

홉필드 모델을 이용한 J-리드 검사 경로 생성

Path Planning of the J-lead Inspection using Hopfield Model

°이중호*, 차영엽**

*원광대학교 대학원 기계공학과, **원광대학교 기계설계공학과

(Tel: 0653-850-6693 ; Fax: 0653-850-6666 ; E-mail: ggypcha@wonnms.wonkwang.ac.kr)

Abstracts As Factory Automation is required, using the vision system is also essential. Especially, the path planning of parts with J-lead on PCB plays a important role of whole automation. Path planning is required because J-lead is scattered compared to L-lead on PCB. Therefore, in this paper, we propose path planning of part inspection with J-lead to use Hopfield Model(TSP: Traveling Salesman Problem). Then optical system suited to J-lead inspection is designed and the algorithm of J-lead solder joint and part inspection is proposed.

Keywords Factory Automation, J-lead, Hopfield Model, Path Planning.

1. 서론

공장 자동화가 가속화 됨에 따라 화상 처리를 이용한 검사 방법이 많은 관심을 끌고 있다. 이는 기존의 카메라를 이용한 인쇄 회로 기판(PCB, Printed Circuit Board)의 육안 검사로는 많은 종류의 불량을 시스템으로 반환 하는데 시간 지연이 많기 때문이다. 게다가 전자 제품의 다기능화와 소형화에 따라 부품이 소형화 되고, 이에 따른 인쇄 회로 기판의 고밀도화가 요구 되었으며, 이를 위해 고밀도화에 장점이 있는 J-리드를 갖는 칩의 사용이 증가하고 있는데, 이 리드가 칩의 안쪽으로 말려 납땜이 되어 있어 기존의 검사 방법을 사용할 수 없다. 이러한 J-리드를 갖는 칩의 검사 방법에 있어서 검사 경로 생성을 자동화 하는 것은 시스템의 전체적인 자동화를 이룬다는 점에서 중요하다.

본 논문에서 중점적으로 다룰 경로 생성의 문제에 대하여 Hopfield와 Tank[1]가 Hopfield Model을 적용한 TSP(Trabelling Salesman Problem)의 해결 방법을 처음으로 제시하였고, Carvalho와 Barbosa[2]는 Boltzmann Machine과 Simulated Annealing을 이용하였으며, Joppe와 Cardon[3]은 2개의 층(Layer)을 이용하여 유효한 식을 얻으려고 노력하였다. 특히 Aiyer, Niranjan, Fallside[4]는 고유치를 이용하여 10개에서 50개 까지의 도시에서도 유효한 값을 얻을 수 있는 식을 얻어냈다. 본 논문에서는 참고문헌[5]에서 제안한 WM-TS Problem 알고리즘을 이용하여 경로 생성을 하고, 이를 화상 처리 시스템에 적용하는 문제를 다루고, 그 구성으로는 2장에서 이 논문의 경로 생성 방법을 적용한 J-리드 납땜 검사 과정[6]을 간략히 설명하고, 3장에서 이 논문의 요점인 경로 생성에 대하여 Hopfield Model과 그 적용 방법을 함께 설명한다. 4장에서는 경로 생성과 이를 적용한 시스템에 대해서 시뮬레이션을 보여준다. 5장에서는 결론을 맺고 향후 연구 과제를 논의한다.

2. J-리드 납땜 검사 과정

2.1 납땜 검사 시스템의 구성

J-리드 납땜 검사에 사용한 검사 장치의 구성은 그림 1과 같이 영상 획득부와, 영상 처리부 그리고, 위치 제어부의 세 부분으로 간략화 할 수 있다. 영상 획득부는 카메라로 영상을 받아

영상 처리부로 보내는 역할을 하고, 영상 처리부는 받은 영상으로 J형 lead의 납땜 검사를 한다. 위치 제어부는 우리가 관심을 가지는 부분으로 검사할 위치로 이동 시키는 역할을 한다. 이때 검사 경로는 검사될 PCB의 data file에서 자료를 받아 TSP 알고리즘을 사용하여 생성하게 된다.

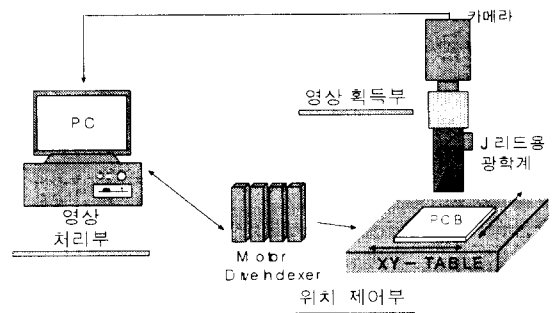


그림 1. 전체 시스템 구성도

Fig. 1. The schematic of total system

2.2 납땜 검사 알고리즘

그림 2의 순서도에서 보는 바와 같이 네 단계로 나누어진다. 1단계에서는 검사 알고리즘을 쉽게 적용하기 위하여 PCB의 부품 위치 데이터를 각 Frame별로 영상 좌표계로 변환하는 준비 파일을 생성하고, 2단계는 우리가 Hopfield Model을 적용할 부분으로 경로를 생성하고, 위치 제어부의 XY-table을 구동시켜 검사하고자 하는 J-리드의 위치가 되게 PCB를 이동시킨다. 그 경로 생성 알고리즘은 3장에 설명 된다. 3단계는 그 위치에서 카메라로부터 얻은 영상에서 특징값을 추출하여 검사영역을 결정한다. 특징값은 납땜의 형태를 정량적으로 표현한 것으로서 그 조건으로 동일한 class에 속하는 개체들의 특징값은 서로 유사한 값을 가져야 하며 서로 다른 class들은 값의 차이가 많아야 한다. 마지막으로 4단계에서는 부품검사와 납땜 상태 검사를 하게 된다.

부품 검사에는 부품의 유·무, 부품 이동 및 틀어짐, 부품 뒤집힘 등을 검사하고, 납땜 상태 검사에서는 리드와 패드 사이에 형성되는 양에 따라 납무, 납부족, 정상, 과납, 브릿지등을 검사한다.

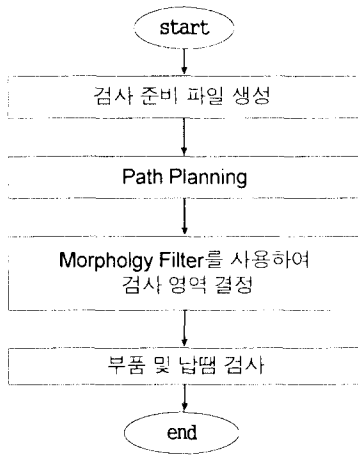


그림 2. 납땜 검사 순서도
Fig. 2. Proceeding of solder joint inspection

3. Hopfield Model을 이용한 경로 생성

3.1 Hopfield Model

Hopfield Model은 뉴런(Neuron)이라고 불리는 각각의 요소들이 피드백(Feedback) 없이 다른 요소들의 출력의 합으로 가중치를 정하는 구조로서 그림 3과 같이 나타낼 수 있다. 일반적으로 i 번째의 뉴런은 현재의 상태 또는 입력값 U_i , 그리고 그 출력값 V_i 로 표현한다.

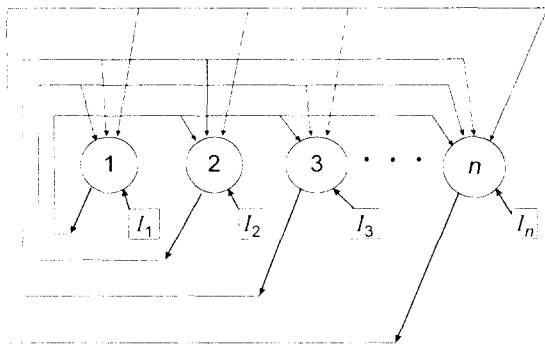


그림 3. Hopfield Model의 구조
Fig. 3. Schematic of the Hopfield Model

i 번째 뉴런 U_i 는 이 뉴런의 총 입력값에 대한 함수로 값이 갱신(Update)되며, 이 함수는 사용하는 값이 연속적(Continuous)인가? 아니면, 이산적(Discrete)인가? 에 따라 구별 되어진다. 이산적인 경우에는 Network의 거동 방정식을 정의할 때, 미분 방정식의 적용이 불가능하므로 연속적인 경우로 설정하여 다룬다. 연속적인 경우의 미분 방정식으로 표시되는 네트워크의 거동 방정식은 다음과 같다.

$$\frac{dU_i}{dt} = \sum_{j=1}^N T_{i,j} V_j - \frac{U_i}{\tau} + I_i \quad (1)$$

식(1)에서 τ 는 네트워크의 시간 상수(Time Constant)이며, 갱신을 위한 출력함수 $g(U_i)$ 는 쌍곡선 탄젠트 함수로 다음과 같이 표현 된다.

$$V_i = g(U_i) = \tanh\left(\frac{U_i}{U_s}\right) \quad (2)$$

3.2 Traveling Salesman Problem

TSP(Traveling Salesman Problem)란 세일즈맨이 여러곳의 도시를 순회할 때 각각의 도시를 단 한 번씩 방문하고, 처음 출발한 도시로 찾아오는 가장 짧은 경로를 찾는 문제로 이것을 J-리드를 갖는 부품의 검사 경로에 응용하면, N 개의 부품을 단 한 번씩 검사하는 가장 짧게 닫힌 연결 경로를 찾는 문제가 된다. 여기서 발생하는 경우의 수는 부품이 N 개라면 $N!/2.N$ 개의 경우의 수가 존재한다. Hopfield와 Tank는 이 문제 해결을 위해 경로를 생성할 부품의 수가 N 개라면, 뉴런의 수를 N^2 개로 설정하였다. N^2 개의 뉴런을 N 개의 뉴런에 대한 N 가지의 그룹으로 분류하고, N 개의 뉴런에 대한 각각의 그룹은 그 순번에 대한 부품으로 나타내었다. 그리고 $V_{x,i}$ 는 x 번째 부품에 대한 i 번째 순번을 나타내며, 만일 x 번째 부품이 i 번째 순번을 갖는다면, 그 값이 1이고 (즉 뉴런이 켜짐), 그렇지 않다면 0(뉴런이 꺼짐)이 된다. 이 조건에 만족하는 에너지 함수는 다음과 같다.

$$E = \frac{A}{2} \sum_{x=1}^N \sum_{i=1}^N \sum_{j=1, j \neq i}^N V_{x,i} V_{x,j} + \frac{B}{2} \sum_{i=1}^N \sum_{x=1}^N \sum_{y=1, y \neq x}^N V_{x,i} V_{y,i} + \frac{C}{2} \left(\sum_{x=1}^N \sum_{i=1}^N V_{x,i} - N \right)^2 + \frac{D}{2} \sum_{x=1}^N \sum_{y=1, y \neq x}^N \sum_{i=1}^N d_{x,y} V_{x,i} (V_{x,i-1} + V_{x,i+1}) \quad (3)$$

위 식에서 $d_{x,y}$ 는 x 부품과 y 부품의 거리를 뜻하며, A, B, C 그리고 D 는 임의의 비례 상수이다. 그리고 A 항은 각 부품을 한 번씩 연결하도록 하고, B 항은 단 한 개의 부품이 그 차례에 존재하도록 하고, C 항은 모든 N 개의 부품을 연결할 수 있도록 하며, D 항은 총 연결 거리가 최소가 되도록 한다. 네트워크의 거동 미분 방정식을 식(1)과 같은 형태로 표기하면

$$\frac{dU_{x,i}}{dt} = \sum_{j=1}^N \sum_{k=1}^N T_{x,i,j,k} V_{j,k} - \frac{U_{x,i}}{\tau} + I_{x,i} \quad (4)$$

$$V_{x,i} = g(U_{x,i}) = \frac{1}{2} (1 + \tanh(\lambda U_{x,i}))$$

이고, 여기서 λ 는 상수로 Gain 변수(Gain Parameter)라 한다. 식(3)과 식(4)에 의해 연결 행렬 $T_{x,i,j,k}$ 은 다음과 같다.

$$T_{x,i,j,k} = -A \delta_{x,y} (1 - \delta_{i,j}) - B \delta_{i,j} (1 - \delta_{x,y}) - C - D d_{x,y} (\delta_{i,i-1} + \delta_{i,i+1}) \quad (5)$$

연결 행렬의 거동 방정식에 양의 가중치를 주어 뉴런을 켜지게(Firing)하는 외부자극 $I_{x,i}$ 대해서

$$I_{x,i} = CN^2 \quad (6)$$

이다. 식(5)에서 $\delta_{i,j}$ 는 $\delta_{i,j} = \begin{cases} 1 & (i=j) \\ 0 & (i \neq j) \end{cases}$ 으로 정의되며, 식(6)에서 N^2 는 뉴런의 수 N 이 아닌 임의의 상수이다. 다음 쪽에서는 Hopfield, Tank와 식을 근거로 하여 Axel, Norman, Fallside[4]가 제안한 고유치를 이용한 TSP식을 설명한다.

3.3 Aier의 알고리즘

Aiyer, Niranjana, Fallside[4]는 연결 행렬의 고유치를 결정하고 그 고유벡터를 계산하기 위하여 D 항을 제거했다. 그 이유는 D 항의 값이 일정치 않고 식(3)의 처음 세 가지 항이 강한 제약 조건(Strong Constraint)이어서 D 항보다 네트워크가 유효한 값에 정착(Settle down)할 수 있게 하는데 영향을 많이 끼치기 때문이다. 또한, C1과 C2가 같은 조건의 항으로 값이 같기 때문에 $A=B$ 라 보고, $N' = N$ 이라 가정하면, 연결행렬 $T_{x,i,y,j}$ 는 다음과 같이

$$T_{x,i,y,j} = -A\delta_{x,y}(1 - \delta_{i,j}) - A\delta_{i,j}(1 - \delta_{x,y}) - C \quad (7)$$

이고, 연결행렬 $T_{x,i,y,j}$ 에 대한 고유치는 다음과 같다.

$$\lambda_1 = -CN^2 - 2A(N-1)$$

$$\lambda_2 = 2A$$

$$\lambda_3 = -A(N-2)$$

위의 세 가지 고유치의 각 요소들을 독립시키고, 최적의 수렴을 위해 A, B, C 그리고 N 의 값들을 조절하면, 유효한 값을 얻을 수 있다. 첫 번째로 C 항을 독립시키기 위해, 여분의 상수 $2A_1$ 을 각각의 연결 행렬에 대입 하고,

$$\lambda_1 = -CN^2 \text{ 이 되도록 각각의 연결행렬에}$$

$\frac{2A(N-1)}{N^2} + \frac{2A_1}{N^2}$ 을 대입하며, 그 고유치들을 정리하면, 다음과 같이 새로운 고유치를 얻을 수 있다.

$$\lambda_1 = -CN^2$$

$$\lambda_2 = -2(A - A_1)$$

$$\lambda_3 = -AN + 2(A - A_1)$$

식(7)의 연결행렬 $T_{x,i,y,j}$ 에 D 항을 포함시키고, 고유치에 의해 변형된 변수로 연결 행렬 $T_{x,i,y,j}$ 와 외부 자극 $I_{x,i}$ 를 정리하면

$$\begin{aligned} T_{x,i,y,j} = & -A\delta_{x,y}(1 - \delta_{i,j}) - B\delta_{i,j}(1 - \delta_{x,y}) \\ & - 2A_1\delta_{x,y}\delta_{i,j} - C \\ & + \frac{2(AN - A + A_1)}{N^2} \\ & - Dd_{x,y}(\delta_{j,i+1} + \delta_{i,i-1}) \end{aligned} \quad (8)$$

이고,

$$I_{x,i} = +CN \quad (9)$$

이다. 식 (8)과 식(9)에서 N 은 실제 부품의 수이며, B 는 항상 A 와 같다고 정한다.

고유치를 이용한 식은 많은 부품을 연결할 때 유효한 연결을 100%낼 수 있는 장점이 있지만, 거리의 항을 고려하지 않아 최적 거리를 갖는 연결을 하지 못한다.

3.4 WM-TS Problem

3.3절의 문제점을 해결하기 위하여 참고문헌 [5]는 거리항에 강한 제약조건을 주는 WMP식과 전체 부품간의 최적 연결에는 TSP식을 사용하여 WM-TS Problem (Weighted Matching-Traveling Salesman Problem)이라 하였다.

WMP란 한 부품이 다른 부품에 연결하여 이들의 최단거리를 찾는 문제로서, TSP식보다 상대적으로 간단하며 거리의 항에 강한 제약조건을 줄 수 있는 알고리즘으로 각 부품은 오직 다른 부품에만 연결되어야 하고, 연결된 거리의 합이 최소가 되어야 한다. 에너지 함수로 표현하면,

$$E = \sum_{x \in S} d_{x,y} V_{x,y} + \frac{\gamma}{2} \sum_x (1 - \sum_y V_{x,y})^2 \quad (10)$$

이고, 여기서 x, y 는 각각의 부품들을 나타내며, γ 는 Penalty 상수로 정의한다. Hopfield model에서는 $V_{x,y} = V_{y,x}$ 가 성립되므로 $x < y$ 인 경우만 고려하여 계산을 줄일 수 있다. 식(10)의 우변 첫 번째 항은 거리에 관한 항이고, 두 번째 항은 각 행렬 중 오직 하나 만이 연결 되어야 하는 제약조건을 나타낸다. 결합 부품간의 최적 연결을 위해 식(10)을 미분하면

$$dE_{x,y} = (d_{x,y} - \gamma + \gamma \sum_{k \neq y} V_{x,k} + \gamma \sum_{k \neq x} V_{y,k}) \cdot dV_{x,y} \quad (11)$$

이 되고, 각 뉴런의 거동에 관한 미분 방정식은

$$\frac{dU_{x,y}}{dt} = \sum_{n=1}^N \sum_{m=1}^N T_{x,y,n,m} V_{n,m} - \frac{U_{x,y}}{\tau} + I_{x,y} \quad (12)$$

$$V_{x,y} = g(U_{x,y}) = \frac{1}{2}(1 + \tanh(\lambda U_{x,y}))$$

와 같이 표현된다.

Hopfield Model의 U_i 가 비동기적으로 갱신되고, 연결 행렬

$T_{i,j}$ 가 대칭적이면, 식(1)로부터 유도되는 에너지 함수

$$E = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N T_{i,j} V_i V_j - \sum_{i=1}^N I_i V_i \quad (13)$$

과 같이 표현된다. 이 에너지 함수는 Liapunov 함수, 또는 목적 함수(Objective Function)라 하는데, 시스템은 언제나 감소 또는 일정한 상수로 수렴하는 방향으로 나아가는 특성이 있다. 그러므로 써 초기의 입력값이 점점 감소하여 에너지 최소점(Energy Minima)에 도달하면 일정 상수가 되고, 이 에너지 값이 우리가 원하는 해가 될 수 있다. 이 에너지 함수를 이용하여 i 번째 뉴런에 대한 거동 방정식을 표현하면

$$\frac{dU_i}{dt} = -\frac{U_i}{\tau} - \frac{\partial E}{\partial V_i} \quad (14)$$

와 같이 표현할 수 있으며, 식(12)를 위와 같은 형태로 하면

$$\frac{dU_{x,y}}{dt} = -\frac{U_{x,y}}{\tau} - \frac{\partial E}{\partial V_{x,y}} \quad (15)$$

로 표현할 수 있다. 식(11)을 식(15)에 대입하면, 미분 방정식에 관한 연결 행렬 $T_{x,y,n,m}$ 과 외부자극 $I_{x,y}$ 는

$$T_{x,y,n,m} = -\gamma(\delta_{m,x} + \delta_{x,n}) - \gamma(\delta_{y,n} + \delta_{y,m}) \quad (16)$$

$$I_{x,y} = \gamma - d_{x,y} \quad (17)$$

이다.

좀 더 빠르고 안정된 최적의 결함을 얻기 위하여 연결행렬에 각 점의 상호 연결 거리항 $\alpha(\gamma - d_{x,y,n,m})$ 을 새로 추가시켜 연결 행렬 $T_{x,y,n,m}$ 를 다시 정의하면

$$\begin{aligned} T_{x,y,n,m} = & -\gamma(\delta_{m,x} + \delta_{x,n} + \delta_{y,n} + \delta_{y,m}) \\ & + \alpha(\gamma - d_{x,y,n,m}) \end{aligned} \quad (18)$$

이다. 식(18)에서

$$d_{x,y,n,m} = d_{x,y} + d_{n,m}$$

$$\alpha = \begin{cases} 1 & (x \neq y, x \neq m, x \neq n, y \neq n, y \neq m) \\ 0 & (\text{그외의 경우}) \end{cases}$$

이다. α 는 상호 연결 변수라 정의한다.

전체 연결경로를 얻기 위하여 TSP식을 이용한다. TSP식은 고유치를 이용한 방법으로 구하고, 전체 부품간의 최적 연결에 알맞은 거리항을 만들기 위해 이론적 식(19), (20), (22)와 식(8)에서 각 항들의 A, B, C 그리고 D 등 값에 여러 비례 상수들을 적용하는 실험에 의하여 얻어진 식(21)을 사용한다.

$$\frac{dU_{x,i}}{dt} = \sum_{j=1}^N \sum_{l=1}^N T_{x,i,j,l} V_{j,l} - \frac{U_{x,i}}{\tau} + I_{x,i} \quad (19)$$

$$V_{x,i} = g(U_{x,i}) = \frac{1}{2}(1 + \tanh(\lambda U_{x,i})) \quad (20)$$

$$T_{x,i,y,j} = -2A\delta_{x,y}(1 - \delta_{i,j}) - 2B\delta_{i,j}(1 - \delta_{x,y}) - A_1\delta_{x,y}\delta_{i,j} - C + \frac{1(AN - A + A_1)}{N^2} - Dd_{xy}(\delta_{i,i+1} + \delta_{j,j-1}) \quad (21)$$

$$I_{x,i} = CN \quad (22)$$

4. 경로 생성의 적용

4.1 경로 생성 알고리즘

3.4절에서 설명한 알고리즘을 이용한 프로그램을 만들기 위해 다음과 같은 알고리즘을 사용하였다.

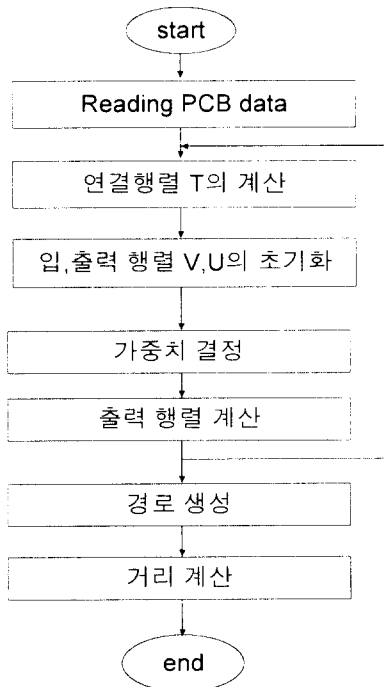


그림 4. 프로그래밍 알고리즘
Fig.4. Algorithm for programing

그림 4의 알고리즘은 1단계로 PCB의 데이터를 읽어 들이고, 2단계로 연결 행렬 $T_{x,y,u,m}$ 을 식(20)과 같이 계산한다. 3단계로 입,출력 행렬을 초기화하고, 4단계로 가중치를 계산한다(식(19)). 5단계로 출력 행렬을 계산하고, 2단계로 돌아간다. 이를 일정 횟수 만큼 반복시킨 뒤, 경로를 생성하고 거리 계산 후 끝낸다.

4.2 경로 적용

4.1절의 알고리즘을 사용한 프로그램에 J-리드가 있는 실제 PCB(그림 5.)를 사용하였다. 그림 5의 내부가 칠해진 부품이 J-리드이고 이들을 연결한 점선이 최적 경로이다.

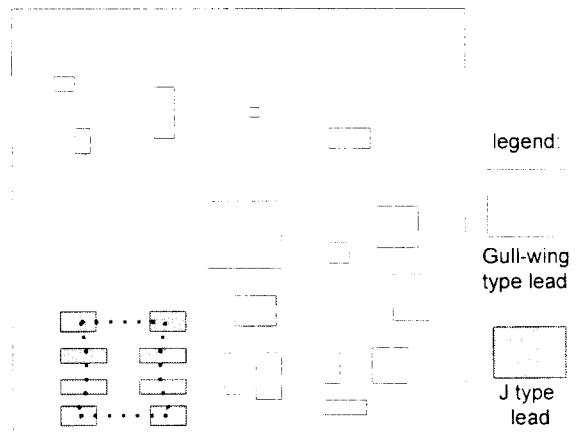


그림 5. 검사용 PCB
Fig. 5 PCB for inspection

5. 적용 결과 및 고찰

본 논문에서는 J-리드를 갖는 칩의 납땜 검사를 하기 위해 부품들 간의 최적 경로를 찾는 TSP에 대해 알아보고 이를 실제 시스템에 하는 것을 목적으로 하였다.

많은 실험 결과 그림 5와 같은 최적 경로가 항상 나타나는 것은 아니었지만 완전한 경로를 거의 대부분의 경우에 찾을 수 있어 실제 시스템에 적용할 수 있었다. 앞으로 J-리드용 광학계(그림 1.)의 특성을 고려한 연구가 폭넓게 진행 되어야 할 것이다.

참고문헌

- [1] Freemam, J.A. and Skapura, D.M., "Neural Networks", Addison-Wesley Pub., pp. 141-168, 1992.
- [2] Carvalho, L.A.V. and Barbosa, V.C., "A TSP Objective Function that Ensures Feasibility at Stable Points", in Proc. INNC90, vol. 1, pp. 249-253, 1990
- [3] Kung, S.Y., "Digital Neural Networks", Prentice Hall, pp. 321-325, 1993
- [4] Aiyer, S.V.B., Niranjana, M. and Fallside, F., "A Theoretical Investigation into the Performance of the Hopfield Model", IEEE Trans. N. N, vol. 1, no. 2, pp. 204-215, 1990
- [5] 오제휘, 차영엽, "신경 회로망을 이용한 부품 조립 공정의 최적화 알고리즘", 한국자동제어 학술대회 논문집(1), pp.143-146, 1996
- [6] 김철우, "비전 시스템을 이용한 PCB 납땜 검사 장치의 개발", 1차년도 최종 보고서, 한국과학기술원(LG전자 디지털 계측실), 1996년.