

스테레오 영상의 실시간 주시각 제어 장치 구현

백승호⁰ 김균수 김석민 김재한
호남대학교 전자공학과 3차원 영상 연구실
baeksh@magicn.com⁰ negro2814@hanmail.net ehftlr@hanmail.net kjh@honam.ac.kr

Implementation of Real-time Vergence Control System for Stereoscopic Camera

Seung-Ho Baek⁰ Kyun-Su Kim Seok-Min Kim Jae-Han Kim
3D Vision Lab., Dept. of Electronic Eng., Honam University

요 약

입체영상을 관측할 때 눈의 피로를 최소화할 수 있는 자연스러운 입체영상을 얻기 위해서는 양안식 카메라의 주시각을 제어하여야 한다. 움직이는 물체에 대한 양안식 카메라의 주시각 제어를 위해 획득 영상의 edge 정보를 이용한 주시각 제어에 대하여 연구하였다. 양안식 입체 카메라로 획득한 좌·우 영상에서 edge정보를 추출하고, 이를 이용하여 양안시차를 구하고 주시각 제어에 이용함으로써 영상의 변화와 관계없이 양안시차를 항상 일정하게 유지할 수 있도록 하였다. 본 논문에서는 간단한 연산만을 사용함으로써 응답속도가 빠르고, 하드웨어 구성도 매우 간단해 지도록 하였다. 또한 영상의 특성에 맞는 알고리즘을 사용할 수 있도록 2 mode 동작 시스템으로 구성하여 오차를 최대한 줄이며 움직이는 물체에 대한 연속적인 주시각 제어를 할 수 있도록 구현하였다.

I. 서 론

인간의 눈과 같은 입체 영상장치에 대한 많은 연구가 진행되고 있다. 인간의 눈과 같이 입체 영상을 얻기 위해서는 두 대의 또는 그이상의 카메라를 사용하여야 한다. 두 대의 카메라를 사용할 경우 이를 양안식 입체카메라라고 한다. 이러한 양안식 입체 카메라에서 두 대의 카메라 사이의 각도를 주시각이라 하며, 획득된 좌·우 영상에서 동일 물체의 영상내 위치 차이를 시차(disparity)라 한다.

양안식 입체 카메라는 일반적으로 입체영상을 획득할 때 두 대의 카메라를 평행으로 배치하여 영상을 획득하는 평행식(horizontal moving axes) 방법과 카메라를 서로 교차되게 배치하는 교차식(crossed axes) 방법의 두 가지 형태로 나눌 수 있다.

본 논문에서 다루는 교차식은 두 대의 카메라를 인간의 눈처럼 서로 교차되도록 하는 방법으로 이동하는 물체를 따라 카메라를 회전시켜 주시각을 제어하는 방식을 사용하였다. 평행식 방법보다 인간의 눈과 더욱 유사한 구조이며 목표물을 추적 또는 주시각을 제어하는 등의 동작에 있어 평행식 방법보다 우수하다.

인간의 눈과 유사한 기능을 갖기 위해서는 카메라의 유동적인 움직임과 함께 실시간으로 주시각의 제어가 가능하여야 한다. 즉, 관측물체의 거리변화에 관계없이 획득된 좌·우 영상에서 시차를 '0' 또는 일정한 값으로 고정하여, 입체영상을 관측할 때 눈의 피로를 최소화할 수 있도록 자연스러운 입체영상을 얻어야 한다.

입체 카메라를 이용한 시스템은 원격 의료, 자동 주행, 산업 자동화 시스템, 로봇의 3차원 물체 인식 등의 장치에 이용될 수 있다. 이러한 시스템이 실시간 자동 주시각 제어 기능을 갖는다면 인간의 눈과 비슷하게 자연스러운 입체영상을 얻을 수 있는 시스템을 구현할 수 있다.

입체 카메라의 주시각 제어 연구의 목표는 궁극적으로 사람의 눈과 유사한 시각장치를 개발하는 것이다. 그러한 측면에서 많은 연구가 진행되어 왔으나 입체 영상 처리의 복잡성으로 인해 아직 많은 연구가 필요한 분야이다.

주시각 제어에 대한 연구는 신호의 cepstrum을 이용한 cepstral 필터를 사용한 주시각 제어[1], 배경시차 보정을 이용한 스테레오 시각장치의 주시각 제어[2], 카메라 사이의 수평거리에 의한 주시각 제어[3]등이 이루어졌다. 이러한 시스템

들은 수식의 복잡성으로 인하여 하드웨어가 복잡해지며 응답속도가 느려 실시간으로 주시각 제어하는데 어려운 면이 있다.

II. 양안시차 획득 및 주시각 제어

본 논문에서는 입체 시각 시스템의 주시각 제어를 위해 움직이는 물체 또는 배경에 대한 edge정보를 이용하여 시차정보를 산출하였다.

즉, edge는 두 카메라의 focus가 일치하는 지점의 특징이라고 말할 수 있다. Focus가 맞지 않는 지점은 잡음제거 처리, edge 강조 연산 등에 의해 edge로 나타나지 않기 때문에 관측 중심 물체만의 edge를 구할 수 있다. 이렇게 구한 edge를 이용하면 영상의 모든 정보를 사용하지 않으며 계산도 간단하기 때문에 다른 방법에 비해 빠른 응답속도를 기대할 수 있고 시스템 장치도 간단하게 구성할 수 있다.

1. 알고리즘

일반적인 환경에서의 edge는 실내와 실외에서 많은 차이를 보인다. 실내환경에서는 인공물 등의 영향으로 인해 직선적인 성분의 edge가 뚜렷하며 많이 나타나게 된다. 그러나 실외환경에서는 특히 인공물이 많지 않은 산이나 나무가 있는 정원 같은 곳에서는 직선적인 성분의 edge가 별로 나타나지 않는다. 이러한 차이 때문에 본 논문에서는 실내환경의 영상과 실외환경(인공적인 구조물이 많은 경우 실내환경의 영상으로 분류)의 영상에 대해 각각 적합한 알고리즘으로 접근하고자 하였다.

실내환경 영상의 경우 수직방향 edge만 사용하여 이를 projection하고 projection profile을 correlation 하여 disparity를 산출한다. 수직방향의 edge 만으로도 disparity를 계산하기에는 충분하며 data양을 줄여 disparity 산출 계산에 소요되는 시간도 줄일 수 있기 때문이다.

실외환경 영상의 경우에는 수직·수평 방향의 edge 모두 검출하여 edge를 직접 correlation하는 알고리즘을 적용하였는데, 획득된 영상에서 특징적인 edge가 나타나지 않는 경우 좌·우 projection profile이 전혀 다르게 나타나거나, projection profile을 구한 후 correlation 연산을 취하면 실제와 다른 부분에서 최대 값이 나타날 수 있기 때문에 이를 회피하기 위한 방법으로 좌·우 각각의 edge 영상에 대해 2차원 edge값을 직접 correlation 하여 오차를 줄일 수 있도록 한다.

1.1 Edge 검출

뚜렷한 직선적인 성분의 edge가 많은 실내환경의 영상에 대해서는 수직방향의 edge 만을 검출하고, 그렇지 않은 실외환경의 영상에 대해서는 수직·수평방향 edge 모두 검출한다. 이때 threshold를 두어 흐리고 불분명한 edge들은 제거하고 뚜렷한 edge 만을 추출한다.

대부분의 영상에서 관심물체는 중앙부분에 위치하며 좌·우 영상의 가장자리 부분에서 좌측영상에서는 보이지 않는 물체가 우측영상에서는 보이거나 또는 반대의 현상이 나타날 수 있다. 그렇기 때문에 그러한 부분의 영향을 줄이고 좌·우 각각의 영상에서 중앙부분을 강조하고 가장자리 부분을 제거하기 위해 Gaussian weighted window를 적용한다.

1.2 Edge 값의 수직 projection

Edge 영상에서 edge 값을 수직 방향으로 projection 한다. Projection profile을 분석하여 어떠한 환경에서의 영상인지 판별한다. Edge line의 pixel 비교를 통해 disparity를 구할수 있으나, 이 방법은 큰 오차를 유발하므로 여기에서는 각 edge 값을 수직으로 projection 하여 그 결과를 비교함으로써 disparity를 구한다.

1.3 Correlation 연산

실내환경 영상의 경우 edge 영상의 좌·우 projection profile에 대하여 correlation 연산을 하여 disparity를 구한다. 그러나 실외환경의 영상에 대해서는 위와는 달리 edge값을 projection 하지 않고 2차원 edge 값을 그대로 사용하여 수평 방향으로 correlation 연산을 한다. 이때 좌·우 영상이 일치하게 되는 위치에서 correlation 결과의 최대값이 생성되므로 그 위치값을 disparity로 취한다.

1.4 Disparity 산출

주시각을 실시간으로 제어하기 위해서는 disparity값을 빠르게 연산하여야 한다. 연속적인 영상에서는 disparity값이 급격하게 변하지 않기 때문에, 다음 영상이 획득되었을 때 앞에서의 모든 연산을 하지 않고 이전 disparity값을 기준으로 연산 구간을 설정하여 그 구간 내에서만 correlation 연산을 한다.

1.5 주시각 제어

산출된 disparity값을 '0' 또는 일정한 값이 되도록 두 카메라를 회전하여 주시각을 제어한다. 일정 시간 간격마다 disparity를 산출하고, 그 값에 따라 카메라를 조절함으로써 연속적인 주시각 제어를 한다.

1.6 2 동작 mode 선택 제어

주시각 제어 시스템에 효율이 좋은 mode로 동작하도록 구성한다. 본 동작에서는 영상 획득시에 수직 직선 성분 edge의 많고 적음을 판단하고 이를 근거로 projection profile을 correlation할 것인지 2D edge 영상을 correlation할 것인지를 결정하고 동작하는 알고리즘을 적용하여 오차를 최소화하며 빠르게 동작하도록 하였다.

2. 실험

2.1 Preprocessing

획득된 좌·우 영상 각각에 대하여 잡음의 영향을 최소화하며 뚜렷한 edge 만을 검출하기 위해 5x5 Gaussian mask를 사용하여 잡음을 제거한다. 잡음이 제거된 영상에 대해 수평방향의 양쪽 끝 부분을 window를 사용하여 제거하였다.



그림 1. 실내환경(책꽂이)의 획득영상과 잡음제거 Mask

2.2 2 mode 제어

입력된 영상에 대해 먼저 수직방향 edge를 구하고 이를 수직 방향으로 projection한후, 이 projection profile의 표준편차를 이용하여 2mode 선택 알고리즘이 적용되도록 하였다.

2.3 수직방향 edge 검출

잡음제거 처리된 그림1의 좌·우 각각의 영상에 대해 수직방향 edge 검출을 한다. 이때 threshold 이상의 pixel은 255의 값으로 고정하고 이진화 하였다. 또한 검출된 edge에 대해 Gaussian 함수를 이용하여 가중치를 적용하였다. 본 연구에서는 실험결과를 바탕으로 Sobel mask를 사용하였다.

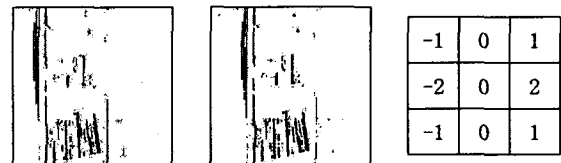


그림 2. 수직방향 edge 검출 (window : 180x256, threshold : 180, 가중치 적용, 반전) 영상과 적용된 Sobel mask

2.4 수직방향 edge 영상의 수직 projection

그림1 영상은 직선 성분의 edge가 정해진 threshold값보다 많은 실내환경 영상이므로, edge값을 수직방향으로 projection 한다. 즉, edge 영상에서 각 수직 pixel line 별로 edge에 해당하는 pixel의 개수를 산출한다.

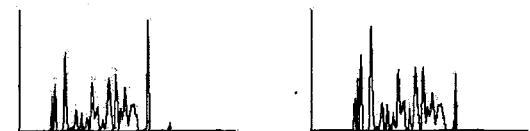


그림 3. 좌·우 Edge 영상에 대한 수직 projection profile

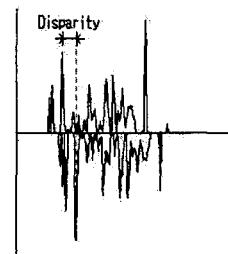


그림 4. 좌·우 projection profile의 겹침

그림3에서 수평축은 영상의 수평위치를 나타내고, 수직축은 각 수평위치에서의 edge에 해당하는 pixel의 개수를 나타낸다.

좌측과 우측을 비교하여 보면 위치와 약간의 값의 차이만 있을 뿐 비슷한 형태임을 알 수 있다.

그림 4는 disparity를 적용하지 않은 상태에서, 좌측 edge 영상에 대한 projection profile을 수평축의 위쪽에, 우측 projection profile을 아래쪽에 나타내 비교하였다

2.5 Correlation 연산에 의한 시차 추출

좌측 projection profile을 오른쪽으로 한 pixel 씩 이동시키 가며 correlation을 취한다. 이 때 좌·우 profile이 일치하게 되는 지점에서 최대값이 산출된다. 그 최대값이 산출된 지점 즉, 좌측 projection profile의 이동거리가 disparity이다.

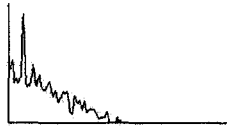


그림 5. 좌·우 projection profile의 correlation 결과

그림 6에 그림 4에서 좌·우 각각의 projection profile을 disparity만큼 보정하여 나타내었고, 그 결과로서 좌·우 profile이 거의 일치함을 볼 수 있다.

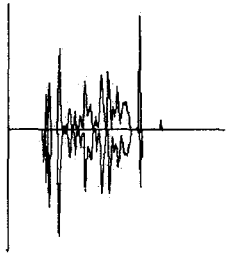


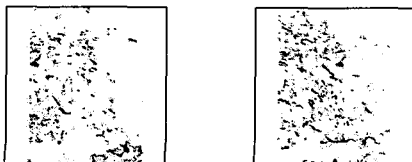
그림 6. Disparity 보정에 의한 주시각 제어 결과

2.6 전방향 edge 검출



(a) 좌측 영상 (b) 우측 영상
그림 7. 실외환경(소나무 숲)의 획득영상

그림7의 좌·우 영상이 직선 성분의 edge가 매우 적게 나타나 실외 영상으로 간주되므로 잡음제거 처리된 영상에 대하여 수평·수직 edge를 검출하였고, 2.3결과 같이 thresholding 및 Gaussian weighted window를 적용하였다.



(a) 좌측 영상 (b) 우측 영상

그림 8. Edge 검출(window : 180x256, threshold : 180, 가중치 적용, 반전) 영상

2.7 Correlation 연산에 의한 시차 추출

검출된 수평·수직방향 edge를 projection하지 않고 gray scale의 edge 값을 직접 수평 방향으로 correlation하여 disparity를 산출하였다.

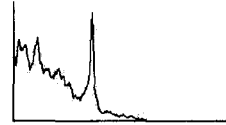


그림 9. Edge 값의 수평방향 correlation 결과

2.8 실험 결과 분석

Gaussian mask를 사용한 결과가 그렇지 않은 경우에 비해 훨씬 더 정확한 disparity를 구할 수 있었으며, 일반적인 edge 검출 mask 중 Sobel mask가 많은 영상에 대해 correlation에서 다중 peak가 발생하지 않아 좋은 결과를 얻었다.

Threshold와 중앙부분 강조를 위한 Gaussian weight를 적용하여 정확한 결과를 얻었으며, window를 사용하였을 때 그렇지 않은 경우에 비해 데이터양의 감소로 인하여 빠른 응답을 얻었다. 또한 2 mode 동작 제어와 disparity 산출시 연산구간을 적용하여 더욱 더 빠른 응답속도를 얻었다.

III. 결 론

본 논문은 양안식 입체 카메라로 얻은 영상에서 양안시차를 획득하고 이를 이용한 주시각 제어를 목표로 한다. 움직이는 물체에 대한 양안식 입체 카메라의 주시각을 항상 일정하게 유지하기 위해 edge 정보를 이용해서 양안시차를 구할 수 있게 하였으며, 사용된 알고리즘은 간단한 연산만을 사용하였기 때문에 하드웨어 구성이 매우 간단해지면서 연속적인 실시간 영상처리 구현을 실현할 수 있도록 하였다.

실내환경의 경우에는 많은 실험 영상에서 수직방향 edge만을 이용한 양안시차 산출 결과가 거의 정확하게 나왔으나, 실외환경의 영상에서는 직선적인 성분의 edge가 거의 없으므로 수평·수직방향 edge 모두를 이용하고 edge값을 직접 이용하는 방식이 유리할 것으로 예상했고 이를 실험적으로 확인하여, 영상종류에 따라 2 mode로 동작하도록 알고리즘을 구현하였다. 그 결과로, 어떤 종류의 영상일지라도 오차를 최대한 줄이면서 빠른 속도로 주시각 제어를 실현할 수 있도록 하였다.

참 고 문 헌

[1] David J. Coombs and Christopher M. Brown, "Cooperative Gaze Holding in Binocular Vision", IEEE Control Systems, pp.24-33, 1991.
 [2] 박순용 and 이용범, "배경시차 보정을 이용한 스테레오 시각장치의 주시각제어" 전자공학회논문지, 34권 S편 9호, pp.67-68, 1997.
 [3] Soon-Yong Park, Yong-Bum, Lee and Sung-II Chien, "Linear relation for Vergence control of parallel stereo camera", Electronics Letters, Vol.34 No.3, pp.255-256, 1998.