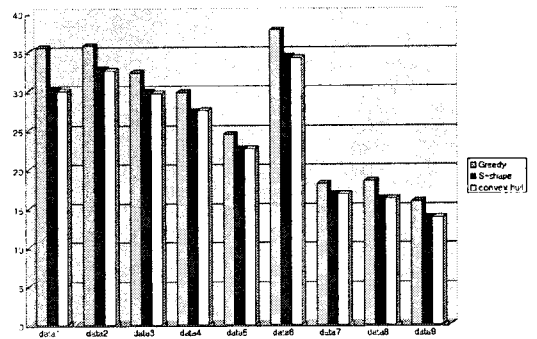
(c) Convex Hull 알고리즘

그림 4. 생성된 FOV의 모습

Fig. 4. Figure of FOV

표 3. 시뮬레이션 결과 총 검사시간

Table 3. Simulation Result Total Inspection Time



5. 결 론

본 논문에서는 SMT형 PCB 조립라인의 AOI 검사장비의 검사시간 단축을 위한 경로계획 알고리즘을 제안하였다. 검사장비의 경로계획 문제를 수식화하고, 문제용 FOV 생성 알고리즘 최적화와 검사경로 최적화의 계층적 구조로 구성하였다. 그리고 FOV 생성을 위하여 heuristic 알고리즘인 convex hull 알고리즘을 제안하여, 시뮬레이션을 통해 FOV의 수와 검사시간의 개선을 확인하였다.

본 논문을 통해 AOI 검사장비의 효율적인 검사 경로를 위한 경로계획 방법을 제시하였다. 제시된 방법보다 더 개선된 알고리즘에 대한 연구가 필요하다.

(참 고 문 헌)

- [1] 박태형, "전자조립용 CAM 시스템의 기술동향", 전자공학회지, vol.26, no.3, pp. 48-61, 1999
- [2] 오세휘, 차영엽, "비전 시스템에서 신경 회로망을 이용한 검사영역에 관한 연구", 제어자동화시스템공학회지, vol.4, no.3, pp. 378-383, 1998
- [3] G. Reinelt, "The Traveling Salesman: Computational Solutions for TSP Applications", Springer-Verlag, 1994
- [4] E. L. Lawler, et. al., "The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization", A Wiley Interscience Publication, 1985
- [5] Samsung SMT Solution Catalog, "AI400", 삼성테크윈, 2002