

# SMT 검사기의 경로계획을 위한 클러스터링 알고리즘

## A Clustering Algorithm for Path Planning of SMT Inspection Machines

김화중 · 박태형

Hwa-Jung Kim and Tae-Hyoung Park

충북대학교 대학원 제어계측공학과, 컴퓨터 정보통신 연구소

### 요약문

인쇄회로기판을 조립하는 SMT (surface mount technology) 라인의 AOI (automatic optical inspection) 형 검사기를 대상으로, 검사시간 단축을 위한 경로계획 방법을 제안한다. 기판에 존재하는 검사 윈도우들은 카메라의 FOV (field-of-view) 크기를 고려하여 클러스터링 되어야 하며, 전체 검사시간의 단축을 위하여 클러스터의 수를 최소화하는 것이 바람직하다. 주어진 기판에 대한 클러스터의 수를 최소화하기 위한 유전자 알고리즘을 새로이 제안하며, 이를 사용한 효과적 경로계획 방법을 제시한다. 상용 검사기를 대상으로 시뮬레이션을 수행하며, 비교 평가를 통하여 제안된 방법의 유용성을 검증한다.

### Abstract

We propose a path planning method to reduce the inspection time of AOI (automatic optical inspection) machines in SMT (surface mount technology) in-line system. Inspection windows of board should be clustered to consider the FOV (field-of-view) of camera. The number of clusters is desirable to be minimized in order to reduce the overall inspection time. We newly propose a genetic algorithm to minimize the number of clusters for a given board. Comparative simulation results are presented to verify the usefulness of proposed algorithm.

**Keywords** : clustering, optimization, genetic algorithm, inspection machines, SMT

### 1. 서 론

최근 컴퓨터, 휴대폰, PDA 등의 보급으로 전자제품의 수요가 급증함에 따라, 표면실장기술(SMT; surface mount technology)을 이용한 인쇄회로기판(PCB; printed circuit board) 전자조립시스템의 중요성이 증대되고 있다. SMT 인라인(in-line) 시스템은 표면실장부품(SMD; surface mount device)을 인쇄회로기판에 조립하는 전자조립시스템으로서, 표면실장 전자부품의 소형화, 고밀도화 등 부품제작기술의 발전과 조립라인 자동화 추세에 의해 현재 대부분의 전자제품 제조라인에서 채택되고 있다. [1]

SMT 인라인 시스템을 구성하는 장비 중 검사기는 부품 장착 이상, 납땜 불량, 회로 단락 등 인쇄회로기판 조립 시 발생하는 이상을 자동으로 검출하는 전용 로봇 시스템이다. 영상처리기술의 발달에 힘입어 최근 많이 보급되고 있으며, 조립라인의 생산성 향상 및 품질 신뢰성 향상에 크게 기여하고 있다. SMT 검사기의 대부분은, 직교좌표형 로봇에 설치된 카메라 (area camera) 가 기판 위를 수평 이동하며 영상을 획득하는 AOI (automatic optical inspection) 형이다. [2] 본 논문은 AOI 형 SMT 검사기의 검사시간 단축을 위한 방

법을 제안한다.

그림 1 은 AOI 형 SMT 검사기의 평면도를 보여준다. FOV (field-of-view) 는 정상적인 처리가 가능한 카메라의 영상 획득영역으로서, 가로 세로 각 수십 mm 의 크기로 제한된다. 따라서 전체 기판의 검사에 여러 번의 영상획득이 필요

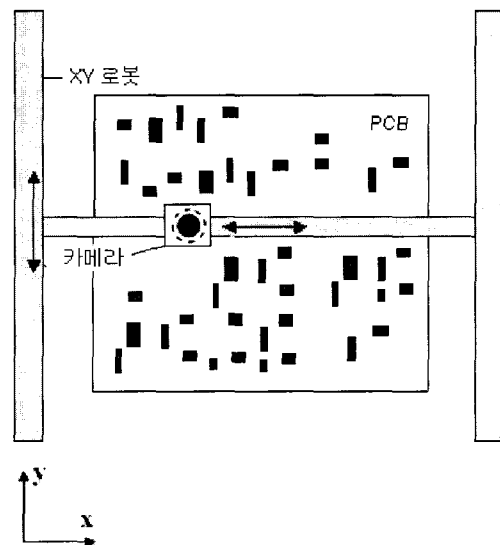


그림 1. SMT 검사기의 평면도.  
Fig 1. A SMT inspection machine.

접수일자 : 2003년 3월 10일  
완료일자 : 2003년 5월 20일  
본 연구는 한국과학재단 목적기초연구(R05-2002-001164-0)의 지원에 의하여 수행 되었습니다.

하며, 이를 위하여 기판에 분포된 검사 대상 윈도우들을 여러 개의 묶음으로 만들어야 한다. 하나의 FOV 에 포함될 검사윈도우들의 묶음을 클러스터라 정의하며, 전체 클러스터의 수는 검사기의 영상획득 횟수와 동일하다. 즉 클러스터의 수를 단축하면 전체 영상획득에 소요되는 시간을 단축할 수 있으며, 또한 카메라의 이동시간도 단축할 수 있다. 본 논문은 전체 클러스터의 수를 최소화하기 위한 알고리즘을 제시하며, 이를 이용한 검사기의 효과적 경로계획방법을 제안하고자 한다.

SMT 검사기를 대상으로 발표된 연구는 많으나, 주로 영상처리방법에 관한 연구에 집중되었으며 [3], 경로계획 방법에 관하여 발표된 논문은 매우 드물다. 김철우 [4] 는 순차적인 클러스터링 알고리즘을 제안하였고, 오제희 등 [5] 은 Kohonen's self organizing map 을 적용한 클러스터링 알고리즘을 제안하였다. 단일 연결 알고리즘, 완전 연결 알고리즘, k-means 알고리즘, ISODATA 알고리즘 등 잘 알려진 클러스터링 알고리즘들 [6][7] 은 본 논문에서 다루는 검사기 클러스터링 문제에 직접 적용하기 힘들다. 그 이유는 전체 클러스터의 수가 주어진 값이 아니고, 결정하여야 하는 값이기 때문이다.

본 논문은 검사기 경로계획을 위한 유전자 알고리즘을 제안한다. 검사기 문제의 해결에 적절한 염색체를 새로이 정의하며, 교배 및 돌연변이 방법을 제시한다. 또한 단일 연결 알고리즘 및 ISODATA 알고리즘을 검사기 클러스터링 문제에 적용하는 방법을 제시하며, 새로이 S자형 알고리즘을 제시한다. 상용 검사기를 대상으로, 여러 가지 알고리즘을 적용하여 비교평가를 수행하며, 본 논문에서 새로이 제시한 유전자 알고리즘의 유용성을 검증한다.

## 2. 검사기 경로계획 문제

그림 2 는 AOI 형 SMT 검사기의 기판과 윈도우, FOV 및 클러스터를 보여준다. 검사기가 기판에서 검사하고자 하는 영역을 윈도우라 정의한다. 윈도우는 모두 사각형의 영역이며, 부품 윈도우, 납땜 윈도우 및 브릿지 윈도우로 구분될 수 있다. 부품 윈도우는 부품 장착의 이상, 납땜 윈도우는 납땜 불량, 브릿지 윈도우는 리드 간 단락을 검출하는 영역이다. 하나의 부품에 대하여 여러 개의 윈도우가 설정될 수 있으며, 하나의 기판에 수백 - 수천 개의 윈도우가 존재한다. 검사기는 카메라를 이동시켜 모든 윈도우에 대한 영상을 획득하고 처리하여 검사한다.

카메라에서 한 번에 획득되어 정상적으로 처리될 수 있는 영역의 크기를 FOV 라 정의한다. FOV 의 크기는 검사기의 고유값으로서 통상 가로 세로 각 수십 mm 이내의 크기를 갖으며, 수 개의 윈도우를 포함할 수 있다. 따라서 기판에 존재하는 모든 윈도우에 대한 검사를 수행하기 위해서는, 카메라를 이동시켜 여러 번의 영상을 획득하여야 한다. 이 때 각 FOV 에 포함되는 윈도우들의 묶음을 클러스터라 정의한다. 각 클러스터의 중심위치에 카메라가 일정 시간 정지하여 영상을 획득하고, 다른 클러스터로 이동하며 검사작업을 수행한다. 카메라가 기판에 존재하는 모든 클러스터를 방문하여야 검사가 완료된다. 이 때 각 클러스터를 방문하는 카메라의 이동경로를 검사경로라 정의한다. 기판에 생성되는 클러스터의 수는 카메라의 정지횟수와 영상획득횟수와 동일하다. 그러므로 클러스터의 수를 감소시키면 영상획득시간은 물론 카메라 이동시간을 단축시킬 수 있으며, 궁극적으로 검

사시간을 단축시킬 수 있다.

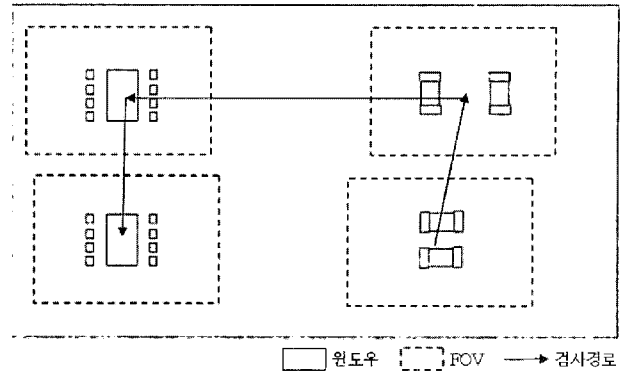


그림 2. 윈도우, FOV 및 검사경로.  
Fig 2. Windows, FOV and path.

일반적인 검사기의 경로계획은 다음의 두 단계로 이루어진다. 우선 주어진 윈도우에 대하여 최소의 수를 갖는 클러스터들을 생성한다. 다음 단계로서, 생성된 클러스터로부터 최소시간의 방문경로를 생성한다. 첫 단계의 문제는 클러스터링 문제의 범주에 해당하며, 두 번째 단계의 문제는 외판원 문제의 범주에 해당한다. 특히 두 번째 단계의 경우 전형적인 외판원 문제로서 잘 알려진 TSP (traveling salesman problem) 알고리즘 들 [8][9]의 직접 적용이 가능하다. 그러나 첫 번째 단계의 경우, 전형적 클러스터링 문제와는 목적 함수 및 결정변수에 있어서 차별화 된다.

전형적 클러스터링 문제는 클러스터의 중심과 윈도우 사이의 총 거리를 최소화하도록 하는 클러스터를 생성하는 문제로서, 전체 클러스터의 수는 주어진 값이다. 그러나 검사기 클러스터링 문제는 각 클러스터의 크기 제한 (FOV 제한조건)을 고려하여 전체 클러스터의 수를 최소화하는 문제이다. 그러므로 전형적 클러스터링 문제에 적용되는 알고리즘을 검사기 클러스터링 문제에 직접 적용하는 것은 매우 어렵다.

검사기 클러스터링 문제를 수학적으로 정의한다. 윈도우의 집합을  $W = \{w_1, \dots, w_m\}$  라 하면, 클러스터  $i$  에 속한 윈도우들의 집합  $W_i$  는 다음과 같다.

$$W_i = \{w_k \in W \mid c_i^x - \frac{f^x}{2} \leq w_k^x \leq c_i^x + \frac{f^x}{2}, c_i^y - \frac{f^y}{2} \leq w_k^y \leq c_i^y + \frac{f^y}{2}\} \quad (1)$$

단,  $(w_k^x, w_k^y)$  는 윈도우  $w_k$  의 x, y 축 좌표값이며,  $(c_i^x, c_i^y)$  는 클러스터  $i$  중심으로서 집합  $W_i$  에 속한 윈도우들의 무게중심 값이다. 또한  $(f^x, f^y)$  는 FOV 의 x, y 축 크기이다.

검사기 클러스터링 문제는 집합  $W_1, \dots, W_n$  을 구하는 최적화 문제로서 다음과 같다.

$$\text{minimize } n \quad (2)$$

s.t.

$$\bigcup_{i=1}^n W_i = W \quad (3)$$

$$\bigcap_{i=1}^n W_i = \emptyset \quad (4)$$

식(3)은 모든 윈도우는 클러스터에 포함되어야 함을 의미하며, 식(4)는 각 윈도우는 여러 개의 클러스터에 중복되어

소속될 수 없음을 의미한다. 본 논문은 위의 검사기 클러스터링 문제에 대한 해법을 제시하며, TSP 알고리즘과 결합하여 검사기 경로계획 방법을 개발한다.

### 3. 검사 클러스터링 알고리즘

앞장에서 정의된 검사기 클러스터링 문제의 해를 구하기 위한 알고리즘을 제시한다. 전형적인 클러스터링 문제에 적용되는 단일연결 알고리즘과 ISODATA 알고리즘을 본 문제에 응용하여 적용하며, 새로이 S자형 알고리즘을 제시한다. 또한 유전자 알고리즘을 검사기 클러스터링 문제에 적용하는 방법을 제안한다.

#### 3.1 단일연결 알고리즘

단일연결 (single link) 알고리즘 [6][10] 은 전형적 클러스터링 알고리즘의 하나로서, 초기에 많은 클러스터를 생성시키고, 이를 병합하며 진행하는 방법이다. 매우 간단한 구현을 특징으로 하나, 국지적 해법을 사용함으로 인하여, 윈도우 분포에 따라 좋지 않은 해가 생성될 수 있다. 검사기 클러스터링 문제에 적용하기 위하여 클러스터 생성 시 FOV 크기를 추가로 고려하여야 한다. 또한 클러스터의 수를 줄이기 위하여, 병합을 가능한 계속하여야 한다. 다음은 검사기 클러스터링을 위한 단일연결 알고리즘이다.

- S1. 각 윈도우를 클러스터로서 생성한다.
- S2. 각 클러스터에 대하여, 병합 후 크기가 가장 작아지는 이웃의 클러스터를 찾는다. 병합 후 크기가 FOV 크기를 초과하지 않는 경우 병합한다.
- S3. 더 이상의 병합이 이루어지지 않을 때까지 S2 를 반복한다.

#### 3.2 ISODATA 알고리즘

ISODATA (iterative self-organizing data analysis) 알고리즘은 전형적 클러스터링 알고리즘의 하나로서, 보편적으로 많이 활용되는 알고리즘이다. k-means 알고리즘 [9] 을 보완한 방법으로서, 클러스터의 병합 및 분리 과정을 포함한다. 단일연결 알고리즘과 마찬가지로 초기에 모든 클러스터를 생성하고 이를 개선하며 진행하는 방식으로서, 초기 해에 따라서 해의 성능이 달라질 수 있다. 이 알고리즘을 검사기 클러스터링 문제에 적용하기 위하여, 클러스터 생성 및 병합 시 FOV 크기를 추가로 고려하여야 한다. 다음은 검사기 클러스터링을 위한 ISODATA 알고리즘이다.

- S1. 기관을 가로 세로 일정한 간격으로 분할하여, 각 구역을 클러스터로서 생성한다.
- S2. 만약 클러스터에 윈도우가 포함되지 않았다면 클러스터를 제거한다.
- S3. 남아있는 클러스터 중 윈도우가 경계선에 있다면 클러스터의 중심점을 이동해서 윈도우를 포함시킨다.
- S4. 클러스터를 이동하여 더 많은 윈도우를 포함할 수 있다면, 클러스터의 중심점을 이동한다.
- S5. 이동을 한 뒤 클러스터에서 속하지 않는 윈도우가 있다면, 이 윈도우에 대한 클러스터를 생성한다.
- S6. 각 클러스터에 대하여, 이웃한 클러스터와 병합했을 때 크기가 FOV 를 초과하지 않는다면, 두 클러스터를 병합한다.

S7. 더 이상 개선이 없을 때까지 S4-S6 과정을 반복한다.

### 3.3 S자형 알고리즘

S자형 알고리즘은 검사기 클러스터링 문제를 위하여 새로이 제안하는 알고리즘이다. 앞의 단일연결 알고리즘 및 ISODATA 알고리즘과 달리, 클러스터를 하나 씩 생성시키며 진행하는 방법을 사용한다. 기관의 모서리에서 최초 클러스터를 생성하며, 이웃한 윈도우로 이동하여 다음의 클러스터를 생성한다. 이때 S 자형으로 클러스터의 중심을 이동하며 클러스터를 생성시킨다. 이 방법은 비교적 간단히 구현되며, 기관 검사에 적절한 해를 얻을 수 있다. 그러나 직관적 경험에 의존한 방법으로서, 윈도우 분포에 따라 해의 성능이 크게 달라질 수 있다. 다음은 검사기 클러스터링을 위한 S자형 알고리즘이다.

- S1. 기관의 우 하단의 윈도우를 중심으로 클러스터를 생성한다.
- S2. 생성된 클러스터의 중심점을 FOV 가 허용하는 범위 내에서 움직이며 주변 윈도우를 포함시킨다.
- S3. 생성된 클러스터를 기준으로 최 근접의 윈도우를 중심으로 새로운 클러스터를 생성한다.
- S4. S2-S3 의 과정을 윈도우가 클러스터에 모두 포함될 때까지 기관을 S자형으로 움직이며 반복한다.

### 3.4 유전자 알고리즘

유전자 알고리즘은 국지적 해에 머물 수 있는 최적화 알고리즘을 보완하기 위하여 널리 사용되고 있다. [11] 또한 전형적 클러스터링 문제에도 적용되고 있다. [12] 여러 개의 염색체가 하나의 세대를 이루며, 세대를 거듭하며 복제, 교배 및 돌연변이를 통하여 염색체가 진화된다. 진화과정에서 우수한 적합도를 갖는 염색체를 선택하여 최적화 문제의 해를 구한다. 유전자 알고리즘은 복잡한 문제에 적용되어 근사 최적해를 구하는 데 적합한 알고리즘이다. 유전자 알고리즘을 구현하기 위하여 문제에 맞는 염색체 및 적합도 함수를 정의하여야 한다. 또한 복제 연산자, 교배 연산자 및 돌연변이 연산자를 제약조건을 고려하여 정의하여야 한다. 본 논문은 검사기 클러스터링 문제에 적용될 수 있는 유전자 알고리즘을 제안한다. 이를 위하여 염색체, 적합도 함수 및 연산자들을 새로이 정의한다.

#### 3.4.1 염색체 및 적합도

유전자 알고리즘 수행의 기본 단위인 염색체 (chromosome) 를 그림 3 과 같이 정의한다. 염색체를 구성하는 유전자 (gene) 는 윈도우 번호  $w_{ij} \in W_i$  또는 0 의 값을 갖는다. 이 때 0 은 클러스터 사이의 구분을 위하여 사용하는 기호이며,  $w_{ij}$  는 0 보다 큰 정수로서,

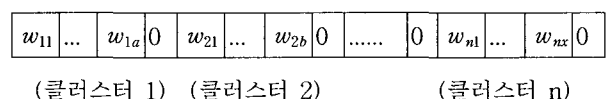


그림 3. 염색체의 정의.

Fig 3. Definition of chromosome.

클러스터  $i$  에 속한 윈도우 번호이다. 염색체는 모든 윈도우를 유전자로서 포함하며, 염색체에 의하여 전체 윈도우 및 클러스터를 한 번에 표현할 수 있다.

한 세대에 대하여  $p$  개의 염색체를 생성시킨다. 처음 세대의 염색체는 S자형 알고리즘을 사용하여 생성하며, 시작 윈도우를 무작위로 선택하여 여러 개의 염색체를 만든다. 어떤 세대의  $k$  번째 염색체  $V_k$  ( $k=1, \dots, p$ )의 길이를  $l_k$ , 클러스터 수를  $n_k$ , 윈도우의 수를  $m$  이라 하면, 다음의 관계가 성립된다.

$$n_k = l_k - m \quad (5)$$

즉, 클러스터의 수는 염색체 길이와 윈도우 수의 차로 서 구할 수 있다. 한 세대에 대한 클러스터수의 최대값을  $n_{max}$  라 하면,

$$n_{max} = \max_{k=1, \dots, p} n_k \quad (6)$$

염색체  $V_k$ 의 적합도  $e_k$ 를 다음과 같이 정의한다.

$$e_k = (n_{max} - n_k) \times \frac{p}{\sum_{i=1}^p (n_{max} - n_i)} \quad (7)$$

최대의 클러스터 수를 갖는 염색체의 적합도는 0이며, 클러스터의 수가 감소할수록 적합도는 증가한다. 평균값을 갖는 경우의 적합도는 1이다.

적합도는 유전자 알고리즘의 복제 또는 선택과정에서 사용된다. 즉 적합도가 높은 염색체는 다음 세대로 복제되며, 낮은 염색체는 도태시킨다. 기본 유전자 알고리즘에서 사용되는 룰렛 휠을 사용하여 확률적인 선택을 하며, 확률샘플링 오류를 방지하기 위하여 잔여확률샘플링 (remainder stochastic sampling) [11] 방법을 사용한다.

### 3.4.2 교배 연산자

교배는 염색체에 변화를 주어 우수한 적합도를 갖는 염색체로 진화시키기 위한 과정이다. 교배확률을 사용하여 두 개의 염색체  $V_1, V_2$ 를 선택하고, 다음의 교배 연산을 수행한다.

- S1. 염색체  $V_1$ 에서 하나의 클러스터를 임의로 선택한다.
- S2. 염색체  $V_1$ 에서 선택된 클러스터의 윈도우들을 염색체  $V_2$ 에서 모두 찾는다. 이들 윈도우를 포함하는 클러스터를 모두 선택한다.
- S3.  $V_2$ 의 선택된 클러스터가 포함하는 윈도우들을 염색체  $V_1$ 에서 모두 찾는다. 이들 윈도우를 포함하는 클러스터를 모두 선택한다.
- S4.  $V_1$ 과  $V_2$ 의 선택된 클러스터가 포함하는 윈도우가 동일할 때까지 S2-S3를 반복한다.
- S5.  $V_2$ 의 선택된 클러스터들을  $V_1$ 으로 이동시켜  $V_1'$ 을 얻고,  $V_1$ 의 선택된 클러스터들을  $V_2$ 로 이동시켜  $V_2'$ 를 얻는다.

예를 들어, 다음과 같은 두 개의 염색체가 선택된 경우,

$V_1$	1	3	0	4	5	10	11	15	0	13	0	16	18	7	8	2	14	0	6	9	12	17	0	
$V_2$	1	3	4	6	10	0	18	7	0	8	2	14	0	9	0	13	16	0	5	11	12	15	17	0

$V_1$ 에서 하나의 클러스터를 임의로 선택한다.

$V_1$	1	3	0	4	5	10	11	15	0	13	0	16	18	7	8	2	14	0	6	9	12	17	0
-------	---	---	---	---	---	----	----	----	---	----	---	----	----	---	---	---	----	---	---	---	----	----	---

선택된 클러스터의 모든 윈도우를  $V_2$ 에서 찾고, 이를 포함하는 클러스터를 선택하면 다음과 같다.

$V_2$	1	3	4	6	10	0	18	7	0	8	2	14	0	9	0	13	16	0	5	11	12	15	17	0
-------	---	---	---	---	----	---	----	---	---	---	---	----	---	---	---	----	----	---	---	----	----	----	----	---

$V_2$ 의 윈도우 13이  $V_1$ 에는 누락되어 있으므로,  $V_1$ 에서 윈도우 13을 포함하는 클러스터를 선택한다.

$V_1$	1	3	0	4	5	10	11	15	0	13	0	16	18	7	8	2	14	0	6	9	12	17	0
-------	---	---	---	---	---	----	----	----	---	----	---	----	----	---	---	---	----	---	---	---	----	----	---

선택된 윈도우가 모두 동일하므로, 선택된 클러스터들을 상대편 염색체로 이동시켜 새로운 염색체를 만든다. 즉,

$V_1$	1	3	0	4	5	10	11	15	0	18	7	0	8	2	14	0	13	16	0	6	9	12	17	0
-------	---	---	---	---	---	----	----	----	---	----	---	---	---	---	----	---	----	----	---	---	---	----	----	---

$V_2$	1	3	4	6	10	0	13	0	16	18	7	8	2	14	0	9	0	5	11	12	15	17	0
-------	---	---	---	---	----	---	----	---	----	----	---	---	---	----	---	---	---	---	----	----	----	----	---

이와 같은 교배를 수행할 경우, 검사기 클러스터링 문제의 제한조건을 만족하는 염색체를 새로이 생성시킬 수 있으며, 적합도에도 변화가 발생된다.

### 3.4.3 돌연변이 연산자

돌연변이는 교배와 달리 하나의 염색체를 임의로 선택하고 이를 변이시키는 것이다. 돌연변이 확률을 사용하여 염색체  $V_1$ 을 선택하고 다음의 돌연변이 연산을 수행한다.

- S1. 염색체  $V_1$ 에서 하나의 클러스터를 임의로 선택한다.
- S2. 선택된 클러스터의 모든 윈도우에 대하여, 염색체 내의 다른 클러스터로 이동가능한지의 여부를 확인한다. 가능 윈도우를 모두 이동시켜 새로운 염색체  $V_1'$ 을 얻는다.

예를 들어 다음의 염색체가 선택된 경우,

$V_1$	3	0	4	17	11	15	0	13	9	6	2	0	16	8	7	1	5	14	0	10	12	0
-------	---	---	---	----	----	----	---	----	---	---	---	---	----	---	---	---	---	----	---	----	----	---

임의의 클러스터를 선택한다.

$V_1$	3	0	4	17	11	15	0	13	9	6	2	0	16	8	7	1	5	14	0	10	12	0
-------	---	---	---	----	----	----	---	----	---	---	---	---	----	---	---	---	---	----	---	----	----	---

선택된 클러스터에 포함된 각 윈도우 (13, 9, 6, 2)를, 이동이 가능한 다른 클러스터로 이동시킨다. 이때 FOV 크기를 고려하여 이동가능 클러스터를 결정하여야 한다.

$V_1'$	3	6	0	4	17	11	15	0	13	9	0	16	8	7	1	5	14	0	10	12	2	0
--------	---	---	---	---	----	----	----	---	----	---	---	----	---	---	---	---	----	---	----	----	---	---

교배와 마찬가지로 조건을 만족시키며 염색체를 변이시키고, 클러스터의 수를 감소하는 방향으로의 변이가 가능하다.

### 4. 시뮬레이션

삼성테크윈(주)의 SMT 검사기 AI400 [13] 을 대상으로 시뮬레이션을 수행하였다. 카메라 FOV 크기는 16(mm)×12(mm) 이고, 영상획득시간은 0.25(sec/FOV) 로 설정하였다. 또한 XY 축은 사다리꼴 가속을 갖는다고 가정하였으며, 최고속도는 700(mm/sec), 가속시간은 0.2(mm/sec<sup>2</sup>) 로 설정하였다. 기관의 최대 크기는 256(mm)×245(mm) 로 하였으며, 검사 윈도우의 크기는 최소 0.9(mm)×0.25(mm) 에서 최대 5.7(mm)×3.9(mm) 로 하였다.

본 논문에서 제시한 4 개의 검사기 클러스터링 알고리즘은 Visual C++을 사용하여 프로그래밍 되었으며, MS-Windows XP / IBM-PC Pentium IV 1.5 GHz 상에서 구현되었다. 유전자 알고리즘의 경우 한 세대의 염색체 수는 100 개, 교배 확률은 50%, 돌연변이 확률은 80%로 하였으며, 세대수는 200 개로 제한하였다.

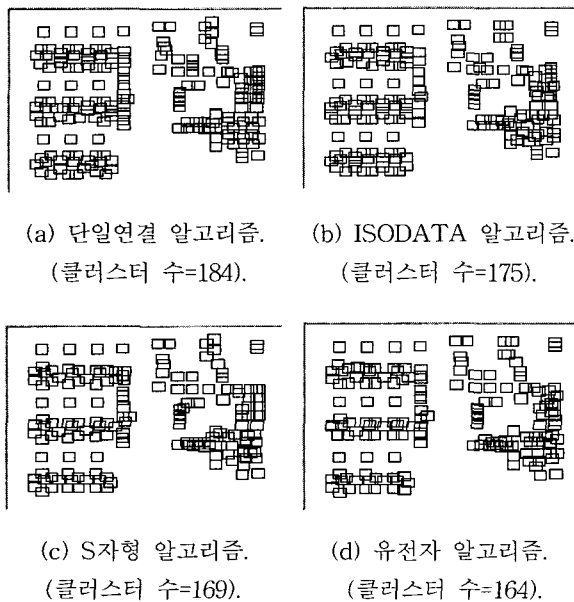


그림 4. 알고리즘 별 클러스터 생성(윈도우 수=3029).  
 Fig 4. Examples of cluster generation (number of windows=3029).

또한 클러스터링이 종료된 후, 클러스터를 모두 방문하는 카메라의 이동경로를 생성하는 프로그램을 전형적 TSP 알고리즘을 사용하여 구현하였다. 최근접 탐색법 (nearest neighbor) 으로 초기 경로를 생성하고 2-opt 방법으로 경로를 생성시키는 방법 [8] 을 적용하였다.

그림 4 는 동일한 기관에 각 알고리즘을 적용하여 생성시킨 전체 클러스터의 모습을 보여준다. 알고리즘에 따라 생성된 클러스터의 수가 상이하며, 또한 분포된 모습은 현저한 차이를 보였다. 즉 같은 수의 클러스터 수를 갖는 경우에도 이동시간에 차이가 발생할 수 있음을 보여준다.

표 1 은 알고리즘 별 클러스터 수 및 검사시간을 비교하여 보여준다. 다양한 윈도우 수 및 분포를 갖는 10개의 기관에 대하여 4 가지 알고리즘을 적용하였다.

표 1. 알고리즘 별 클러스터 수 및 검사시간.

Table 1. Total cluster and inspection time for algorithms.

기관번호	윈도우 수	단일연결 알고리즘		ISODATA 알고리즘		S자형 알고리즘		유전자 알고리즘	
		클러스터 수	검사시간 (sec)	클러스터 수	검사시간 (sec)	클러스터 수	검사시간 (sec)	클러스터 수	검사시간 (sec)
1	3029	184	67.48	175	65.09	169	62.92	164	60.45
2	3029	181	66.56	180	65.84	165	61.73	162	59.75
3	2885	172	63.38	164	60.73	155	57.76	151	56.58
4	2441	154	58.05	150	56.02	145	54.54	139	52.01
5	2049	121	46.13	119	45.11	116	44.39	110	41.99
6	3029	177	65.05	173	63.99	170	63.68	161	59.52
7	3029	182	68.04	181	67.85	170	63.78	166	62.60
8	1350	84	32.81	85	33.35	81	31.79	77	30.30
9	1428	83	32.04	82	31.30	80	30.98	76	29.19
10	1133	72	27.81	69	26.85	68	26.46	64	24.87

표 2. 유전자 알고리즘의 개선율.

Table 2. Improvement ratio of genetic algorithm.

기관번호	윈도우 수	단일연결 알고리즘		ISODATA 알고리즘		S자형 알고리즘	
		클러스터 수 (%)	검사시간 (%)	클러스터 수 (%)	검사시간 (%)	클러스터 수 (%)	검사시간 (%)
1	3029	12.1	11.6	6.7	7.7	3.0	4.1
2	3029	11.7	11.4	11.1	10.2	1.9	3.3
3	2885	13.9	12.0	8.6	7.3	2.6	2.1
4	2441	10.8	11.6	7.9	7.7	4.3	4.9
5	2049	10.0	9.9	8.2	7.4	5.5	5.7
6	3029	9.9	9.3	7.5	7.5	5.6	7.0
7	3029	9.6	8.7	9.0	8.4	2.4	1.9
8	1350	9.1	8.3	10.4	10.1	5.2	4.9
9	1428	9.2	9.8	7.9	7.2	5.3	6.1
10	1133	12.5	11.8	7.8	8.0	6.3	6.4
평균 개선율		10.0	10.4	8.5	8.2	4.2	4.6

각 기관에 대한 알고리즘 별 생성 클러스터 수 및 검사시간을 볼 수 있다. 이 검사시간은 영상획득시간과 이동시간의 총합으로 계산하였다. 평균적으로 유전자 알고리즘, S자형, ISODATA, 단일연결의 순서로 좋은 해를 얻을 수 있었다. 표 2 는 알고리즘 별 성능비교를 위하여, 가장 우수한 유전자알고리즘을 기준으로 개선율을 비교한 것이다. 유전자 알고리즘은 다른 알고리즘에 비해 4 ~ 10 (%) 의 성능 개선 효과가 있음을 확인하였다.

### 5. 결론

인쇄회로기관 조립라인에서 그 중요성이 증가하고 있는 SMT 검사기를 대상으로, 검사시간 단축을 위한 경로계획방법을 제시하였다. 검사 클러스터 생성 단계와 카메라 이동순

서 결정 단계의 계층적 경로계획 방법을 채택하였고, 이 중 검사 클러스터 생성의 최적화를 위한 알고리즘들을 제시하였다. 검사기 클러스터링 문제는 전형적 클러스터링 문제와 목적함수 및 제한조건에 있어서 차별화 된다. 전형적 알고리즘으로부터 유도한 단일연결, ISODATA 알고리즘을 구현하였고, 새로운 S자형 알고리즘을 제안하였다. 국지적 해법의 한계를 극복하기 위하여 유전자알고리즘을 검사기 클러스터링 문제에 응용하였으며, 이를 위하여 염색체, 적합도 및 연산자들을 새로이 정의하였다.

시뮬레이션 결과, 전형적 클러스터링 문제에 좋은 성능을 갖는 단일연결, ISODATA 에 비하여 S자형 및 유전자알고리즘이 더 좋은 해를 생성시킴을 확인하였다. 특히 유전자알고리즘은 여러 가지 경우의 문제에 안정적으로 수렴되는 결과를 볼 수 있었다.

본 연구에서는 전체의 검사기 경로계획문제를 클러스터 생성 문제와 이동순서 결정문제로 분리하여 접근하였다. 이는 문제의 복잡도를 완화하여 용이하게 해를 구하고자 함이나, 결과적으로 해의 최적성이 저하될 수 있다. 추후 두 개의 문제를 하나로 통합하여 검사기 경로계획을 수행하는 연구를 진행할 예정이다.

**참 고 문 헌**

[1] 박태형, "전자조립용 CAM 시스템의 기술동향", 전자공학회지, 제26권 제3호, pp. 48-61, 1999.  
 [2] 최재완, 박경택, "비전 시스템을 이용한 검사 공정 자동화 기술 개발에 관한 연구", 한국자동차제어학술회의 논문집 제1호, pp. 128-130, 1994.  
 [3] T. S. Newman and A. K. Jain, "A survey of automated visual inspection", Computer Vision and Image Understanding, vol. 61, no. 2, pp. 231-261, 1995.  
 [4] 유창목, 차영엽, 김철우, 권대갑, 윤한중, "비전 시스템을 이용한 J-리드 납땜검사", 한국자동차제어학술회의, 제2호, 1169-1172, 1997.  
 [5] 오제휘, 차영엽, "비전 시스템에서 신경 회로망을 이용한 검사영역에 관한 연구", 제어자동화시스템공학회지, 제4권, 제3호, pp. 378-383, 1998.  
 [6] A. K. Jain, M. N. Murty, P. J. Flynn, "Data clustering: a review", ACM Computing Surveys, vol. 31, no. 3, pp. 264-323, 1999.  
 [7] B. S. Everitt, S. Landau, M. Leese, Cluster Analysis, Arnold, Fourth Edition, 2001.  
 [8] G. Reinelt, The Traveling Salesman: Computational Solutions for TSP Applications, Springer, 1994.  
 [9] M. Bellmore, G. Nemhauser, "The traveling salesman problem: a survey", Operation Research, vol. 16, pp. 538-558, 1968.

[10] E. Dahlhaus, "Fast parallel algorithm for the single link heuristics of hierarchical clustering", Parallel and Distributed Processing, IEEE Symposium on vol. 1, no. 4, pp. 184 -187, 1992.  
 [11] Z. Michalewicz, Genetic Algorithms + Data Structures = Evolution Programs, Springer, Third Edition, 1996.  
 [12] Lin Yu Tseng, Shiueng Bien Yang, "A genetic approach to the automatic clustering problem", Pattern Recognition vol. 34, pp. 415-424, 2001.  
 [13] 삼성 테크윈(주), "SMT 검사장비 AI400", <http://www.samsungtechwin.com>.

**저 자 소 개**



**김화중(Hwa-Jung Kim)**

2002년 : 충북대학교 제어계측공학과 졸업 (공학사)

2002년~현재 : 동 대학원 제어계측공학과 석사과정

관심 분야 : 최적화 알고리즘, 반도체 및 전자 조립 시스템, 영상처리.

E-mail : netjpg@korea.com



**박태형(Tae-Hyoung Park)**

1988년 서울대학교 제어계측공학과 졸업 (공학사)

1990년 동 대학원 졸업(석사)

1994년 동 대학원 졸업(박사)

1992년~1994년 제어계측신기술 연구센터 연구원

1994년~1997년 삼성항공산업(주) 정밀기 기연구소 선임연구원

2000년~2001년 Univ. of Toronto 방문 교수

1997년~현재 충북대학교 전기전자컴퓨터공학부 교수

관심분야 : 반도체 및 전자 조립 시스템, 로봇 동작계획 및 최적화 알고리즘

Phone : 043-261-3240

Fax : 043-268-2386

E-mail : taehpark@chungbuk.ac.kr