

고해상도 비전 모듈시스템 개발

장홍식, 박한길, 김형남*, 조상복
 (주)STI, 울산대학교 전기전자 및 자동화공학부
 man2man@hitel.net, stinc@chollian.net*, sbcho@uou.ulsan.ac.kr

Development of a high resolution vision module system

Hong Sik Jang Han Kil Park Hyung Nam Kim* Sang Bock Cho
 STI Inc.*
 532-2, Uchon-ri, Kumwang-up, Umsung-gun, Chungbuk, Korea*
 Dept. of Electronic Engineering
 Mugue 2 Dong, Namgu, Ulsan, Korea

요약

본 논문에서는 리드 피치가 15mil 인 208 핀 QFP 칩을 검사하는 고해상도의 비전 모듈 시스템을 개발하였다. 전체적인 하드웨어와 고해상도에 적합한 알고리즘 및 이를 실현하는 프로그램을 개발하였으며, 특히 칩 리드 검출을 위한 새로운 알고리즘을 제안하였다. 본 논문에서 제안한 방법의 특징은 조명의 간섭에 대해 강하며 검사 속도 및 정밀도가 향상되었다.

1. 서론

고기능의 전자회로의 급속한 발전으로 대규모의 시스템을 한 칩에 담을 수 있는 반도체 칩의 종류와 규모와 형태가 다양화되고 있다. 특히 패키징에 있어 리드간격(pitch)이 줄어들면서 소형화되는 추세이며, 이렇게 소형, 경량화되어 가면서 사람의 육안으로 부품의 위치, 방향 및 리드의 불량 유무 등을 검사하고 판정하는 것은 시간과 정밀도 면에서 한계가 있으므로 생산성 향상을 위하여 비전 모듈 시스템의 개발이 꾸준히 요구되어 왔다. 반도체 칩을 설계하여 제작하고 난 뒤 칩의 패키징 형태를 테스트하는 것은 매우 중요하다. 아무리 바른 설계와 칩 제작과정을 거쳐 제작되었다 할지라도 칩의 패키징이 제대로 되지 않으면 정상동작을 행하기 힘들기 때문이다. 최근 컴퓨터 시스템의 급속한 발전으로 반도체 부품의 자동조립과 검사에 있어 시각 시스템의 필요성이 증가하고 있으며, 이에 대한 연구가 진행되어왔다. 기존에 제안된 방법으로는 반도체 소자의 위치 및 방향

을 탐지하고 물체의 관성 모멘트를 이용하여 물체의 중심 및 기울어짐을 찾는 방법이 있으며, 이 방법은 소자에 해당하는 모든 화소에 대하여 1 차 및 2 차 모멘트를 구하여 관성주축을 지나는 직선을 찾는 방법이고, Hom 이 제안한 투영 기법은 물체를 x 축, y 축, $y=x$ 에 각각 투영시킨 결과로부터 물체의 중심점 및 방향을 찾는 방법이다[1,2]. 앞에서 설명한 방법들은 칩의 영상들은 LED 확산 조명을 반도체 칩의 후방에서 비춰 그림자 영상 즉 투영 영상을 취득하여 칩 리드의 불량을 검사하므로 검사 항목에 있어서 신뢰성이 다소 떨어진다. 이러한 칩을 검사하는 비전 모듈 시스템은 칩의 형태에 따른 검사의 신뢰성이 핵심적인 사항이라 하겠다.

본 연구에서 제안한 방법은 QFP 반도체 칩의 4 면 영상을 모두 습득하여 제안된 알고리즘으로 칩 리드의 불량을 검사하므로 기존의 방법에 비해 검사 항목의 신뢰성이 증가한다. 리드간격이 15mil 단위와 같은 고해상도 시스템을 요구하는 칩을 종래의 방법으로 실시간 검사하는 것은 부적합하며 새로운 알고리즘과 그에 맞는 프로그램이 필요하다. 따라서 본 연구에서는 15mil 단위의 고해상도로 대규모의 칩을 테스트하는 고해상도의 비전 모듈 시스템을 개발하였다.

향후 칩의 형태가 다양화되고 규모가 커지며 15mil 이하의 고밀도 패키징이 일반화됨에 따라, 이러한 고해상도의 비전 모듈 시스템의 사용은 필수적이며 계속 증가하리라 생각된다.

2. 비전 모듈 시스템 구성

비전 모듈 시스템은 반도체 칩의 각 면의 리드 영상을 각각 취득할 수 있는 하드웨어와 취득한 영상을 처리하여 빠른 시간 내에 리드의 불량 유무 등을 검사할 수 있는 소프트웨어로 구성되어있다.

2.1 비전 모듈 시스템 하드웨어

본 연구에서 사용한 비전 모듈 시스템 하드웨어는 그림 1과 같이 칩 인스펙터 검사 소프트웨어를 탑재한 PC로 구성되어있다. 칩 인스펙터는 광원, 반사경, 디지털 카메라, 전송케이블로 구성되어 있어 칩의 측면 영상을 디지털 카메라에서 취득하여 PC로 전송한다.[4]

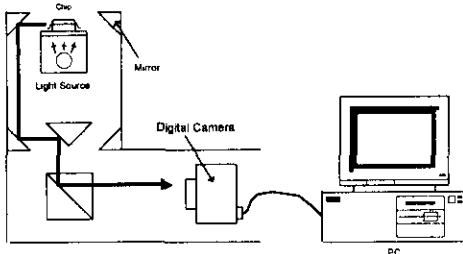


그림 1. 비전 모듈 시스템 하드웨어

2.2 비전 모듈 시스템 소프트웨어

디지털 카메라에서 취득한 칩의 측면 영상은 비전 모듈 시스템 소프트웨어 프로그램(V MSS)에서 칩 리드의 불량 검사를 항목 별로 실시하며 화면에 표시되는 형태는 그림 2와 같다.

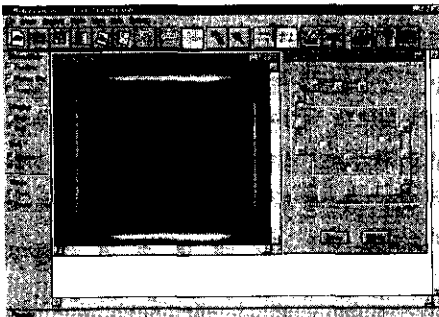


그림 2. 프로그램(V MSS)의 전체 화면

3. 제안된 알고리즘

3.1 칩 리드 경계 알고리즘

기존의 칩 리드의 경계를 검출하기 위해 흑백 농도의 문턱값을 이용하여 경계를 검출하고 리드를 찾는 방법을 사용하였다. 그러나 리드의 피치가 20mil 정도인 칩에서 기존의 방법을 적용하는 경우, 리드의 경계를 검출하여 리드 폭을 결정하면 오차가 상당히 크게 발생하였다. 리드의 피치가 작을수록 리드와 리드 사이의 그레이 농도의 분포가 완만한 값을 가지므로 그 문턱농도값을 정하기란 매우 어렵고, 디지털 카메라로부터 습득한 칩의 이미지에서 칩 가운데 부분과 양쪽 끝부분의 밝기가 서로 다르므로 문턱값을 하나로 결정하기란 매우 어렵다. 또한 가장 큰 문제점은 그레이 농도 문턱값이 라이트 소스의 세기에 따라 크게 변하므로 빛의 세기를 높이면 문턱값이 높아지고, 빛의 세기를 낮추면 문턱값이 전반적으로 낮아지는 현상을 보이고, 칩의 가운데 부분과 양쪽 끝부분에서 빛의 세기가 서로 다르므로 그레이 농도 문턱값도 역시 서로 다르다. 따라서, 칩 리드의 피치가 20mil 이하로 줄어들고 칩의 가운데 부분과 양쪽 끝부분에서의 빛의 밝기가 서로 다른 경우에 빛의 밝기에 영향을 덜 받으면서 가장 효과적으로 리드의 경계를 검출하고 리드의 폭을 결정할 수 있는 새로운 알고리즘이 필요하다. 따라서, 본 연구에서 칩 리드의 경계를 검출하기 위해 제안된 알고리즘은 그레이 농도값의 기울기를 이용하여 경계를 검출하고 리드의 폭을 결정하는 방법이다. 서치원도를 따라 칩 리드의 픽셀 농도값을 각각 조사해 보면 그림 3과 같은 농도분포가 나타난다.

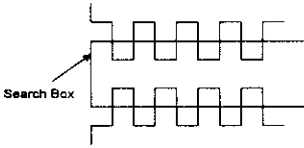


그림 3 (a). 칩 리드 이미지

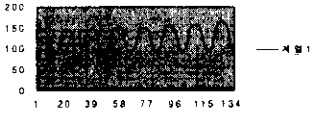


그림 3(b). 리드의 그레이 농도분포

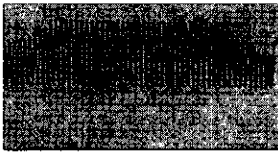


그림 3 (c). 칩 전체 그레이 농도 분포도

칩의 이미지에서 빛의 밝기가 고르지 못하므로 일정한 그레이 농도 분포를 정하는 것은 어렵기 때문에 리드의 경계를 결정할 때 농도의 기울기가 가장 가파른 부분을 경계로 선택하는 것이다. 서치박스 좌측상단 픽셀에서 농도값을 읽어서 pix 에 저장시키고 리드의 폭만큼 픽셀의 수를 증가시킨다. 이 때 농도차 $Thresh_diff$ 는 pix 와 pre_pix 의 차로서 정의되고, $Thresh_diff$ 가 Th_Max 라는 값보다 크면 현재 픽셀 x 좌표값 i 가 $enter_edge_x$ 변수에 저장된다. Th_Max 는 현재의 $Thresh_diff$ 로 갱신되고 pre_pix 역시 현재의 pix 로 갱신된다. 이런 과정을 반복하게 되면 $enter_edge_x$ 에 저장되어 있는 좌표값은 결국 그레이 농도차가 가장 큰 값이며 이 좌표가 바로 리드의 경계가 되는 것이다. 리드의 검사항목을 결정하는데 필요한 좌표들은 그림 4 에 나타내었고, 이들을 구하기 위한 알고리즘은 다음과 같다.

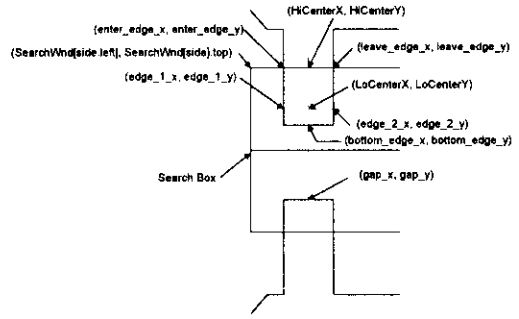


그림 4. 검사항목에 필요한 좌표들

1. $(enter_edge_x, enter_edge_y)$ 좌표구함
2. $(leave_edge_x, leave_edge_y)$ 좌표구함
3. $(HiCenterX, HiCenterY) = ((enter_edge_x + leave_edge_x)/2, (enter_edge_y + leave_edge_y)/2)$
4. $(HiCenterX, HiCenterY)$ 에서 아래 방향으로 탐색하여 리드의 끝을 검색한다.
5. $(bottom_edge_x, bottom_edge_y)$ 좌표구함
6. $(bottom_edge_x, bottom_edge_y)$ 좌표에서 $+y$ 방향으로 2 픽셀 증가
7. 좌측으로 좌표값을 감소시켜 리드 왼쪽 경계 검출하여 $(edge_1_x, edge_1_y)$ 좌표구함
8. 6 번에서 우측으로 좌표값을 증가시켜 리드의 오른쪽 경계 검출하여 $(edge_2_x, edge_2_y)$ 좌표구함
9. $(LoCenterX, LoCenterY) = ((edge_1_x + edge_2_x)/2, (edge_1_y + edge_2_y)/2)$
10. 위에서 구한 좌표들에 대하여 slope 보정을 한다.
11. $(bottom_edge_x, bottom_edge_y)$ 좌표에서 $-y$ 축 방향으로 증가시켜가며 허상의 리드 끝 부분을 검출하여 (gap_x, gap_y) 좌표를 구한다.

3.2 리드 검사 항목 및 리드 검사 코드

칩 리드의 검사항목은 Coplanarity, Pitch, Width, Skew, Bur, Stand off, Length Deviation, Terminal Device 등 총 8 개가 있다.

4. 결론

반도체 칩 리드의 불량 유무 등을 검사하고 판정하는 비전 모듈 시스템을 개발하였다. 본 연구는 하드웨어와 소프트웨어의 개발의 2 가지형태로 나누어져서 진행이 되었다.

첫째가 반도체 칩 리드의 실제 이미지를 디지털 카메라로 찍을 수 있는 하드웨어의 개발이었다. 하드웨어의 개발에서는 가장 선명한 이미지를 얻을 수 있도록 하는데 중점을 두고 진행이 되었다. 또한 하드웨어의 크기와 무게를 작고 가볍게 하기위해서 여러가지 사항들이 고려되었다.

둘째로 하드웨어에서 찍은 이미지를 받아서 리드의 불량여부를 결정하는 소프트웨어의 개발이었다. 기존에는 칩 리드의 경계를 검출하기 위해 흑백농도의 문턱값을 이용하여 경계를 검출하고 리드를 찾는 방법을 사용하였다. 그러나, 리드의 피치가 작을수록 리드와 리드 사이의 그레이 농도의 분포가 완만한 값을 가지므로 그 문턱농도값을 정하기란 매우 어렵고, 디지털 카메라로부터 습득한 칩의 이미지에서 칩 가운데 부분과 양쪽 끝 부분의 밝기가 서로 다르므로 문턱값을 하나로 결정하기란 매우 어렵다. 또한 가장 큰 문제점은 그레이 농도 문턱값이 라이트 소스의 세기에 따라 크게 변하므로 빛의 세기를 높이면 문턱값이 높아지고, 빛의 세기를 낮추면 문턱값이 전반적으로 낮아지는 현상을 보이고, 칩의 가운데 부분과 양쪽 끝 부분에서 빛의 세기가 서로 다르므로 그레이 농도 문턱값도 역시 서로 다르다. 따라서 칩의 가운데 부분과 양쪽 부분에서의 빛의 밝기가 서로 다른 경우에 빛의 밝기에 영향을 덜 받으면서 가장 효과적으로 리드의 경계를 검출하고 리드의 폭을 결정할 수 있는 새로운 알고리즘이 필요하다. 따라서, 본 연구에서 칩 리드의 경계를 검출하기 위해 제안된 알고리즘은 그레이 농도값의 기울기를 이용하여 경계를 검출하고 리드의 폭을 결정하는 방법이다.

이 알고리즘을 이용하여 실험을 해본 결과는 기존의 알고리즘이 가지고 있던 문제점들을 해결할 수 있었

으나, 리드를 찾아서 그 경계를 결정하는 부분에서 문제점이 나타나게 되었다. 디지털 카메라에서 읽어 오는 그레이 값이 항상 일정하지가 않아서 경계가 항상 일정하게 나오지 않았다. 이러한 이유로 1mil 이상의 오차가 발생하게 되었다. 이 오차를 줄이기 위해서 Interpolation 과정을 거치게 오차를 약 0.5mil 정도로 줄일 수가 있었다.

앞으로의 과제로는 하드웨어부분은 더욱 더 선명한 이미지를 얻을 수 있도록 거울과 칩 사이의 거리를 줄이며 작고, 가볍게 만드는 것이다. 그리고 소프트웨어부분은 새로운 Interpolation 알고리즘을 개발해서 오차를 0.5mil 이하로 줄이는 것이다.

참고문헌

- [1] B. K. P. Horn, "Robot Vision", Cambridge, MA, MIT Press, pp.46-61, 1988
- [2] 차국찬, 박일수, 최종수 "SMD 에서 정합 및 부품 검사 알고리즘에 관한 연구", 전자공학회논문지, B 편 29 권 제 1 호, pp67-84, 1992 년 1 월
- [3] Randy Crane, "A simplified approach to Image Processing", PTR
- [4] Kodak Megaplug Remote Panel Software Users Manual