

스테레오 비전을 이용한 움직임 검출

권 창 일, 원 성 혁, 김 민 기, 이 기 식, 김 광 택, 정 일 준*
고려대학교 전자 및 정보공학과, *주식회사 에스원
전화 : 0415-860-1352 / 팩스 : 0415-862-1527

Motion detection using stereo vision

Chang Il Kwon, Sung Hyuk Won, Min Gi Kim, Gee Sik Lee, Kwang Taek Kim, *Il Jun Jeong
Department of Electric and Information Engineering, Korea University,
*S1 Corporation
E-mail : node2@hard.korea.ac.kr, mgkim@hard.korea.ac.kr, techwon@kebi.com,
gslee@hard.korea.ac.kr, luster91@hotmail.com, ijeong@samsung.co.kr

Abstract

Almost vision application systems use 2-D information by taking only one camera. Recently it arises to utilize 3-D information, which is distance from camera to object, because 2-D information is not sufficient. Therefore, we take stereo camera system.

In motion detection algorithm using stereo vision, it operates like one camera system, which takes advantage of correlation, edge, and difference algorithm, when it detects any motion.

At that time, to detect motion, it compares two images, which is from two cameras, to calculate disparity that contains distance information. By disparity, it can compute real distance and size of object information.

We describe a motion detection algorithm which computes 3-D distance and object size in real time.

I. 서 론

현재까지는 실시간 처리가 요구되는 시각 응용 시스템에서는 주로 하나의 카메라에서 얻은 영상을 처리하여 필요한 2차원 정보를 획득하여 많은 응용분야를 만들었다. 이는 2차원적인 정보를 함유하고 있는 데이터를 중요시 여겨 많은 응용분야를 탄생시켰던 것이다. 그러나 많은 응용분야에서 2차원적인 정보로는 충분하지 않음이 입증되었고 이에 3차원 정보추출로 방향을 선회하고 있다. 기존의 Motion Sensor와 하나의 화상 관제 카메라를 사용한 2차원 화상 Data에서 분석할 수 있는 움직임 감지보다는 3차원 화상 Data를 사용함으로써 사람이 인지할 수 있는 수준보다는 덜 하지만 지능적인 움직임 감지와 거리에 따른 크기 정보를 통합

으로써 오보율을 감소시킬 수 있다. 본 논문은 감시 카메라 시스템에서 스테레오 카메라를 사용하여 얻은 두 개의 영상(좌, 우 카메라 영상)을 통해 얻은 카메라와 물체사이의 거리 정보를 사용하여 사람 크기 정도의 물체 움직임을 구분하여 하나의 카메라를 사용할 때 나타날 수 있는 문제점을 해결하였다.

II. 움직임 검출 알고리즘

사용된 알고리즘은 잘 알려진 correlation을 기본으로 하고 있다. NCC(Normalize Cross Correlation)는

두 영상의 상관성을 '(-1) ~ (+1)'의 숫자로 나타내는 지표로 주로 영상의 인식에 많이 활용해온 알고리즘이다. 즉 두 개의 영상이 같으면 '+1', 역상의 영상이면 '-1'의 값을 갖고, 그 사이의 값들 중 '0 ~ +1'의 값은 이 영상이 얼마나 일치하는가, '0 ~ -1'은 두 개의 영상 중 하나의 영상을 역상으로 만들었을 때 이들이 얼마나 일치하는가를 나타낸다. 이 값을 두 개의 영상에 대해 구하면 이 결과가 '+1'근처의 값을 가지면 움직임이 없었던 것으로 판단하는데 이용할 수 있다.

NCC를 사용할 때 조명에 대한 가정은 빛이 전 영역을 통해 선형적으로 변화해야 한다는 것이다. 원래 이러한 가정은 현실적으로 만족하기 어려운 조건이지만 비교 화면의 크기가 작다면 이러한 가정은 어느 정도 만족시킬 수 있다. 본 알고리즘의 경우 referenced 영상과 카메라를 통해 들어온 영상의 correlation 결과로 움직임의 여부를 판별하고자 하므로 블록단위로 correlation을 구하여 이들을 종합적으로 검토해서 움직임의 여부를 판단한다면 조명의 비선형성에도 견실한 판단 알고리즘을 만들어 낼 수 있을 것이다.

위와 같이 하여 각각의 블록에 대한 correlation 값을 구한 후 0 ~ 1까지의 값 중 적절한 값을 threshold 값으로 정하고 threshold 값을 기준으로 하여 입력 영상을 binary화한다.(본 알고리즘에서는 블록 크기를 5x5, threshold 값을 0.5로 두어 correlation 값이 0.5 이상이면 움직임이 없는 것으로 판단하였다.)

그러나 correlation만을 사용할 경우 threshold 값의 변화에 따라 noise 발생이나 먼 거리에서의 움직임 검출 실패 등의 문제가 발생 수 있기 때문에 본 논문에서는 edge추출과 difference를 병행하여 사용하였다.

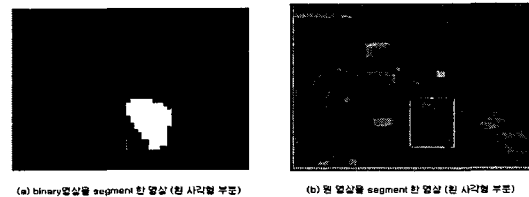
Sobel edge 추출을 사용하여 추출한 binary edge 영상, gray level에서 5x5 블록 단위의 차분 영상을 binary화 한 영상과 NCC를 사용해 획득한 영상 세 가지를 비교하여 두 개 이상의 영상에서 움직임이 있다고 판단된 부분만을 추출하여 움직임 검출 결과 영상을 만들었다.

앞의 단계를 통해 얻어진 binary 영상에서 움직임이 검출된 부분을 원 영상에서 추출하기 위해서 움직임이 있는 영역의 좌표를 찾는다. (이 알고리즘에서는 픽셀 단위의 좌표가 아닌 움직임을 있는 영역을 둘러싸는 사각형의 네 개의 모서리 좌표를 구한다.) 원 영상에서 추출된 움직임이 있는 영역은 다른 쪽 카메라에서 같

은 부분을 찾기 위해서 쓰여진다.

원 영상에서의 작업은 상당히 복잡한 과정을 필요로 하지만, 이 알고리즘에서는 이미 전 단계에서 배경과 움직임이 있는 물체를 binary 영상(배경은 검은 영역, 움직인 물체는 흰 영역)으로 구분하였으므로 보다 간단히 움직인 물체의 좌표를 찾아 낼 수 있다. 본 알고리즘에서는 histogram을 사용하여 segmentation을 수행하였다.

Histogram을 사용한 segmentation은 binary 영상을 이용한다는 점과 histogram의 특성을 가지고서 처리된다. 이진화 영상은 움직임이 있는 블록은 255값을 가지고 배경부분은 0의 값을 가지기 때문에 x축으로 255값의 분포와 y축으로 255값의 분포를 보면 움직임이 있는 물체를 감싸는 사각형의 네 개의 좌표를 구할 수 있다. 결과 영상을 x축 -> y축 -> x축으로 세 번 히스토그램을 수행하여 움직임이 검출된 부분들의 좌표를 구한다.



<그림 1> Segment한 샘플 영상

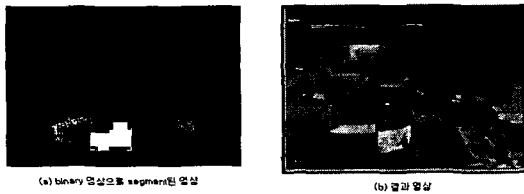
III. 경로 추적 알고리즘

경로 추적 알고리즘은 고정적으로 움직이는 물체(예 선풍기 등) 문제를 처리하기 위한 과정이다. 사무실이나 가정집 등에서 선풍기 같은 움직임이 있는 물체를 끄지 않고 시스템을 작동시키는 경우에 오보가 발생하게 된다. 이러한 경우에 침입자가 아님을 판단하고 침입자와 구분할 수 있어야 한다. 본 알고리즘에서는 물체의 움직임을 포착하고 segment 과정을 거친 후 시간상으로 연속된 영상에서 그 물체의 움직인 거리를 계산하여 이 문제를 해결하고자 하였다.

선풍기의 경우 움직임이 있지만 한곳에서 계속 같은 움직임이 반복된다. 그러나 침입자의 경우에는 한곳에만 머물러 있지 않는다. 그러므로 물체의 움직인 거리

를 계산하면 두 가지 경우를 구분 할 수 있다.

본 알고리즘에서는 segment된 사각형의 무게 중심을 기준점으로 하여 10프레임 내에 움직인 거리가 픽셀 단위로 x축으로는 40픽셀, y축으로는 25픽셀이상이 되면 고정적이 움직이지 아닌 침입자로 판단하였다. (입력 영상 크기 320x240 기준) 이 때 문제가 되는 점은 시간상으로 연속된 영상에서 segment된 움직인 물체의 사각형의 크기가 항상 일정하지 않다는 것이다. 사각형의 크기가 변하게 되면 사각형의 중심점이 변하게 되므로 고정적으로 움직이는 물체의 움직인 거리도 증가하게 된다. 이러한 문제를 방지하기 위해 사각형의 크기 변화를 함께 계산한 후 사각형의 크기 변화가 움직인 거리 보다 작은 경우에는 움직인 거리에서 크기 변화량을 뺀 값을 실제 움직인 거리로 간주하였고 크기 변화가 움직인 거리 보다 큰 경우에는 움직이지 않은 것으로 간주하여 계산하였다.



<그림 2> 고정적 움직임으로 간주된 물체 영상

IV. 스테레오 정합

스테레오 정합은 좌우 영상의 동일점을 찾는 작업으로 단순화 될 수 있으며, 이때 가능한 해의 범위를 줄임으로서 계산 오차를 줄이고 시간을 상당히 줄일 수 있다. 해의 범위를 줄이는 방법은 <그림 3>과 같이 극상선(epipolar line)이 수평 주사선과 평행하게 두 카메라를 설치함으로써 대응점 탐색의 범위를 영상 전체에서 수평 주사선 내로 제한 할 수 있다.

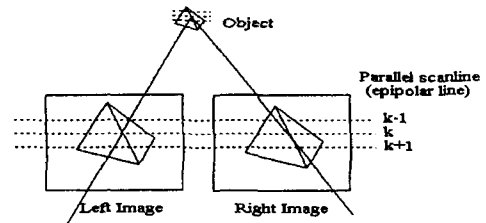
본 논문에서는 정합 오류를 배제하고 변이의 연속성을 유지하면서 신뢰성 있는 정합을 위해 위에서 언급된 제한 조건과 움직임이 생긴 부분에 대한 영역만을 이용한 정합을 제안한다.

좌우 화소는 하나의 일치점을 갖는다고 가정하면 하나 하나의 화소가 모인 영역의 disparity 평균값과 움직임을 추출한 영역 자체의 disparity를 찾는 것은 의

미가 같다고 할 수 있다. 이는 정확한 의미의 Area-Based Stereo Matching 방법은 아니나 Area - Based Stereo Matching을 통한 하나 하나의 화소를 찾는 방법보다 빠르게 disparity 값을 구할 수 있는 장점이 있다.

움직임이 있는 물체에 대해서 화소단위로 Segment가 이루어진 것이 아니기 때문에 Block Matching을 통한 disparity를 구하는 데 있어서 배경이 정합의 오류를 만들 수 있다. 그래서 Epipolar Line 상에 물체가 있다고 가정하고 기준 영상과 현재 영상의 차 영상을 가지고 matching을 하게 된다.

오른쪽 카메라에서 잡히는 Motion의 위치와 Epipolar Line 상의 왼쪽 카메라에서 정합이 되는 부분의 위치를 Mean of absolute difference를 구하여 최소가 되는 값의 위치를 구하고, 두 영역의 x 좌표의 차이를 disparity로 판단한다. 즉 움직인 물체의 disparity로 판단한다.



<그림 3> 스테레오 정합을 위한 제한 조건

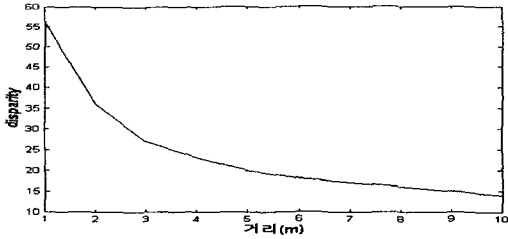
$$MAD = \frac{1}{Block\ Size} \sum_{(x,y) \in B} | I_1(x,y) - I_2(x,y) |$$

V. Disparity에 따른 거리와 크기 계산

Effective focal length는 거리에 따라서 가변적으로 변하기 때문에 disparity와 거리의 관계가 반비례 한다는 것은 알 수 있지만 실제 거리를 유추해내기란 힘들다. 실제로 10m이내의 거리에서 18cm가 떨어진 병렬로 구성된 Stereo Camera로 거리에 따른 disparity를 구해보면 <그림 4>와 같다.

<그림 4>와 같이 10m이내의 거리에서 기준이 되는 물체를 1m씩 거리를 점점 멀게 하면서 Disparity를 구하고 이에 따른 Table을 만든다. 그리고 거리에 따른 크기 값도 구하여 Table을 만든다. 단 물체와 camera

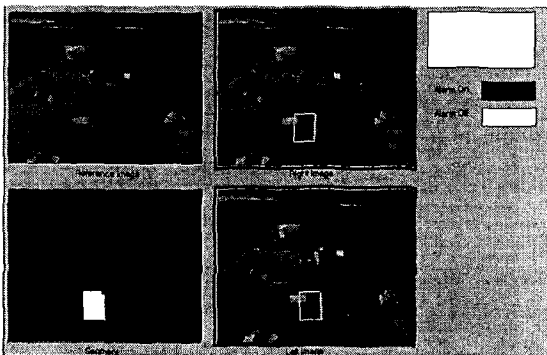
의 거리가 1m이내 일 때는 10cm마다 disparity를 구하면서 크기에 대한 Table을 만들어 주어야 한다.



<그림 4> Disparity와 거리와의 관계

움직임이 있는 물체에 대하여 stereo camera를 이용하면 근사적인 거리와 크기 정보를 알 수 있게 된다. 그리고 이 자료를 가지고 침입자인지 아닌지를 판단하는데 우선 크기의 가로와 세로가 Threshold 값 이상이면 침입자로 판단을 하고 다음의 단계로 넘어 간다. 다음은 크기정보의 가로 세로 비가 정상적인 침입자의 비율인지 아니면 작은 동물의 가로 세로 비인지를 판단하고 비정상적인 침입자(기어서 움직이는 경우나 쪼그려서 움직이는 경우 등)의 비율과 비교하여 침입자를 가려낸다. 즉 크기정보와 침입자의 가로세로 비를 판단의 근거로 삼는다. (이 실험에서는 실제 높이가 1m 이상이거나 길이가 1m 이상인 경우에 경보가 울리게 하였다.)

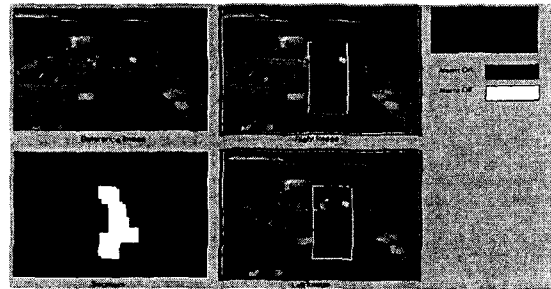
VI. 실험 결과 및 결론



<그림 5> 작은 물체의 움직임 실험 결과 I



<그림 6> 작은 물체의 움직임 실험 결과 II



<그림 7> 사람이 침입한 경우 실험 결과

작은 물체의 움직임이 있을 경우와 사람이 침입했을 경우의 결과를 그림 5,6,7에서 볼 수 있다. 이 상에서와 같이 그림 6에서는 그림 7의 사람의 크기보다 영상에 잡힌 물체는 크기가 크지만 3차원 정보를 이용하여 실제 크기를 계산하여 실제 크기가 작다는 것을 알아 내어 경보가 발생하지 않는 것을 볼 수 있다.

참고 문헌

- [1] Oliver Faugera "Three-Dimensional Computer Vision A Geometric Viewpoint", The MIT Press, 1993
- [2] David Marr, "Vision", W.H. Freeman and Company New York, 1982
- [3] David Marr and T. Poggio, "A Computational theory of human stereo vision", Proc. Royal Society of London. B.204, pp. 301-328, 1979
- [4] Anil K. Jain, "Fundamentals of Digital Images Processing", Prentice Hall
- [5] J.P.Lewis, "Fast Normalized Cross_Correlation", Industrial Light & Magic, <http://www.yeeyoga.com/~zilla/Papers/nvisionInterface/nip.html>