

비전 시스템을 이용한 2-D 원형 물체 추적 알고리즘의 비교에 관한 연구

한규범*, 김정훈**, 백윤수***

A Study on the Comparison of 2-D Circular Object Tracking Algorithm Using Vision System

Kyu Bum Han*, Jung Hoon Kim**, and Yoon Su Baek***

ABSTRACT

In this paper, the algorithms which can track the two dimensional moving circular object using simple vision system are described. In order to track the moving object, the process of finding the object feature points - such as centroid of the object, corner points, area - is indispensable. With the assumption of two-dimensional circular moving object, the centroid of the circular object is computed from three points on the object circumference. Different kinds of algorithms for computing three edge points - simple x directional detection method, stick method, T-shape method are suggested. Through the computer simulation and experiments, three algorithms are compared from the viewpoint of detection accuracy and computational time efficiency.

Key Words : Dynamic Scene(동영상), Centroid Detection Algorithm(중심점 검출 알고리즘), 2-D Circular Object Tracking(2 차원 원형 물체 추적)

1. 서론

비전 시스템의 특징은 구조화되지 않은 주위 환경(unstructured environment)으로부터의 시각 정보를 이용하여 원하는 작업에 필요한 정보를 얻어 내는 데에 있다. 이러한 비전 시스템은 위치가 알려지지 않은 목표 물체(target object)에 대하여 동작하는 로봇 매니플레이터의 제어 시스템^[1-3], 감시 시스템(inspection system)^[4-5], 자율 이동 로봇 시스템^[6-8] 등 많은 응용 분야에 적용될 수 있다.

비전 시스템은 크게 비전 센서(CCD)로부터 얻

어지는 영상의 특성에 따라 정적 영상 시스템과 동적 영상 시스템으로 구분할 수 있으며, 이중 동적 영상을 대상으로 하는 시스템은 영상의 정확한 해석 뿐만 아니라 영상 처리 속도에도 큰 영향을 받으므로 정적 영상 시스템에 비하여 상대적으로 연구가 뒤쳐진 분야이다^[9-10].

이러한 동적 영상 시스템의 연구에 있어서 이동하는 물체에 대한 정보를 얻어내기 위해서는 영상내의 특징 요소(feature)를 검출하는 과정이 중요하다. 특징 요소로는 보통 물체 내의 구멍, 물체의 코너점(corner point), 중심점(centroid) 등이 사용되

* 연세대학교 기계공학과 대학원

** 한국과학기술원 기계공학과

*** 연세대학교 기전공학부

며 이러한 특징 요소의 해석을 통하여 이동 물체의 동적 특성을 파악하게 된다. 이러한 특징 요소 중 물체의 중심점은 정적 영상의 경우 영상 전체에 대하여 1차 모멘트를 계산함으로써 쉽게 계산할 수 있으나^[10-11] 이 경우 계산량이 크게 증가하므로 동영상의 해석에는 적합하지 않다.

따라서 본 연구에서는 물체의 형상이 원형이라는 가정 하에 이동 물체의 에지(edge) 상에 존재하는 세 개의 점을 계산함으로써 중심점을 계산하였다. 실제로 현재 산업 현장의 비전 시스템을 이용한 부품 검사 등의 공정에서 구멍 등의 원형 물체 인식이 상당 부분을 차지하고 있으며 이에 따라 본 논문에서는 이동하는 원형 물체의 추적에 중점을 두었으나, 임의의 정형화된 물체의 형상에 대해서도 약간의 알고리즘의 수정으로 적용이 가능하다.

본 연구에서는 이러한 세개의 에지점 검출을 위하여 세 종류의 알고리즘을 제시하고 컴퓨터 시뮬레이션과 간단한 X-Y 테이블로 구성된 비전 시스템의 실험으로부터 각 알고리즘의 특성을 검출 정확도와 계산 속도의 효율성의 관점에서 비교하였다.

2. 원형 물체의 중심점 검출 알고리즘

비전 시스템을 이용하여 원형 물체의 중심점을 계산하기 위해서는 최소한 3 개의 원주상에 존재하는 에지점의 검출이 필요하다. 에지점을 검출하는 방법으로는 Robert 오퍼레이터, Sobel 오퍼레이터, Canny edge detector 등을 사용하는 방법들이 있으며^[10-11] 본 연구에서는 이중 3×3 크기의 Sobel 오퍼레이터를 통해 계산된 gradient 값을 이용하여 에지점을 검출하였다. 영상 평면상에서 검출된 에지점의 위치를 각각 (x_1, y_1) , (x_2, y_2) , (x_3, y_3) 라고 하면 영상 평면에서의 원형 물체의 중심점 (c_x, c_y) 는 다음 식(1), (2)와 같이 구해진다.

$$c_x = \frac{e \times d - b \times f}{\det} \quad (1)$$

$$c_y = \frac{a \times f - e \times c}{\det} \quad (2)$$

여기서 각 계수들은 다음과 같다.

$$\begin{aligned} a &= 2(x_1 - x_2) \\ b &= 2(y_1 - y_2) \\ c &= 2(x_2 - x_3) \\ d &= 2(y_2 - y_3) \\ e &= x_1^2 - x_2^2 + y_1^2 - y_2^2 \\ f &= x_2^2 - x_3^2 + y_2^2 - y_3^2 \\ \det &= ad - bc \end{aligned}$$

다음으로 각 알고리즘에 의하여 원형 물체 원주상에 존재하는 세개의 에지점을 구하는 과정을 간단히 설명한다. 여기서 중요한 사항은 이동 물체를 효과적으로 추적하기 위해서는 특히 계산 속도가 빨라야 하며 따라서 반복적인(iterative) 방법보다는 순차적인(sequential) 방법에 의하여 알고리즘이 구성되어야 한다는 점이다.

X 방향 단순 검출법(simple x directional detection method)

이 방법은 단순히 3×3 크기의 Sobel 오퍼레이터를 이용하여 전체 이미지 평면에 대하여 좌 상단으로부터 우 하단의 방향으로 순차적 컨볼루션 적분한 후 영상 픽셀의 gradient 값을 계산하여 에지점을 검출하는 방법이다. 이때 영상 평면상의 y 축에 대하여 Δy 만큼 증가시켜가며 세개의 에지점을 검출한다. 이 경우 비교적 알고리즘이 간단히 구현될 수 있다는 장점을 가지고 있으나 Fig. 1에서 볼 수 있듯이 점 1과 점 2의 간격이 너무 좁으면 정확한 중심점을 찾을 수 없고, Δy 의 크기가 너무 커지면 3 개의 에지점 검출이 실패할 가능성이 있다는 단점을 갖는다. 또한 탐색 시간이 오래 걸리므로 추적 물체의 속도가 큰 경우 제한을 가지게 되고 따라서 실시간 추적에는 적합하지 않다.

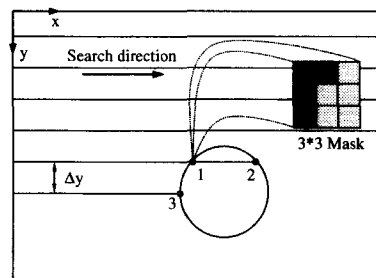


Fig. 1 Edge detection using simple x directional method

Stick Method^[12]

이 방법은 본래 임의의 형상을 가진 물체의 형상을 검출하기 위한 방법으로 고안된 방법이지만 이 역시 이동 물체의 실시간 추적에는 알고리즘의 처리 속도에 한계를 가지므로 본 연구에서는 추적 대상 물체가 원형이라는 가정하에 빠르고 정확하게 중심점을 찾을 수 있도록 수정하였다.

우선 x 방향 단순 검출법과 마찬가지로의 방법으로 최초의 에지점 1을 찾은 후, 그 점을 중심으로 반지름 r인 원호를 따라서 반시계 방향으로 0도에서 360도까지 회전시켜 두 번째 에지점 2를 검출하고, 같은 방법으로 두 번째 에지점을 중심으로 반지름 r인 원호를 회전시켜 세번째 에지점 3을 검출한다. 이 방법은 x 방향 단순 검출법에 비하여 Δy의 크기를 증가시킴으로 계산 속도를 빠르게 할 수 있다는 장점을 갖는다. 그러나 stick 반지름 r값이 너무 크게 설정되면 에지점 검출이 실패할 수 있고, 반면에 r값이 너무 작게 설정되면 세 개의 에지점이 너무 집중되어 정확도가 떨어진다 단점을 가지므로 적절한 r값의 선정이 매우 중요하다.

Fig. 2(a)는 위에서 설명한 stick method에 의한 에지점의 검출법을 보여주며, Fig. 2(b)는 (a)의 방법을 수정한 형태로 r의 회전방향을 두번째 에지점 검출과 세번째 에지점 검출에서 다르게 설정하여 에지점의 집중을 방지하도록 한다.

T-shape method

앞의 방법들이 가진 단점들을 보완하기 위해 제안된 방법으로 다음과 같은 과정을 거친다.

단계 1 : X 방향 단순 검출법에 의한 방법과 마찬가지로 에지점 1과 2를 검출한다. 여기서 각 에지점의 위치는 각각 (x₁, y₁), (x₂, y₂)이고 L은 두 에지점간의 거리이며 r은 미리 설정한 파라미터 값이다.

단계 2 : 에지점 1과 에지점 2의 수직 이등분상의 두점 a와 b를 찾고 두점 중 어느 점이 물체상의 점인지 판별한다. 여기서 점 a, b의 위치는 다음 식(3)과 같이 설정한다. 또한 두점 중 물체 내부의 점을 a, 물체 외부의 점을 b로 한다.

$$a, b \rightarrow [x_1 + L/2, y_1 \pm r \sin \theta] \quad (3)$$

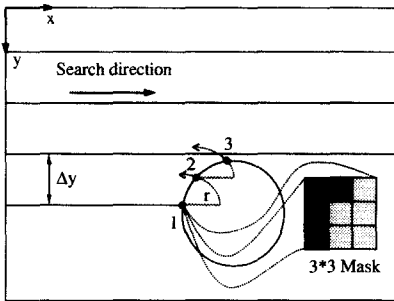
$$\theta = \cos^{-1}(L/2r)$$

단계 3 : 점 a로부터 수직 방향으로 탐색하여 에지점 3을 검출한다. 이때 수직 방향은 점 b에서 점 a의 방향으로 결정한다.

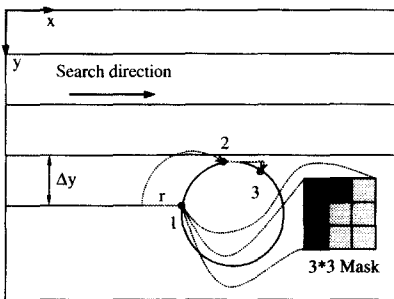
단계 4 : L과 r을 비교하여 L ≥ r인 경우 Fig. 3(a)에서 나타낸 바와 같이 에지점 1, 2, 3을 이용하여 중심점을 검출한다. L < r인 경우에는 단계 5를 수행한다.

단계 5 : 에지점 1과 에지점 3의 y 방향 평균점으로부터 x 방향으로 탐색하여 새로운 에지점 4를 검출한다. 최종적으로 에지점 1, 3, 4를 이용하여 중심점을 계산한다.

이 방법은 원형 물체의 정확한 중심점 계산시 오차의 중요한 원인이 되는 검출된 에지점들간의 집중화를 방지할 수 있다는 장점을 가진다. 또한 stick method와 마찬가지로 Δy의 크기를 x 방향 단순 검출법에 비해 크게 설정할 수 있고 따라서



(a) Original stick method



(b) Modified stick method

Fig. 2 Edge detection using stick method

계산 속도를 증가시킬 수 있다는 장점이 있다. 그러나 이 방법은 탐색 물체가 원형이라는 제약이 있기 때문에 보다 일반적인 물체 탐색에 적용될 수 있는 stick method에 비해 상대적으로 단점을 가진다.

다음 Fig. 3은 T-shape method에 의한 에지점의 검출 과정을 보여주고 있다.

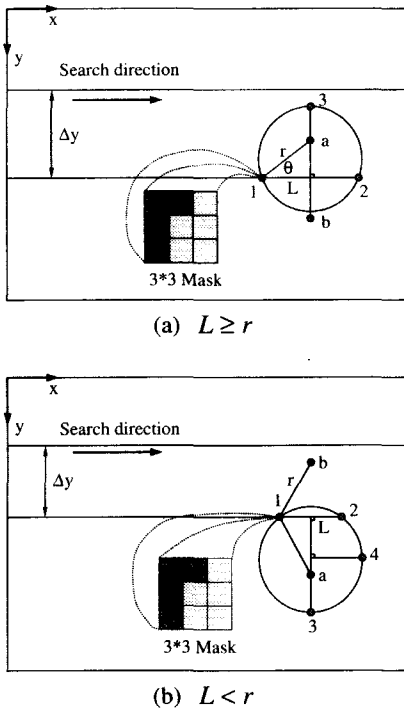


Fig. 3 Edge detection using T-shape method

3. 컴퓨터 시뮬레이션

앞서 제시한 알고리즘들을 비교하기 위해 우선 컴퓨터 시뮬레이션을 통해 알고리즘들을 정확도와 계산 속도의 측면에서 비교하였다.

시뮬레이션을 위한 영상은 620×480 이미지 크기의 정영상 이미지이며 PC 스크린 상에서 원형 물체의 지름을 30~70 pixel로 변화시켜가며 중심점을 계산하였다. 단순 x 방향 검출법은 특성상 Δy가 다른 방법들에 비하여 작게 설정되어야 하므로 원 지름의 0.45 배로 설정하였으며 나머지 두 방법에 있어서는 각각 원 지름의 0.6 배와 0.9 배의 크

기로 설정하였다. 또한 stick method에 대해서는 stick 반지름 r을 원 반지름의 √3 배로 설정하여 검출되는 에지점 3개가 원주상에 고르게 분포되도록 하였다.

다음 Fig. 4는 각 계열에 대한 평균 검출 오차를 보여준다. 시뮬레이션 시행 횟수는 150 회이며 각 계열의 의미는 다음과 같다.

- Series 1 : simple x directional detection method (Δy = 0.45 × diameter of circle)
- Series 2 : stick method(modified) (Δy = 0.6 × diameter of circle)
- Series 3 : stick method(modified) (Δy = 0.9 × diameter of circle)
- Series 4 : T-shape method (Δy = 0.6 × diameter of circle)
- Series 5 : T-shape method (Δy = 0.9 × diameter of circle)

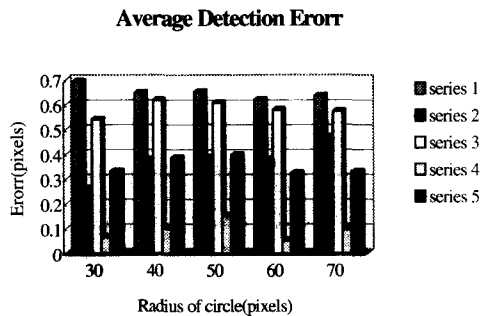


Fig. 4 Average detection error

Fig. 4에서 평균 검출 오차 E는 스크린 상의 원의 실제 중심점 (x₀, y₀)와 검출된 세 개의 에지점으로부터 계산된 중심점 (x, y)사이의 거리를 의미하며 다음 식 (4)에 의하여 계산된다.

$$E = \sqrt{(x - x_0)^2 + (y - y_0)^2} \quad (4)$$

위의 Fig. 4의 결과로부터 세 가지 방법 중 T-shape method에 의하여 가장 정확한 값을 얻을 수 있으며 단순 x 방향 검출법에 의한 결과의 오차가 가장 크게 나타나고 있음을 알 수 있다. 다음 Fig.

5는 각 계열에 따른 평균 검출 시간을 비교한 결과를 보여준다.

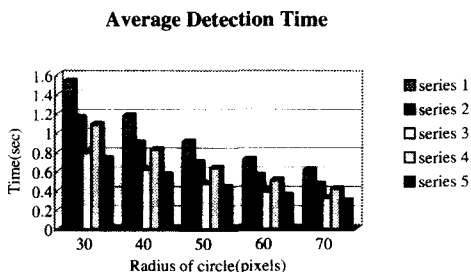


Fig. 5 Average detection time

Fig. 5로부터 원형 물체의 반지름이 커질수록 세가지 방법 모두에 대하여 계산 속도가 빨라짐을 관찰할 수 있으며, 계산 속도의 측면에서는 T-shape method의 경우가 가장 계산 속도가 빠름을 보여 주고 있다. 또한 stick method와 T-shape method로부터 Δy 를 크게 할수록 계산 속도가 빨라짐을 알 수 있다.

두가지 시뮬레이션의 결과로부터 얻을 수 있는 결론은 다음과 같다. 첫 번째로 검출의 정확성과 검출 속도의 측면에서 T-shape method와 stick method가 x방향 단순 검출법에 비하여 더 좋은 결과를 줌을 알 수 있다. 두 번째로 T-shape method와 stick method의 경우 Δy 가 커질수록 계산 속도는 빨라지나 오차도 같이 증가함을 관찰할 수 있다.

이러한 관점에서 살펴볼 때, Δy 가 원 지름의 0.9 배로 설정된 T-shape method는 Δy 가 원 지름의 0.6 배로 설정된 stick method와 비슷한 오차를 가지면서 검출 속도는 약 63% 정도가 걸린다는 점에서 T-shape method가 가장 우수하다는 결론을 얻을 수 있다.

4. 실험 및 결과고찰

앞 절에서 구현된 알고리즘을 이용하여 실제 시스템에서의 추적 성능을 관찰하기 위하여 Photo 1.과 2.에서 보여주는 실험 장치를 구현하였다.

실험 장치는 크게 X-Y 테이블과 이에 부착된 카메라로 구성되는 구동부(driving unit)와 구동부에

수직으로 위치하여 2차원 평면 상에서 회전 이동하는 원형 물체를 구현하는 원반형 경로 생성부(path generator)로 구성된다.

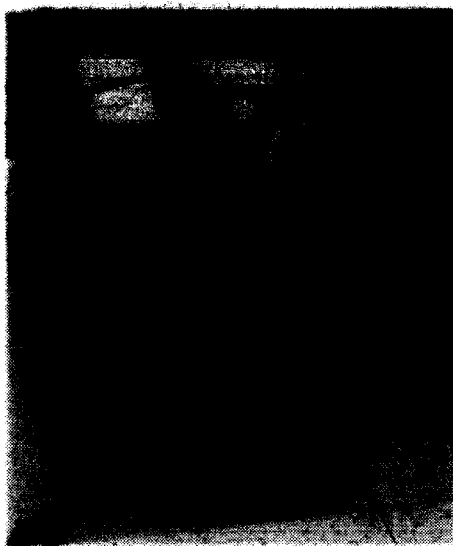


Photo 1. Experimental setup



Photo 2. X-Y table and CCD camera

구동부는 2개의 스테핑 모터로 구성되어 X-Y 테이블의 각축을 구동시키고, 끝단부에 CCD 카메라가 장착되어 영상 정보를 PC로 전송한다. 또한 원반형 경로 생성부는 DC 모터에 의해 구동되어 원형 물체의 운동을 생성한다.

실험 장치에 대한 주요 제원은 다음 Table 1.과 같다.

Table 1. Specifications of experimental setup

Item	Specification
CCD Camera	Panasonic WV-BP500
Image board	CI-512, 512×512, 256 gray level
X-Y table	Work space : H390×W200
Frame	H800×W800×V1000
Personal computer	IBM 486 DX2-66

CCD 카메라로부터 얻어진 이미지로부터 각각 다른 알고리즘을 적용하여 원형 물체상의 중심점을 계산한다. 이로부터 얻어진 정보로부터 현재의 카메라 이미지 평면상의 중심의 위치와 원형 물체의 중심점간의 위치 벡터를 계산하여 이미지 평면상의 중심이 원형 물체의 중심점을 추적 하도록 X-Y 테이블을 구동시킨다. X-Y 테이블의 구동은 2 개의 스테핑 모터에 의하여 이루어지므로 제어 방식은 개루프 방식이다. 다음 Fig. 6 은 추적 실험의 알고리즘을 보여주는 순서도이다.

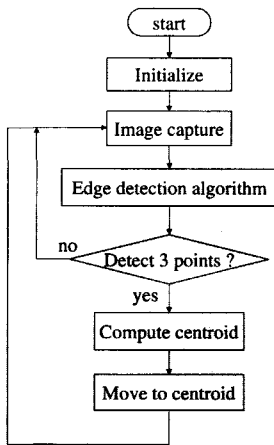


Fig. 6 Program algorithm

DC 모터로 구동되는 원반형 경로 생성부에 의하여 원형 물체는 $x^2 + y^2 = 60^2(mm)$ 의 원주상의 회전 운동을 하며 회전중심에 대한 회전 속도는 약 4 rpm 고 샘플링 주기는 1.0 sec 이다. 또한 원형 물체의 지름은 12mm 이며 카메라 프레임과 2D 평면간의 거리는 680 mm 이다.

X 방향 단순 검출법에서는 시뮬레이션과 같이 Δy 를 원 지름의 0.45 배로 설정하였으며 stick

method 와 T-shape method 에서는 원 지름의 0.7 배로 Δy 를 설정하였다.

위와 같은 실험을 통하여 얻은 결과를 Fig. 7 ~ Fig. 9 에서 보여주고 있다. 이미지 프레임 상에서의 목표 위치는 (265,395)로 설정되어 이동하는 원형 물체의 중심점이 항상 이 위치를 유지하도록 실험하였다.

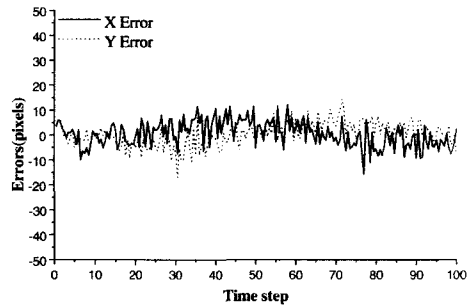


Fig. 7 Error by simple x directional method

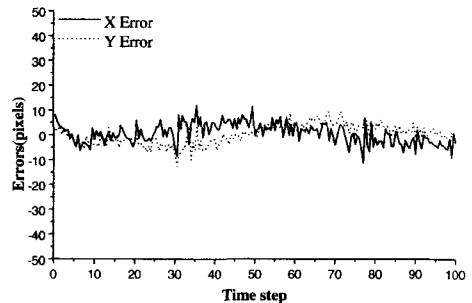


Fig. 8 Error by stick method

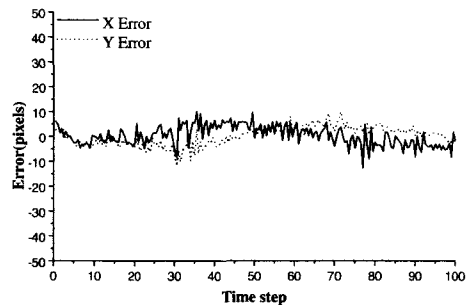


Fig. 9 Error by T-shape method

위 그림들의 결과로부터 분당 4 회전 정도의 속도 이내에서는 세 방법 모두에 의하여 ± 10 pixel 이내의 오차를 가지고 원형 물체의 추적이 잘 이루어짐을 관찰할 수 있다. 또한 T-shape method와 stick method에 의한 결과가 단순 x 방향 검출법에 비하여 더욱 추적 오차가 작음을 관찰할 있다.

다음 Table 2.는 각 방법에 의한 RMS 오차값을 보여주고 있다.

Table 2. RMS Errors

Method	X	Y
Simple x dir Method	74.2309	67.0046
Stick Method	58.6552	56.0446
T-shape Method	57.0095	53.1173

5. 결론

본 연구에서는 이동하는 원형 물체의 중심점을 찾기 위한 세 가지 알고리즘을 제시하고 각 방법들에 대하여 시뮬레이션과 간단한 비전 시스템을 이용한 실제 실험을 통하여 각 알고리즘의 효율성을 검출 정확도가 검출 속도의 측면에서 비교하였다.

X 방향 단순 검출법은 특성상 Δy 의 크기가 원형 물체 지름의 50%를 넘게 되면 검출이 실패할 가능성을 가지고 있으므로 Δy 의 설정에 제한을 많이 받고 따라서 검출 속도의 측면에서 효율이 떨어진다. 그러나 T-shape method와 stick method에서는 Δy 의 값을 원형 물체 지름의 70% 이상으로도 검출이 가능하므로 검출 속도가 빠르고, 알고리즘의 특성에 의하여 정확성도 증가할 수 있었다. 또한 검출 정확성이 유사한 경우 T-shape method가 stick method에 비해 다소 빠른 결과를 얻을 수 있었다. 그러나 stick method는 임의의 정형 물체에 적용할 수 있다는 점에 비해 T-shape method는 탐색 물체가 원형이라는 제약점을 가진다. 차후로 적용 기법을 이용하여 탐색 물체의 크기에 대해 Δy 를 최적화하고 실험 장치의 하드웨어의 성능을 향상시킨다면 보다 강건하고 빠른 성능을 가진 시스템의 구현이 가능할 것이다.

참고문헌

1. W. J. Wilson et al., "Relative End - Effector Control Using Cartesian Position Based Visual Servoing," IEEE Trans. Robotics and Automation, Vol. 12, No. 5, pp. 684-696, 1996.
2. R. Kelly., "Robust Asymptotically Stable Visual Servoing of Planar Robot," IEEE Trans. on Robotics and Automation, Vol. 12, No. 5, pp. 759-755, 1996.
3. D. B. Westmore. and W. J. Wilson., "Direct Dynamic Control of a Robot Using an End - Point Mounted Camera and Kalman Filter Position Estimation," Proc. IEEE Int. Conf. on Robotics and Automation, pp. 2376-2384. 1991.
4. T. S. Newman., and A. K. Jain., "A survey of Automated Visual Inspection," Computer Vision Image Understanding, Vol. 61, No. 2, pp. 231-262, 1995.
5. R. T. Chin., "Automated Visual Inspection Techniques and Applications : A Bibliography," Pattern Recognition, Vol. 15, No. 4, pp 343-357, 1982.
6. W. M. Wells., "Visual Estimation of a 3-D Line Segments from Motion-a Mobile Robot System," IEEE Trans. on Robotics and Automation, Vol. 5, No. 6, pp. 820-825, 1989.
7. L. Matthies., and S. A. Shafer., "Error Modeling in Stereo Navigation," IEEE Trans. on Robotics and Automation, Vol. RA-3, No. 3, pp. 239-248, 1987.
8. A. Kosaka., and A. C. Kak., "Fast Vision-Guided Mobile Robot Navigation Using Model-Based Reasoning and Prediction of Uncertainties," Journal of Computer. Vision, Graphics and Image Processing - Image Understanding, Vol. 56, No. 3, pp. 271-329, 1992.
9. B. K. P. Horn., Robot Vision, Cambridge, MA, MIT Press, 1986.
10. R. Jain et al., Machine Vision, McGraw-Hill, 1995.
11. R. C. Gonzalez., and R. E. Woods., Digital Image Processing, Addison-Wesley, 1987.
12. Y. S. Baek et al., "A New Algorithm for Edge Detection for an Arbitrary two Dimensional Shape," 1994 Int., Symposium on Industrial Robots, pp. 541-547, 1994.