

최적화된 신경회로망을 이용한 동적물체의 비주얼 서보잉

Visual Servoing of Robot Manipulators using the Neural Network with Optimal structure

°김 대 준, 전 효 병, 심 귀 보

중앙대학교 제어계측공학과 로봇틱스 및 지능제어 연구실
Tel:820-5319; Fax:817-0553; E-mail: kbsim@juno.cie.cau.ac.kr

Abstract This paper presents a visual servoing combined by Neural Network with optimal structure and predictive control for robotic manipulators to tracking or grasping of the moving object. Using the four feature image information from CCD camera attached to end-effector of RV-M2 robot manipulator having 5 dof, we want to predict the updated position of the object. The Kalman filter is used to estimate the motion parameters, namely the state vector of the moving object in successive image frames, and using the multi layer feedforward neural network that permits the connection of other layers, evolutionary programming(EP) that search the structure and weight of the neural network, and evolution strategies(ES) which training the weight of neuron, we optimized the net structure of control scheme. The validity and effectiveness of the proposed control scheme and predictive control of moving object will be verified by computer simulation.

Keywords Visual Servoing, Optimal Structure, Predictive Control, Evolutionary Programming, Evolution Strategies

1. 서 론

로봇 매니퓰레이터로 작업을 함에 있어서 주변 환경이나 혹은 대상물체에 대한 시각 정보는 매우 중요하다. 특히 물체가 움직이는 경우에 실시간으로 작업을 하기 위해서 predictive control 은 반드시 거쳐야만 하는 시각처리과정이다. 동적 물체에 대한 predictive control에 대한 연구는 로봇 매니퓰레이터와 결합되어 이동물체의 visual tracking이나 grasping의 분야에 많이 적용되어 왔고 지금도 활발히 연구되어지고 있다^{[1][2][3]}.

본 논문에서는 Kalman filter에 기초한 predictive filter를 사용하여 직육면체의 4개의 특징점에 대하여 각각 예측을 행함으로써 움직이는 물체에 대한 카메라의 파라미터를 변화시켜 물체의 자세를 정확히 측정하고자 하였으며, 제어계로서 최적화된 신경회로망을 이용하여 동적 물체에 대한 visual servoing을 실현하고자 한다.

일반적으로 신경회로망을 이용하여 제어계를 설계할 때 그 신경망의 구조와 학습능력에 대하여 고려할 필요가 있다. 신경회로망을 구성하고 학습을 시키거나 구조를 최적화시키는 방법으로서 진화연산에 대한 연구가 최근 활발히 진행되고 있다. 신경회로망의 구조를 표현하기 위해 구조행렬을 사용하거나, 이진 구조행렬과 가중치행렬로 표현한 유전 알고리즘을 이용하여 최적화 하는 방법^{[4][5][6]}등 여러 가지 최적화 기법에 대한 연구가 보고되어 있다. 그러나 유전 알고리즘은 각 노드와의 연결이 이진수로 표현되므로 실제적으로 weight가 연결되어 있지 않더라도 0의 값을 가지게 되어 염색체의 길이가 길어지고 진화시간이 길

어질 우려가 있다. 따라서 다 자유도를 갖는 로봇 제어계의 설계에는 적합하지 않다.

따라서 본 논문에서는 그래프 구조를 이용하는 코드화 방법^[7] 즉 진화프로그래밍을 이용하여 전방향 신경회로망의 최적구조(최적의 node 수, weight 수)를 구하는 방법을 제안하였고, 신경세포들 간의 결합은 feedforward만을 고려하였다. 또한 진화전략(ES)을 이용하여 각 뉴런사이의 연결강도를 학습시키는 구조로서 제어계를 구성하였다.

본 연구에서는 제안한 제어계를 RV-M2 로봇 매니퓰레이터의 비주얼 서보잉에 적용하였다. 종래의 이미지 자코비언(Image Jacobian)을 이용하는 비주얼 서보잉에서는 카메라의 이미지로부터 로봇 매니퓰레이터의 제어입력을 얻기 위하여 대량의 연산을 필요로 하는 좌표변환이 필요하다^{[8][9][10]}. 따라서 본 논문에서는 이러한 문제점을 개선하고 실시간의 제어를 목표로 이미지상의 물체의 특징량을 이용해서 매니퓰레이터를 직접적으로 제어하여 제어계의 동적인 특성을 향상시키고자 하였으며 Kalman filter에 기초한 predictive filter를 사용함으로써 실시간으로 동적 물체를 제어 하고자 하였다.

마지막으로 제안한 두 가지의 제어계 즉 진화전략과 신경회로망만을 사용한 제어계와 전 방향 신경회로망을 진화프로그래밍으로 구조를 최적화한 제어계를 컴퓨터 시뮬레이션을 통하여 비교·검토하였다.

2. 움직이는 물체의 예측

본 논문에서는 움직이는 물체의 예측을 위하여 Kalman filter를 사용하였다. Kalman filter는 외부의 잡음에 대하여 강하고 움직임을 예측을 부드럽게 할 수 있으므로 predictive control에 자주 사용이 된다^{[1][2][3]}

움직이는 물체를 로봇 매니퓰레이터로 제어하는 알고리즘은 크게 다음의 두 단계로 나눌 수 있다^[2].

① filtering and prediction

② kinematic transformation computation

①번 단계에서는 filter의 입력으로 CCD카메라로부터 얻은 물체의 실제 위치가 들어가고 출력 값으로는 물체의 예측 위치 값이 나온다. ②번 단계에서는 물체의 목표 이미지와 예측 값과의 오차가 신경망의 입력으로 들어가서 출력으로 RV-M2로봇 매니퓰레이터의 5개 관절각의 변화량이 나오고 가중치를 진화전략을 이용하여 학습시킴으로써 오차를 감소시키는 방향으로 로봇 매니퓰레이터가 이동하게 된다.

2.1 운동 모델

이동물체가 등가속도 운동을 한다고 가정한다.

$$A = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} T^2/2 \\ T \\ 1 \end{bmatrix} \quad (1)$$

$$X(k+1) = A \cdot X(k) + B \cdot w(k) \quad (2)$$

$X = [x \ \dot{x} \ \ddot{x}]^T$, $w(k)$ 는 잡음을 나타내는 변수이고, T는 샘플링 시간을 나타낸다.

2.2 측정 모델

$$m(k) = [1 \ 0 \ 0] \cdot X(k) \quad (3)$$

$m(k)$: measurement vector

$$\hat{X}(k) = A \cdot \hat{X}(k-1) + G \cdot e(k) \quad (4)$$

$$e(k) = x(k) - [1 \ T \ T^2/2] \cdot \hat{X}(k-1) \quad (5)$$

$$px = [1 \ T \ T^2/2] \cdot \hat{X}(k) \quad (6)$$

$$G = \left[\alpha \frac{\beta}{T} \quad \frac{\gamma}{T^2/2} \right]^T \quad (7)$$

$\hat{X}(k)$: updated stated vector

G: Kalman gain

px: predicted value of position

α, β, γ : end-effector의 roll, pitch, yaw

위의 식에서 볼 수 있는 바와 같이 움직이는 물체의 자세를 잘 측정하기 위하여 Kalman gain에 α, β, γ 를 사용하여 물체의 updated stated vector를 구함으로써 이동물체의 예측위치를 구하였다^[1].

3. 진화전략과 신경망을 이용한 비주얼 서보잉

3.1 제어계 설계

본 논문에서는 4개의 특징점에 대한 목표 이미지와 2장에서 구한 움직이는 물체의 예측되어진 이미지와의 x, y 8개 좌표편차 값을 입력으로 가지고, $\theta_1 \sim \theta_5$ 5개의 매니퓰레이터 관절각을 출력으로 가지며 8개의 중간층을 가진 신경망을 구성하였다. 입력층-중간층-출력층은 완전히 연결되어 있고 104개의 weight를 가지고 있다. 따라서 104개의 벡터열과 표준편차가 진화전략의 하나의 염색체로 구성된다.

Kalman filter를 통하여 예측되어진 4개의 특징점들 목표 이미

지에 일치시키기 위해 각각의 특징점을 비교하여 그 오차를 0으로 감소시키는 벡터열을 찾는다. 오차가 0으로 수렴하는 염색체의 진화를 위해서 본 논문에서는 다음과 같은 적합도 함수를 선정하였다. 진화전략에 아래와 같이 적합도 함수를 선정한 것은 카메라의 이미지 평면에서 목표이미지와 2차원 정보를 비교하기 위해 각 특징점의 x, y 좌표편차와 선분의 길이편차를 고려하고자 한 것이다.

$$\text{fitness} = a \cdot \Delta R + b \cdot \Delta L \quad (8)$$

여기서 a는 직육면체 특징점의 거리 변화량에 대한 비례상수, b는 직육면체 한 변의 길이 변화에 대한 비례상수이다. 1회의 측정은 50 step (1초)을 넘지 않도록 하였고 진화전략은 (10+100)-ES를 사용하였다^{[1][12]}.

진화과정을 단계별로 나타내면 다음과 같다.

[step 1] 초기위치 결정

물체(Object)의 초기위치 및 로봇 매니퓰레이터의 초기 자세를 설정한다.

[step 2] prediction of the moving object

CCD카메라로부터 얻은 물체의 실제 위치를 입력으로 하여 움직이는 물체의 예측위치를 이끌어낸다.

[step 3] 전방향 신경회로망

예측되어진 이미지와 목표 이미지와의 8개의 x, y 좌표편차를 신경망의 입력으로 받아들여 RV-M2 매니퓰레이터의 5개 관절각의 변화량을 출력으로 이끌어낸다.

[step 4] 엔드 이펙터의 이동

step 3에서 구한 관절각의 변화량을 이용하여 Direct kinematics를 이용하여 이동 후 매니퓰레이터의 position, orientation을 구한다.

[step 5] 물체의 2차원 x, y정보

step 4에서 구한 3차원 정보를 카메라의 이미지 평면에 투영해서 물체의 2차원 정보를 구한다.

[step 6] 진화전략을 이용한 비교

(8)의 fitness식을 이용하여 평가함으로서 카메라의 이미지가 목표이미지에 일치될 때까지 step 2로 되돌아가서 같은 과정을 반복하고, 오차가 0으로 수렴하면 진화의 과정을 종료한다.

3.2 시뮬레이션에 의한 검토

시뮬레이션에서는 다음과 같은 가정에 실시하였다.

[1] 물체의 초기위치와 각각의 특징점 좌표에 대한 정보는 이미 알고 있고,

[2] 대상물체는 매니퓰레이터의 작업공간 내에서 20 ± 5 mm/sec로 X-Y평면에서 랜덤하게 운동하고 있으며, 모든 특징점이 카메라에 포착되어 있다.

대상물체는 가로와 세로 각각 50mm, 높이 30mm인 직육면체로 하였고, 로봇 매니퓰레이터는 RV-M2를 대상으로 하였으며, 5개 관절각의 동작 범위는 아래와 같다.

표1. RV-M2 로봇 매니퓰레이터의 각 관절의 동작범위
Range of RV-M2 robot manipulator's joint angle

θ_1	θ_2	θ_3	θ_4	θ_5
-150°~150°	-30°~100°	-120°~0°	-200°~20°	-180°~180°

시뮬레이션을 위한 매니퓰레이터의 초기 관절각은 $\theta_1 = 0^\circ$, $\theta_2 = 60^\circ$, $\theta_3 = -60^\circ$, $\theta_4 = -90^\circ$, $\theta_5 = 0^\circ$ 로 정면을 바라보도록 설정하였으며, 진화전략의 표준편차는 0.15, 돌연변이 확률은 0.25로하여 지역최소값에 빠지지 않게 했다. 그리고 이미지 샘플링 시간은 0.02초로 설정하였다.

- dx : 동적물체의 실제이미지와 목표이미지 사이의 특징점 x좌표 변화량의 평균.
- dy : 동적물체의 실제이미지와 목표이미지 사이의 특징점 y좌표 변화량의 평균.

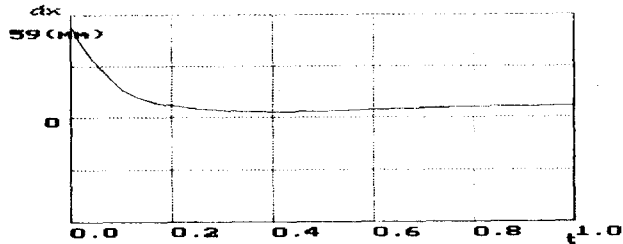


그림 1. ES를 이용한 x좌표편차 변화량의 평균(50세대 진화)
Fig. 1. Average on the deviation of x using ES

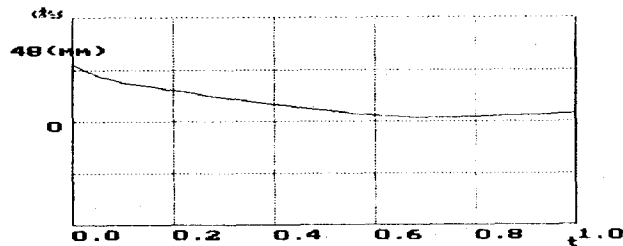


그림 2. ES를 이용한 y좌표편차 변화량의 평균(50세대 진화)
Fig. 2. Average on the deviation of y using ES

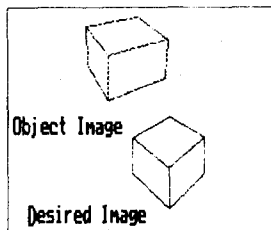


그림 3. 초기 이미지
Fig. 3. Initial image

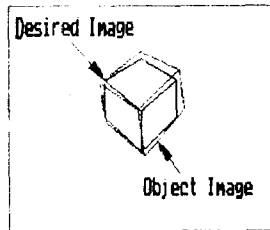


그림 4. 최종 이미지
Fig. 4. Final image

4. 진화프로그래밍을 이용한 전방향 구조 최적화

4.1 코드화 방법

신경회로망을 일차원적인 배열로 생각한 코드화 방법을 사용하였다. 우선 입력층 노드부터 출력층 노드까지 순서를 정하였는데 입력층은 0~7을 사용하였고 중간층은 최소 1개, 최대 12개의 node를 사용하였으며 출력층은 마지막 5개의 node를 사용함으로써 전체적으로 최소 14개, 최대 25개의 node를 가지도록 하였다. 또한 입력층에서는 신경세포간의 결합이 시작되기만 하고 출력층에서는 끝나기만 하며 중간층에서는 결합에 제한을 두지 않고 다만 모든 결합이 feedforward 하다는 가정을 하였다^[7].

4.2 적합도 함수

전방향 신경망의 구조를 최적화하기 위하여 적합도 함수로서 fitness1(fit), node의 수(nn), weight의 수(nw)를 사용하여 아래와 같이 나타내었다.

$$\text{cost} = \alpha/\text{fit} + \beta \cdot \text{nn} + \gamma \cdot \text{nw} \quad (9)$$

$$\text{fitness2} = \frac{1}{\text{cost}} \quad (10)$$

여기서 α 는 $\frac{1}{\text{fit}}$ 에 대한 비례상수, β 는 node의 수에 대한 비례상수, γ 는 weight의 수에 대한 비례상수를 나타내며 적합도 함수를 계산할 때 진화전략에서 구한 fitness1에 대한 영향을 가장 많이 받고, 다음으로 node의 수에 대한 영향을 많이 받도록 $\alpha = 1$, $\beta = 0.01$, $\gamma = 0.005$ 로 가중치를 조절함으로써 진화전략에서 구한 좋은 개체에 대하여 간단한 구조로서 목표 이미지와의 편차를 감소시키고자 하였다.

4.3 돌연변이

본 논문에서는 주로 돌연변이가 연산자를 이용하여 연산을 행하였으며, 신경세포를 더하는 경우(P_{an}), 신경세포를 빼는 경우(P_{dn}), 신경세포간의 weight를 더하는 경우(P_{aw}), 신경세포간의 weight를 빼는 경우(P_{dw})를 4가지의 확률로 표현하였다.

$$P_{an} = \frac{2}{3}x - \frac{1}{3}x^2 \quad (11)$$

$$P_{dn} = \frac{2}{3}(x-1)^2 \quad (12)$$

$$P_{aw} = \frac{2}{3}x^2 \quad (13)$$

$$P_{dw} = \frac{1}{3} - \frac{1}{3}x^2 \quad (14)$$

$$\text{with } x = 1 - \frac{1/\max \text{ fitness}}{1/\text{fitness}} = 1 - \frac{\min \text{ cost}}{\text{cost}}$$

여기서 min_cost값은 cost 값이 가질 수 있는 최소의 수로 미리 지정해 주었고, 각각의 확률과 x와의 관계를 그림 5, 6에 나타내었다.

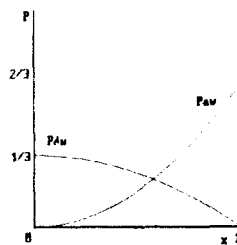


그림 5. weight를 더하고 빼는 확률

Fig. 5. Probability of add or delete weight.

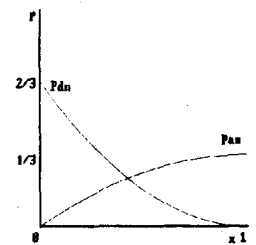


그림 6. node를 더하고 빼는 확률

Fig. 6. Probability of add or delete node.

진화전략에서 구한 fitness가 크면 cost 값이 작아지고 x의 값도 작아져서 돌연변이가 연산자의 확률이 다음과 같이 된다.

$$P_{dn} > P_{dw} > P_{an} > P_{aw}$$

반대로 진화전략에서 구한 fitness가 작으면 돌연변이가 연산자의 확률은 아래의 순서와 같이 된다.

$$P_{aw} > P_{an} > P_{dw} > P_{dn}$$

즉 진화전략에서 구한 fitness가 크면 node를 감소시킬 확률을 크게 하고 fitness가 작으면 weight를 더하는 확률을 크게하여 진화의 성능을 개선하였다.

4.4 시뮬레이션에 의한 검토

돌연변이에서 사용한 확률에 의해서 node와 weight의 추가 및 삭제를 행하였으며 진화 프로그래밍을 통하여 얻은 구조의 가중치를 진화전략을 통하여 학습시킴으로써 동적물체의 실제 이미지와 목표이미지와의 오차를 감소시켰다.

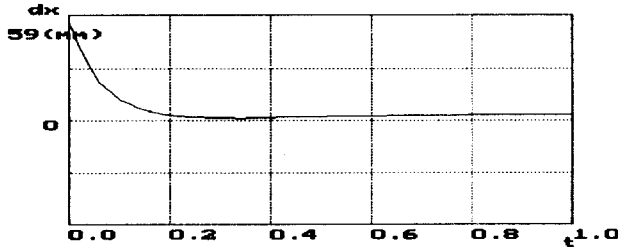


그림 7. EP를 사용한 x좌표편차 변화량의 평균 (20세대 진화)
Fig. 7. Average on the deviation of x using EP

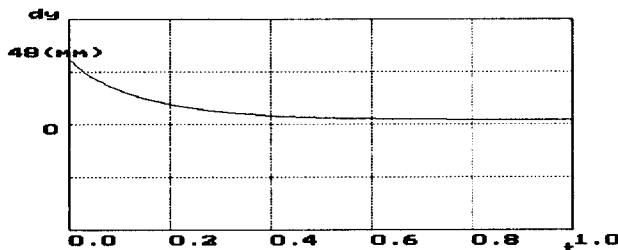


그림 8. EP를 사용한 y좌표편차 변화량의 평균 (20세대 진화)
Fig. 8. Average on the deviation of y using EP

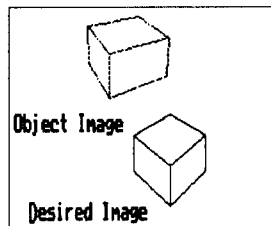


그림 9. 초기 이미지
Fig. 9. Initial image

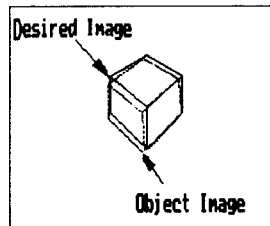


그림 10. 최종 이미지
Fig. 10. Final image

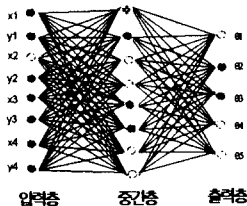


그림 11. 초기 구조

Fig. 11. Initial structure

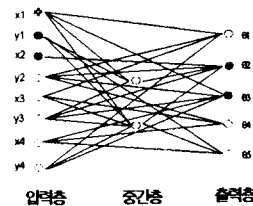


그림 12. EP 진화 후 구조

Fig. 12. After evolution structure

위의 시뮬레이션에서 볼 수 있듯이 EP진화 후 얻은 2개의 중간층 node와 27개의 weight를 가진 신경망을 구성하여 좁아진 탐색체로 진화시킴으로써 20세대 후 진화전략을 사용하여 50세대 진화시킨 결과보다도 우수한 성능을 얻을 수 있었다.

5. 결론

본 논문에서는 Kalman filter에 기초한 predictive filter를 사용하여 움직이는 물체의 예측위치를 갱신함으로써 실시간으로 물체를 제어하고, 제어계로서는 신경망의 weight를 진화전략(ES)를 이용하여 학습시킴으로써 예측되어진 이미지와 목표 이미지와의 오차를 줄이고 그러한 신경망의 전 방향 구조를 그래프 구조를 이용하는 진화 프로그래밍(EP)으로서 최적화하는 비주얼 서보잉을 제안하였다. 시뮬레이션 결과에서 볼 수 있는 바와 같이 대량의 연산을 필요로 하는 좌표변환 대신 최적의 신경망을 이용하여 움직이는 물체에 대하여 좋은 결과를 얻을 수 있었다. 또한 진화 프로그래밍을 이용하여 구조를 간단하게 함으로써 적은 개수의 염색체(weight 수)와 node 수로서 더욱 우수한 성능을 이끌어 낼 수 있었다.

제어계를 구성하여 학습을 시키는 도중에 제어계의 성능이 각 파라미터의 가중치에 의해 크게 좌우됨을 알았으며 가중치를 선정하는 방법에 대한 연구가 필요할 것이다.

참고문헌

- [1] Takashi Oya, Hideki Hashimoto, Fumio Harashima, "Predictive Filtering for Visual Tracking," IECON, pp. 1718-1723, 1993.
- [2] Peter K.Allen, et al."Trajectory Filtering and Prediction for Automated Tracking and Grasping of a Moving Object," Proc. of the IEEE international Conference on R&A, pp 1850-1856, 1992.
- [3] Koichi Hashimoto, Hidenori Kimura, "Visual Servoing with Nonlinear Observer," Proc. of the IEEE international Conference on R&A, pp. 484-489, 1995.
- [4] 조철현, 공성근, "유전 알고리즘을 이용한 전방향 신경망의 구조 최적화," 인공지능,신경망 및 퍼지 시스템 춘계종합학술대회 논문집, pp. 131-136, 1996.
- [5] S.G. Roberts and M. Turega, "Evolving neural network structures : an evaluation of encoding techniques," Proceeding of the International Conference in Ales, France, pp. 96-99, 1995.
- [6] Minoru Fukumi et al. "A Method to Designa Neural Network by the Genetic Algorithm with Partial Fitness," 시스템制御情報學會論文集, Vol. 9, No. 2, pp. 74-81, 1996.
- [7] 이지호, 김종환, "순서화된 진화 신경 회로망 및 이의 로봇 매니플레이터 제어에의 응용," 제어계측,자동화, 로봇틱스 연구회 합동학술발표회 논문집, pp. 243-246, 1996.
- [8] 심귀보, "동적 물체에 대한 로봇 매니플레이터의 Visual Servoing," 대한 전자공학회 논문지, Vol. 4, No. 2, pp. 15-24, 1996.
- [9] 심귀보, "Visual Control of Robot Manipulator Based on Neural Network," 한국자동제어 학술회의 논문집, pp. 1042-1046, 1993.
- [10] P. J. Mckerrow, "Introduction to Robotics," ADDISON WESLEY, 1991.
- [11] Beack et al. "An Survey of Evolution Strategies," Proc. of 4th International Conference on Genetic Algorithms(ICGA), 1991.
- [12] Zbigniew Michalewicz, "Genetic Algorithms + Data Structures = Evolution Programs," Springer-Verlag, pp. 160-168, 1995.