

코호넨 네트워크 및 시간 지연 신경망을 이용한 움직이는 물체의 중심점 탐지 및 동작특성 분석에 관한 연구

A Study on Center Detection and Motion Analysis of a Moving Object by Using Kohonen Networks and Time Delay Neural Networks

황 정 구* 김 종 영** 장 태 정***
Hwang, Jung-Ku Kim, Jong-Young Jang, Tae-Jeong

Abstract

In this paper, center detection and motion analysis of a moving object are studied. Kohonen's self-organizing neural network models are used for the moving objects tracking and time delay neural networks are used for dynamic characteristic analysis. Instead of objects brightness, neuron projections by Kohonen Networks are used. The motion of target objects can be analyzed by using the differential neuron image between the two projections. The differential neuron image which is made by two consecutive neuron projections is used for center detection and moving objects tracking. The two differential neuron images which are made by three consecutive neuron projections are used for the moving trajectory estimation. It is possible to distinguish 8 directions of a moving trajectory with two frames and 16 directions with three frames.

키워드 : 코호넨 네트워크, 시간지연신경망, 중심점 탐지, 경로추적
Keywords : Kohonen Networks, TDNN, center detection, tracking locus

1. 서론

컴퓨터 시각(computer vision)에 관한 연구는 어떤 장면의 영상을 분석하여 장면에 대한 정보를 추출해내고 추출된 객체를 인식하는 것에 주된 관심이 있다. 초기에는 단일 화상만을 이용하여 객체를 탐지하고 객체에 대한 여러 가지 정보를 추출해 내고자 하였던 것이 움직임 분석 기법 및 컴퓨터 처리속도가 급속히 발전한 지금은 연속적인 화상에서 움직이는 물체에 대한 여러 가지 정보를 실시간으로 추출하고자 하는 움직임 처리에까지

확대되고 있다. 이러한 움직임의 처리는 군사, 교통, 기상, 보안 등 실세계의 다양한 응용 분야에서 유용하게 사용될 수 있다. [1]

연속적으로 입력되는 화상으로부터 실시간으로 움직이는 물체를 감지하고 이러한 물체를 추적하는 물체 추적 기법은 화상 처리 기술 중에서도 가장 중요한 기술 중의 하나로 인식되고 있으며 특히 물체의 동작 분석은 시각적 정보에서 잉여 정보를 줄이는 방법으로 많은 연구의 대상이 되어 왔다. 기존의 대부분의 연구 방법에서 화상으로부터 얻을 수 있는 정보로는 Gray, RGB, YUV 등이 있으나, 계산량의 고리를 위해 Y값, 즉 명암값을 가지고 블록정합에 의한 방법, 광류에 의한 방법, 차영상에 의한 방법 등으로 중심점의 탐지 및 추적, 동작 분석을 하여 왔다. 블록정합에 의한 방법은 참조 영상과 블록화된 영상의 유사성을 가지고

* 강원대학교 제어계측공학과, 공학석사
** 강원대학교 제어계측공학과, 석사과정
*** 강원대학교 제어계측공학과 부교수, 공학박사

움직임을 추정하는 방법이다. 하지만 전체 영상을 블록화하여야 하기 때문에 방대한 계산량이 다르게 되고 블록 크기의 결정에도 어려움이 있다. 광류에 의한 방법은 연속된 영상에서 광강도의 변화로 움직임을 추정한다. 이 또한 세밀한 광강도를 요구하는 단점이 있다. [2]-[5]

본 논문에서는 이러한 명암값에 의존하던 방법이 아닌 신경망 모델을 도입하여 중심점 탐지 및 동작 분석에 관한 방법을 제안한다. 초기 격자화된 뉴런들을 가지고 신경망 모델을 도입하여 물체의 사영을 만들었다. 물체의 중심점 탐지 및 추적은 다층 신경망이고 경쟁학습 네트워크의 일종인 코호넨 네트워크(Kohonen Networks)를 사용하였으며 표적의 동작 분석은 기존의 차영상 분석에 의한 방법과 시간 지연 신경망(Time Delay Neural Networks)을 사용하여 경로 추적을 하였다.

2. 기본 과정 및 동작

2.1 기본 과정

먼저 시스템의 동작에 앞서 다음과 같은 기본적인 동작의 가정을 필요로 한다.

- ① 고정된 카메라와 움직이는 물체
- ② 표적 물체의 명암값은 배경과 다르다
- ③ 카메라를 향해 오거나 멀어지는 운동이 아니다
- ④ 표적 물체는 관련된 연속 영상에 대하여 항상 등속도 운동을 한다
- ⑤ 동작은 표적 물체의 사영(projection) 내에서만 일어난다

본 논문에서는 차영상에 의한 방법을 기본으로 하기 때문에 차영상이 발생하지 않는 조건, 즉 물체의 동작이 없는 경우와 동작이 목표 물체의 사영외에서 일어나는 경우에는 원하는 결과를 얻지 못한다.

2.2 시스템의 동작

전체 시스템에서 첫 동작은 연속적으로 입력되는 영상에서 불필요한 잡음의 제거와 표적 물체의 잡영에 의한 정보의 손실을 보상해 줄 수 있는 전처리 과정을 거치는 것이다. 그 다음으로 영상에 대하여 격자화된 뉴런(neuron)들을 삽입하게 된다. 초기 뉴런들은 일정한 간격으로 격자화되어 있으며 코호넨 네트워크를 통과후 입력 물체의 사영적 요소로 변화하게 된다. 각각의 영상에 대하여 코호넨 네트워크를 적용하여 얻은 뉴런들을 가지고 물체의 명암값에 의한 차영상(이하 뉴런 차영상)을 얻을 수 있다. 이렇게 얻어진 뉴런 차영상을 기반으로 물체의 중심점 탐지 및 추적, 경로 추정이 가능하게 된다.

물체의 중심점은 연속적으로 입력되는 2개의 영상에서 얻어지는 뉴런 차영상 1개로 구할 수 있으

며, 표적의 이동 경로는 연속적인 3개의 영상에서 얻을 수 있는 2개의 뉴런 차영상들을 시간 지연 신경망의 입력으로 하여 추정이 가능하게 된다.

3. 영상의 전처리

영상의 전처리 과정은 모든 영상 처리 시스템의 성능을 좌우하는데 큰 역할을 차지하지만 그 결과가 좋을수록 방대한 계산량과 복잡해지는 수식때문에 본 논문에서는 가장 간단하면서 만족할만한 결과값 얻을 수 있는 가중치(weight) 필터링과 평활화(smoothing) 방법 그리고 크기 필터링에 의한 방법을 사용하였다.

가중치 필터링은 메디안(median) 필터링보다 상대적으로 이미지의 왜곡이 적다는 장점이 있으며 크기에 의한 필터링 방법은 어느 정도 영상의 손실을 감안하는 영상, 즉 표적의 크기가 어느 정도 큰 영상에 적합하다고 할 수 있다.

4. 물체의 중심점 탐지 및 추적

4.1 코호넨 네트워크

코호넨 네트워크는 다층 신경망이며 경쟁학습 네트워크의 일종이다. 코호넨 네트워크의 클러스터링 능력을 사용하여 표적 물체에 대한 사영을 만들었다. 여기서 클러스터링(clustering) 능력이란 학습이 진행됨에 따라 각 뉴런이 특정 패턴의 중심으로 이동함으로써 각 패턴들을 구분 지을 수 있게 하는 것이다. [6]

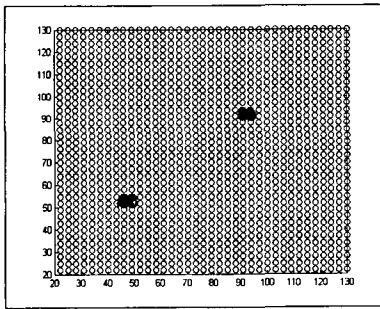
또한 경쟁학습 네트워크의 동작이 입력에 대하여 가장 가까운 가중치를 갖는, 즉 유클리드(euclid) 거리가 가장 작은 뉴런 한 개만 활성화되는 반면에 코호넨 네트워크는 가장 가까운 가중치를 갖는 출력 뉴런, 즉 승자 뉴런과 인접 뉴런들이 동시에 활성화된다. 이때 인접 뉴런들은 승자 뉴런의 이웃 뉴런으로 사용된다.

본 논문에서는 코호넨 네트워크의 입력으로 표적 물체의 각각의 픽셀에 대한 X, Y 좌표값을 사용하였으며 인접 뉴런은 승자 뉴런을 중심으로 24개의 이웃하는 뉴런들을 사용하였다. 활성화된 뉴런은 학습율(learning rate)에 의해 입력값에 근사하게 학습되어 진다. 학습은 활성화된 뉴런이 입력값과 일치하게 되면 끝나게 되지만 표적 물체의 중심점에 가장 근접하는 결과를 얻기 위해서는 많은 반복 계산이 필요하다.

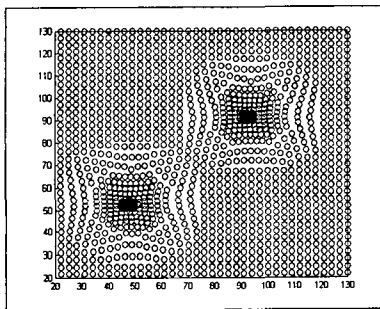
4.2 표적 물체에 대한 사영 추출

표적 물체의 사영은 연속된 3개 프레임의 영상을 전처리 한 후 각각의 프레임에 대하여 격자화된 초기 뉴런들의 삽입으로부터 시작되며 초기 뉴

런들은 모든 프레임의 영역에 일정한 간격으로 채워지게 된다. 본 논문의 Matlab을 이용한 시뮬레이션에서는 뉴런들의 간격이 위, 아래로 모두 3 픽셀의 간격을 가지고 초기화하였다. 이렇게 격자화된 뉴런들의 X, Y 좌표값은 각 뉴런들의 초기 가중치가 된다. 영상의 전체를 스캐닝(scanning)하여 표적 물체를 추출하고, 이렇게 찾아낸 표적 물체의 각 픽셀의 X, Y 좌표값이 코호넨 네트워크의 입력이 된다. 코호넨 네트워크의 동작에 의하여 격자화된 초기 뉴런 중에서 입력에 가장 가까운 뉴런(승자 뉴런)과 승자 뉴런을 중심으로 24개의 뉴런이 활성화되어 가중치가 갱신된다. 여기서 코호넨의 자기조직화(self-organizing) 신경망을 사용하여 표적 물체에 의한 뉴런들의 사영을 추출하게 된다. 이러한 뉴런들에 의한 사영을 추출하는 이유는 표적 물체의 각 픽셀 모두에 대하여 계산하는 비효율적인 면을 보완하기 위해서이다.



(a) 표적 물체 및 격자화된 초기 뉴런



(b) 코호넨 네트워크에 의한 표적 물체의 사영

그림 1. 초기 뉴런과 표적 물체의 사영

그림 1은 초기 격자화된 뉴런에서 코호넨 네트워크 통과 후의 뉴런들의 변화를 나타낸다. 뉴런들은 'o'로 표시하였고 표적 물체는 '*'으로 표시하였다. 격자화된 초기 뉴런에서 코호넨 네트워크 통과 후의 뉴런들은 표적 물체에 대하여 근접하게 변화

기 때문에 표적 물체를 대신하는 사영으로 나타난다.

4.3 뉴런 차영상 및 표적 물체의 중심점 탐지

위와 같은 방법으로 연속적으로 입력되는 3개의 프레임에 대하여 3개의 뉴런 사영을 얻을 수 있으며 프레임과 프레임 사이에서는 차영상이 발생함으로써 2개의 뉴런 차영상을 구할 수 있다. 그리고 각 뉴런은 가중치로 대표되기 때문에 뉴런들의 변화, 즉 가중치의 변화는 X, Y 좌표값으로 나타낸다. 여기서 뉴런 차영상은 X, Y의 각 차이 중에서 큰 값을 취하게 된다. 뉴런 차영상의 각각의 값은 표적 물체의 동작을 나타낸다. 연속적으로 입력되는 프레임에 두 개의 물체가 있고 물체에 대한 사영이 겹치지 않는다는 가정이면 동작이 없는 물체에 대하여 뉴런 차영상은 0인 값을 가지게 된다. 따라서 움직임이 없는 물체에 대한 뉴런들의 변화는 뉴런 차영상을 구함으로써 없어진다. 표적 물체의 중심점은 위에서 구한 뉴런 차영상을 기반으로 구할 수 있다. 초기 격자화된 뉴런에서 뉴런의 변화는 표적 물체를 중심으로 일정하게 변화하게 되며, 변화하는 뉴런의 범위도 항상 일정하다. 따라서 동작이 있는 부분은 각 프레임의 표적 물체에 대한 사영들의 변화를 모두 가지고 있다고 할 수 있다.

표적 물체의 중심점은 0이 아닌 값을 가지는 뉴런 차영상의 각 값에 대응하는 두 사영의 뉴런에서 구할 수 있다. 각 프레임에 대한 사영은 초기 뉴런이 표적 물체에 따라 가중치가 갱신되므로 한번이라도 활성화된 뉴런들을 알 수 있다. 만약 연속된 프레임에서 물체가 단지 하나인 경우에는 표적 물체에 대하여 한번이라도 활성화된 뉴런들의 가중치값, 즉 X, Y 좌표값을 모두 더하여 평균을 취함으로써 구할 수 있다. 하지만 물체가 두 개인 경우는 위와 같은 방법으로는 중심점을 찾기가 어렵기 때문에 뉴런 차영상에서 0이 아닌 값에 대응하며, 각 프레임에 대한 사영의 뉴런 중에서 한번이라도 활성화된 뉴런들의 X, Y 좌표값의 평균을 취함으로써 구할 수 있다.

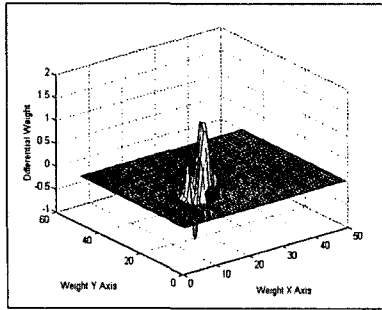
5. 물체의 동작 특성 분석

5.1 누적 차영상의 획득

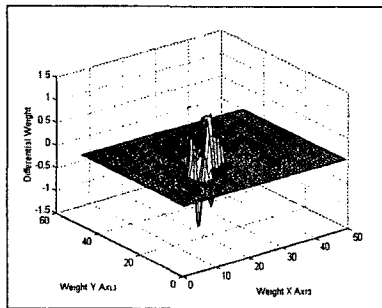
연속적으로 입력되는 3개의 프레임에 대하여 코호넨 네트워크를 사용하여 각 프레임의 표적 물체에 대한 사영을 구하고 이를 통해 얻어진 3개의 사영을 가지고 2개의 뉴런 차영상을 추출하는 것이다.

그림 2는 표적 물체가 연속된 3개의 프레임의 영상에 걸쳐 오른쪽 오른쪽으로 이동하였을 때 추출된 2개의 누적 뉴런 차영상을 나타낸다. 이상

적인 경우 첫 번째 뉴런 차영상과 두 번째 뉴런 차영상이 일치하게 나타나야 하지만 초기 뉴런들이 일정한 간격을 두고 격자화되어 있기 때문에 정확히 일치하지는 않는다.



(a) 첫 번째 사영과 두 번째 사영에 의한 뉴런 차영상



(b) 두 번째 사영과 세 번째 사영에 의한 뉴런 차영상

그림 2. 표적 물체의 이동에 따른 누적 뉴런 차영상

5.2 시간 지연 신경망

시간 지연 신경망은 기존의 시간의 개념이 없는 정적인 데이터만 이용할 수 있었던 신경망 연결에 시간 지연을 넣어 시간에 따라 변화하는 값도 고려할 수 있게 설계한 신경망이다. 시간 지연 신경망의 기본 구조는 지연요소 D_1 에서 D_N 을 하나의 입력에 시간상으로 지연시키고 지연된 입력과 가중치가 곱해진 합을 활성화 함수 F 를 통해 다음 계층에 인가하는 구조로 되어 있다. 따라서 처리 요소들은 시간축 상에서 N 개의 서로 다른 위치에 배열되어 있는 입력들을 검출하게 된다. 이렇게 함으로써 일정한 간격을 두고 연속적으로 입력되는 화상에서 영상의 시간적 특징을 추출할 수 있게 된다. 시간 지연 신경망의 입력으로는 앞에서 구한 누적 차영상의 각각의 차영상값이 사용된다. 현재 프레임과 한 프레임 전의 사영에 의한 뉴런 차영

상(이하 현재 차영상)의 각각의 차이값들이 시간 지연 신경망의 입력이 되며, 한 프레임 전의 사영과 두 프레임 전의 사영에 의한 뉴런 차영상(이하 과거 차영상)이 지연값으로 현재 차영상의 각 값들과 연계되어 시간 지연 신경망의 은닉층(hidden layer)으로 연결된다.

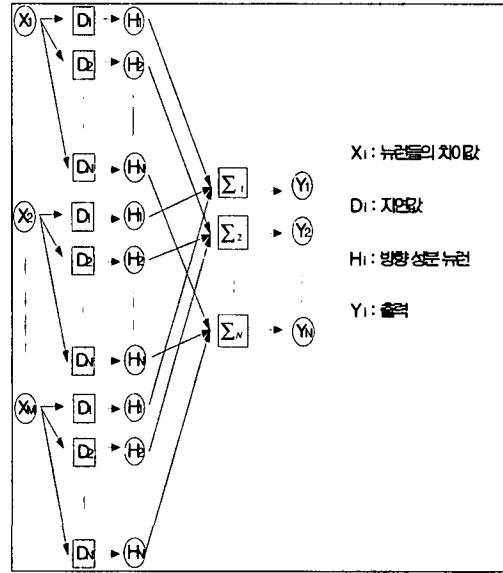


그림 3. 논문에서 사용된 TDNN의 구조

본 논문에서 시간 지연 신경망의 입력은 현재 뉴런 차영상의 0이 아닌 값을 가지는 뉴런(이하 입력 뉴런)이 사용되었다. 현재 뉴런 차영상의 모든 뉴런 중에서 동작이 발생한 뉴런만 시간 지연 신경망의 입력으로 사용되기 때문에 은닉층의 개수는 일정하게 정해져 있지 않다. 은닉층은 각 입력에 대하여 24개로 구성되어 있다. 은닉층의 개수는 입력 뉴런의 개수와 24의 곱이 된다. 각 입력에 대하여 24개의 은닉 뉴런을 만든 것은 표적 물체의 경로 추정의 방향을 모두 24가지 방향으로 나타내기 위함이다. 따라서 시간 지연 신경망의 하나의 입력에 대하여 24개의 방향적 요소를 포함하는 은닉 뉴런(이하 방향 성분 뉴런)이 연결되어 있는 것이다. 은닉층으로의 입력은 입력 뉴런과 입력 뉴런을 중심으로 인접한 24개의 뉴런의 위치와 대응되는 과거 뉴런 차영상의 뉴런들의 값이 사용된다. 여기에서 인접한 24개의 뉴런의 위치와 대응되는 과거 뉴런 차영상의 뉴런들이 입력 뉴런에 대한 지연치와 연계되는 것이다. 이렇게 은닉층으로 입력된 값은 각 입력 뉴런에 대한 방향 성분을 가지게 된다.

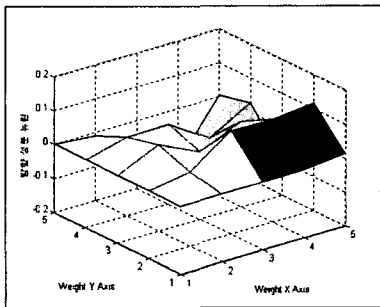
시간 지연 신경망의 출력층은 모두 24개로 구성되어 있다. 24개의 각 출력값은 같은 방향 성분을

갖는 은닉층의 가중치의 합으로 나타난다. 각 입력 뉴런의 방향 성분을 가지고 있는 은닉층이 같은 방향 성분을 가지는 출력값으로 나타나게 되는 것이다. 그 결과로 출력층의 값들은 현재 뉴런 차영상을 기준으로 과거 뉴런 차영상의 지연치와 연계되어 차영상의 방향 성분이 합쳐져서 나타난다.

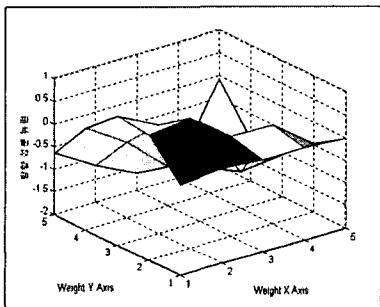
5.3 표적 물체의 경로 추정

본 논문에서는 보다 정확한 표적 물체의 경로 추정을 위하여 물체 하나만을 대상으로 하였다.

연속된 3개의 프레임의 영상에서 표적 물체의 경로는 TDNN의 출력층의 결과값으로 추정이 가능하다. 24개의 방향 성분으로 구성된 TDNN의 출력층의 각각의 값이 표적 물체의 이동 경로를 나타내게 된다.



(a) 첫 번째 입력 뉴런에 대한 방향 성분 뉴런의 가중치



(b) 세 번째 입력 뉴런에 대한 방향 성분 뉴런의 가중치

그림 4. TDNN의 각 입력 뉴런에 대한 방향 성분 뉴런의 가중치

시간 지연 신경망의 학습에는 back propagation 방법을 사용하였다. back propagation 방법은 다층의 구조를 갖는 복잡한 신경망 학습 방법으로 최급 하강법(gradient descent method)을 기본으로 한 방법이다. 학습의 활성화 함수로는 sigmoid 합

수를 사용하였다. back propagation 방법으로 학습이 끝난 후 학습된 가중치를 가지고 위와 같은 과정을 반복한다면 표적 물체의 이동 경로 추정이 가능하다.

그림 4에서는 오른쪽 위 오른쪽 위로 이동하는 표적 물체에서 TDNN의 각 입력값에 대하여 이웃하는 24개의 과거 뉴런 차영상과의 곱으로 구해지는 방향 성분 뉴런의 가중치를 나타내었다. 각각의 입력에 대하여 이러한 24개의 방향 성분 뉴런을 가지고 있으며, TDNN의 출력층은 각각의 입력에 대한 24개의 방향 성분의 합으로 나타난다.

6. 실험 및 결과

본 논문은 연속적으로 입력되는 영상을 실시간으로 처리하여 표적 물체의 중심점 탐지와 표적 물체의 이동 경로를 추정하는데 있었다. 실험은 Matlab을 이용한 시뮬레이션과 실제 영상 시스템을 대상으로 하였다.

6.1 Matlab을 이용한 시뮬레이션

Matlab 시뮬레이션에 사용된 영상은 뉴런 차영상이 발생하는 조건하에서 일정한 간격으로 여러 가지 방향으로 임의적으로 표적 물체를 움직인 임의의 화상을 사용하였다. 표적 물체는 등속도 운동을 한다는 가정에 부합되게 함으로써 물체의 중심점 탐지 및 이동 경로 추정이 가능하였다.

표적 물체의 크기가 상당히 큰 경우에는 물체 크기에 따라 초기 뉴런들이 표적 물체를 충분히 대표할 만큼의 정보를 가지고 있어 좋은 결과를 얻었다. 또한 표적 물체에 대한 뉴런들의 사영을 구할 때 코호넨 네트워크의 반복 횟수가 많을수록 더욱 표적 물체의 중심점에 근접할 수 있었다. 하지만 초기 격자화된 가중치의 간격이 클수록 표적 물체에 대한 사영이 불충분하였고, 코호넨 네트워크의 반복 횟수가 많아질수록 계산 시간이 길어지고 중심점 오차의 비율이 좋지 않았다. 그래서 본 논문에서는 실시간 처리를 목적으로 반복 횟수를 단 한번만 하였다. 그리고 코호넨 네트워크의 학습율이 높으면 물체의 중심점 탐지에서는 표적 물체에 대한 승자 뉴런 및 인접 뉴런들이 표적 물체의 중심점에 매우 근접하였지만 물체의 이동 경로 추적에서는 높은 학습율로 인해 현재 뉴런 차영상과 과거 뉴런 차영상이 거의 흡사하게 나타나서 오히려 역효과를 나타내었다.

6.2 실제 영상 시스템의 적용

본 논문에 사용된 영상 시스템은 Pentium II 350Mhz CPU, 128M RAM의 컴퓨터와 영상 입력 장치는 CCD 카메라, 영상 획득 장치는 30 frame/sec의 속도를 갖는 MyVision99를 사용하였

다. 알고리즘 구현을 위하여 MyVision99 Function Library와 Visual C++을 사용하였다.

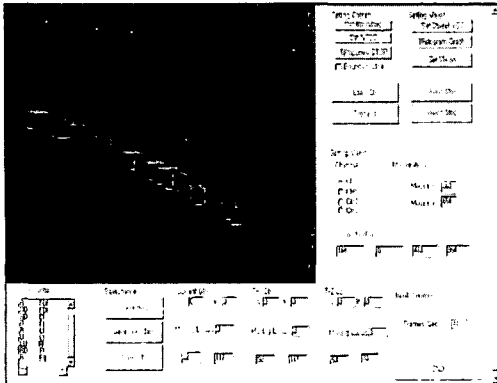


그림 5. 실제 영상 시스템의 구성

표적 물체는 30 x 30 픽셀 크기의 흰색으로, 배경은 검은색으로 Gray Color를 사용하였다. 영상은 640 x 480을 사용하였으나 국지 검색 방법(local searching method)을 이용하여 계산량을 줄일 수 있었다. 화상의 전처리 과정은 평활화 필터링을 사용하였다. 화상은 빛에 의해 명암값이 많이 바뀌므로 가장 적절한 임계값(threshold)을 구하기 위해 직접 화상을 보면서 결정하였다.

(1) 물체의 중심점 탐지 및 추적 실험

물체의 중심점 탐지에서 한 프레임에 하나의 표적 물체가 있을 때는 매우 정확한 결과를 얻을 수 있었던 반면에 두 개의 물체가 있는 경우에는 비교적 앞의 전자에 비하여 중심점 오차가 발생하였다. 또한 후자의 경우는 전자에 비하여 계산 시간을 많이 요구하기 때문에 30 frame/sec로 입력되는 영상에 대하여 모두 결과를 내지는 못하였다.

표적 물체의 중심점 탐지 및 추적의 실험에서 표적 물체가 등속도 운동을 하는 구간에서는 매우 부드럽게 중심점을 탐지하였으나 등속도 운동을 하지 않는 구간과 물체가 정지해 있거나 빠른 이동이 있는 구간에서는 표적 물체의 중심점 오차가 커졌으며 찾지 못하는 곳이 발생하였다.

그림 6에서 시뮬레이션 한 것과 같이 표적 물체 하나만을 놓은 경우와 물체 두 개를 놓은 경우에 대하여도 실험을 하였다. 물체의 하나만을 실험하였을 때는 뉴런 차영상이 발생하지 않는 상황이 아닌 경우를 제외하고는 거의 근사하게 표적 물체를 추적하였다. 하지만 두 개의 물체를 놓고 실험하였을 때는 중심점을 찾지 못하는 구간이 많이 늘어났으며 중심점의 오차 또한 크게 나타났다. 그 이유는 30 frames/sec로 입력되는 영상을 처리하는데 한 개의 물체가 있는 경우보다 두 개의 물체

가 있는 경우가 뉴런들의 사영(projection)을 구하는데 있어서 계산 시간의 요구가 배로 늘어났기 때문이다.



(a) 한 개의 물체에서의 표적 물체 추적



(b) 두 개의 물체에서의 표적 물체 추적

그림 6. 실제 영상 시스템에서의 표적 물체 추적

실제 시스템에서 중심점 탐지 및 추적의 실험은 Matlab을 이용한 시뮬레이션에 비하여 정확하게 표적 물체의 중심점을 구하지 못하였으나 근사한 결과를 가지고 물체의 추적은 가능하였다.

(2) 물체의 이동 경로 추정 실험

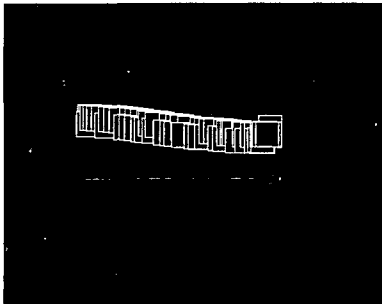
물체의 이동 경로 추정을 위해서는 연속적으로 입력되는 3개 프레임 영상들의 뉴런 차영상 2개를 필요로 한다. 뉴런 차영상 2개를 시간 지연 신경망의 지연치와 연계된 입력으로 하여 각 입력에 대하여 24개의 방향 뉴런 성분의 값을 구한 후, 표적 물체 전체의 방향 성분을 구할 수 있게 된다. 물체의 이동에 따라 시간 지연 신경망의 출력값은 뉴런 차영상의 각각의 방향 성분의 합이므로 움직임이 있는 정보를 모아 놓은 것이라고 할 수 있다. 이러한 특성이 표적 물체의 이동 경로를 추정할 수 있게 한다.

실제 시스템에서의 학습은 back propagation 방법으로 학습을 하였다. 하지만 학습 영상을 얻는 것은 시뮬레이션에서처럼 임의로 목표 물체의 이

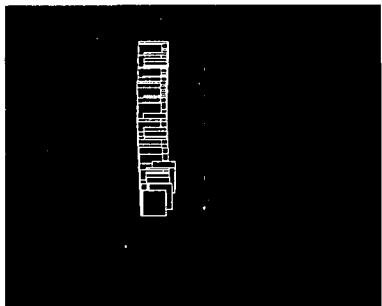
코호넨 네트워크 및 시간 지연 신경망을 이용한 움직이는 물체의 중심점 탐지 및 동작특성 분석에 관한 연구

동을 조작할 수 없기 때문에 표적 물체를 여러 방향으로 움직이면서 입력 영상을 bitmap file로 저장하여 학습 영상으로 사용하였다. 이렇게 얻은 영상에 대하여 TDNN의 출력값을 구하고 그 출력값을 다시 back propagation 방법으로 학습을 시켜서 학습된 가중치를 얻었다. 이 부분은 Visual C++로 구현하는데 어려움이 있어서 Matlab을 이용하였다. 학습된 가중치를 Visual C++로 구현한 프로그램에서 파일로 읽어서 실제 영상에 구현 TDNN의 출력값을 적용함으로써 물체의 경로 추정이 가능하였다.

이동할 때의 이동 경로 궤적을 그린 것이다.



(a) 표적 물체가 오른쪽으로 이동

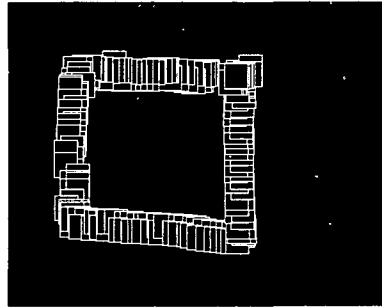


(b) 표적 물체가 아래쪽으로 이동

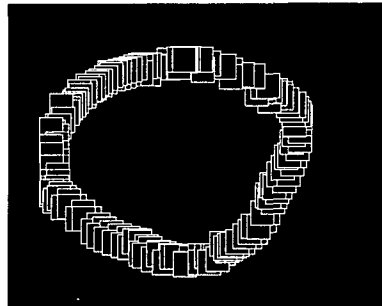
그림 7. 이동 경로 추정에 따른 표적 물체의 궤적

표적 물체의 이동 경로 추정은 연속적으로 입력되는 영상에서 3개의 프레임마다 위와 같은 방법으로 계산하여 결과를 얻었다. 추정된 결과를 나타내는 방법으로 3개의 프레임마다 추정된 결과에 따라 표적 물체의 중심점을 표시하였다. 즉, 표적 물체의 이동 경로 추정의 결과가 오른쪽 오른쪽인 결과를 얻으면 표적 물체의 시작 중심점에서 오른쪽으로 움직인 중심점을 일정한 간격으로 두 번 나타내었다.

그림 7은 실제 영상 시스템에서 위와 같은 이동 경로 추정 방법을 사용하여 표적 물체가 왼쪽에서 오른쪽으로 이동할 때와 위에서 아래쪽으로



(a) 표적 물체가 사각형 모양으로 이동



(b) 표적 물체가 원 모양으로 이동

그림 8. 표적 물체의 이동 형태에 따른 이동 경로 추정

그림 8은 표적 물체가 왼쪽에서 오른쪽으로, 위에서 아래쪽으로 움직이는 직선 형태의 움직임이 아니라 원 형태로, 사각형 형태로 움직이는 경우에 표적 물체의 이동 경로를 추정한 것이다.

경로 추정이 가능한 범위는 TDNN의 출력값이 24개의 노드(node)로 구성되어 있기 때문에 방향 성분이 24가지가 된다. 2개의 프레임의 영상에 대해서는 $0^\circ, 45^\circ, 90^\circ, 135^\circ, 180^\circ, 225^\circ, 270^\circ, 315^\circ$ 의 모두 8가지 방향 성분을 구할 수 있으며, 3개의 프레임의 영상에 대해서는 $0^\circ, 27^\circ, 45^\circ, 63^\circ, 90^\circ, 117^\circ, 135^\circ, 153^\circ, 180^\circ, 207^\circ, 225^\circ, 243^\circ, 270^\circ, 297^\circ, 315^\circ, 342^\circ$ 의 16가지 방향 성분으로 나타난다. 즉 3개의 프레임의 영상에 대한 표적 물체의 이동 경로는 $0^\circ \rightarrow 0^\circ$ (오른쪽 오른쪽으로 이동), $0^\circ \rightarrow 90^\circ$ (오른쪽 위쪽으로 이동), $0^\circ \rightarrow 342^\circ$ (오른쪽 아래쪽으로 이동), $90^\circ \rightarrow 0^\circ$ (위쪽 오른쪽으로 이동), $90^\circ \rightarrow 90^\circ$ (위쪽 위쪽으로 이동), $90^\circ \rightarrow 180^\circ$ (위쪽 왼쪽으로 이동) 등으로 추정할 수 있게 된다. 하지만 본 논문에서는 물체의 이동 경로가 매우 많아서 학습 영상을 얻는 데도 많은 시간이 필요하기 때문에 24가지의 경우에만 학습 영상을 구하여 실험하였다.

표적 물체의 경로 추정의 실험에서는 표적 물체의 이동에 의한 경로에 근사한 결과를 얻을 수 있었으나 정확한 결과를 얻을 수는 없었다. 표적이 등속도로 이동하는 구간에서는 표적 물체의 이동 경로와 근접하게 추정이 가능하였지만 그렇지 않은 구간에서는 학습되지 않는 입력에 의하여 전혀 엉뚱한 결과가 나오기도 하였다. 물체의 경로 추정은 시뮬레이션의 결과와 마찬가지로 표적 물체가 등속도 운동을 하고 있는 구간에서는 만족할만한 결과를 얻을 수 있었으며 초기 격자화된 뉴런들의 격자 사이의 간격이 좁을수록, 표적 물체의 크기가 클수록 성능이 향상되었다. 이것은 표적 물체에 대한 움직임의 정보가 많을수록 좋은 성능을 나타내는 것이다. 이러한 문제점은 표적 물체에 대한 사영(projection)을 취할 때 초기 격자화된 뉴런들의 간격과 코호넨 네트워크(Kohonen Networks)의 학습율(learning rate), 정확한 학습 영상의 획득으로 개선될 수 있다.

물체의 경로 추정에서 가장 큰 문제점은 계산 시간이었다. 기존의 명암값에 의한 방법에 비하여 코호넨 네트워크의 계산량을 감안하더라도 계산량이 줄어든 것은 사실이다. 하지만 실제 실험에서 실시간으로 처리할 수 있는 속도는 10 frames/sec에 해당하였다. 즉 30 frames/sec의 입력으로 들어오는 화상에서 1/3밖에 처리하지 못한다는 것이다. 표적 물체의 이동 경로를 추정하는 부분 중에서 back propagation에 의하여 많은 계산 시간을 소비하고 있다는 점과 물체의 중심점 탐지 및 추적을 하는 부분을 이동 경로 추정을 하는 동안 같이 처리했었다는 점이 시스템 성능저하의 원인이었다. 표적 물체의 이동 경로 추정을 하는 부분 중에서 back propagation에 의한 방법을 개선하고, 표적 물체의 중심점 탐지하는 부분을 따로 처리한다면 시스템의 성능이 향상될 것이다.

7. 결론

본 논문은 움직이는 물체의 중심점 탐지 및 추적, 동작 특성 분석(이동 경로 추정)을 실시간으로 처리하는 시스템의 구현을 목적으로 하였다. 기존의 명암값에 의한 물체의 추적 및 움직임 감지를 하는 방법에 비하여 비교적 계산량을 줄일 수 있었다. 하지만 그 만큼의 손실된 정보에 의하여 표적 물체의 움직임에 대한 근사한 값으로 만족하여야 했다.

본 논문에서 구현한 시스템의 개선해야 할 점은 세 가지로 정리할 수 있다. 첫 번째로 차영상을 기반으로 하였다는 점이다. 물론 차영상 기법이 표적 물체의 동작 특성을 나타내기에는 매우 확실한 방법이지만 차영상이 생기지 않는 조건하에는 결과를 얻기 힘들었다. 따라서 차영상 기법은

장점인 동시에 단점이 되는 것이다. 두 번째로 확실한 학습 영상의 획득이다. 구현한 시스템이 아무리 완벽하다고 하더라도 잘못된 학습 영상으로 학습을 하면 당연히 시스템의 성능이 저하된다. 학습 영상의 획득은 신경망 모델의 사용에서 매우 중요한 부분이다. 마지막으로 계산 시간의 개선이다. 대부분의 영상 처리 알고리즘이 많이 진행되어 왔었지만 성능이 좋아도 계산 시간을 너무 많이 소비하여 버리면 실시간 처리 시스템에서의 사용이 불가능하다. 따라서 계산 시간의 단축이 매우 중요하다.

참 고 문 헌

- [1] 차의영, "신경회로망을 이용한 다중 표적 추적에 관한 연구" 한국과학기술재단 과제번호 : 961-0901-010-2, pp. 6, 1998
- [2] B.K.P. Horn and B.G. Schunk, "Determining optical flow" *Artificial Intelligence*, vol. 17, pp. 185-203, 1981
- [3] H. Shariat and K.E. Price, "Motion estimation with more than two frames" *Patt. Anal. Machin Intell.*, vol 12, pp. 417-434, 1990
- [4] M. Sonka and V. Hlavac and R. Boyle, "Image Processing, Analysis and Machin Vision", *Chapman & Hall*, pp. 507-547, 1995
- [5] R.M. Kuczewski, "Neural network approaches to multitarget tracking" *Proc. of IEEE ICNN*, vol. 64, pp. 619-633, 1987
- [6] Gail A. Carpenter and Stephen Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machines" *Computer vision, Graphics and Image Processing* 37, pp. 54-115, 1987
- [7] 장동식, 이만희, 이종원, "시간 지연 신경망을 이용한 동작 분석" *제어 자동화 시스템 공학회지*, vol. 5, pp. 3-5, 1999
- [8] 이현엽, 문경일, *MATLAB을 이용한 퍼지-뉴로*, 아진, 1999
- [9] 최형일, 이근수, 이양원, *영상처리 이론과 실제*, 홍릉과학출판사, 1997