

# 원격 비전시스템을 위한 HMD의 방향각 측정 알고리즘에 관한 연구

논문  
58-3-26

## A Study on the Rotation Angle Estimation of HMD for the Tele-operated Vision System

노영식<sup>†</sup> · 윤승준<sup>\*\*</sup> · 강희준<sup>\*</sup> · 서영수<sup>\*\*\*</sup>  
(Young Shick Ro · Seung Jun Yoon · Hee Jun Kang · Young Soo Suh)

**Abstract** - In this paper, we studied for the real-time azimuthal measurement of HMD (Head Mounted Display) to control the tele-operated vision system on the mobile robot. In the preexistence tele-operated vision system, a joystick was used to control the pan-tilt unit of the remote camera system. To give the sense of presence to the tele-operator, we used a HMD to display the remote scene, measured the rotation angle of the HMD on a real time basis, and transmitted the measured rotation angles to the mobile robot controller to synchronize the pan-tilt angles of remote camera with the HMD. In this paper, we suggest an algorithm for the real-time estimation of the HMD rotation angles using feature points extraction from pc-camera image. The simple experiment is conducted to demonstrate the feasibility.

**Key Words** : Tele-operated vision system, HMD(Head Mounted Display), Labeling, Rotation angle estimation

### 1. 서론

#### 1.1 연구 개발의 배경

원격 제어(Tele-operation)란 작업 공간에 있는 로봇이나 기계와 같은 시스템을 직접 사람의 손이나 발로 조작하지 않고 어떤 장치를 이용하여 간접적으로 제어하는 것을 의미한다. 원격 제어의 적용 분야로는 공장과 같은 산업 현장이나 NASA에서 현재 진행하고 있는 우주무인탐사, 핵 처리와 관련된 원전 설비 유지보수, 공장 자동화, 심해 작업 등에서 많이 이루어지고 있다.[1] 현재 컴퓨터와 네트워크 기술의 발전과 더불어 원격제어에 관한 연구는 더욱 활발히 이루어지고 있다. 원격제어 이동로봇은 인간이 직접 작업하기에 위험하고 복잡한 환경에서의 작업이나 무인탐사와 같은 일을 하기 위해서 독자적으로 주변 환경을 인식하고 작업을 수행, 완료할 수 있는 기능을 필요로 한다.[2] 원격제어를 하기 위한 한 분야로서 원격 비전시스템은 사람의 눈과 같은 중요한 역할을 한다. 본 논문은 원격제어 이동로봇의 비전 시스템을 제어하기 위한 연구의 한 분야로 HMD를 착용한 사용자 머리의 회전을 측정하고자 하는 연구이다. 그림 1에서 보듯이 원격제어 비전 시스템은 사용자가 착용하여 원격제어 이동로봇의 주위환경을 볼 수 있는 HMD와 HMD의 회전각을 측정하기 위한 PC카메라, 이동로봇에 설

치되어 회전을 제어할 수 있는 Pan-Tilt, 이동로봇 주위환경의 정보를 파악하기 위한 Stereo Camera로 구성 되어 있고 본 논문에서는 PC카메라를 이용하여 HMD의 회전각을 실시간으로 측정하는 알고리즘을 제안하고자 한다.



그림 1 (a)PC 카메라, (b)HMD, (c)스테레오 카메라, (d)팬틸트  
Fig. 1 (a)PC camera, (b)Head Mounted Display, (c)stereo camera, (d)Pan-Tilt

#### 1.2 연구 내용 및 목표

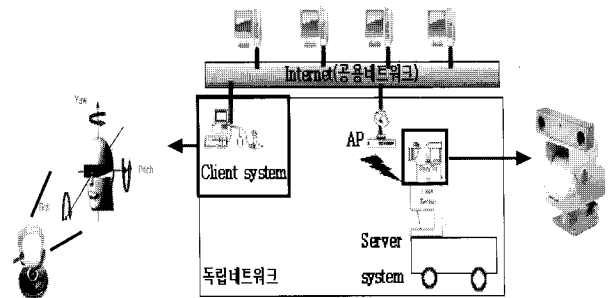


그림 2 원격제어 비전 시스템의 구조  
Fig. 2 Architecture of tele-operated vision system

<sup>†</sup> 교신저자, 정회원 : 울산대 전기전자정보시스템학과  
정교수 · 공박

E-mail : ysro@mail.ulsan.ac.kr

<sup>\*</sup> 정 회원 : 울산대 전기전자정보시스템학과 정교수 · 공박

<sup>\*\*</sup> 준 회원 : 울산대 전기전자정보시스템학과 석사과정

<sup>\*\*\*</sup> 정 회원 : 울산대 전기전자정보시스템학과 부교수 · 공박

접수일자 : 2008년 10월 30일

최종완료 : 2009년 1월 16일

본 논문은 이동로봇 원격제어 시스템에서 원격비전 시스템을 제어하기 위해 클라이언트 시스템에서 PC카메라를 사용하여 HMD를 착용한 사용자 머리의 회전각을 측정하고자 하였다. 그림 2에서 보듯이 클라이언트 시스템은 사용자가 직접 제어하는 시스템으로 원격제어기라 말하고 서버 시스템은 이동로봇을 의미한다. 기존에는 Pan-Tilt 회전을 조이스틱을 사용하여 제어하였기 때문에 위치를 컨트롤하는 점에서 불편하였다.[3] 본 논문에서는 이러한 점을 개선하고자 PC 카메라에 의해 실시간으로 사용자가 착용한 HMD의 회전각을 측정하고자 한다. 이 연구의 최종 목표는 클라이언트 시스템에서 측정된 HMD의 회전각 정보를 네트워크와 무선 랜을 통해서 이동로봇의 Pan-Tilt에 전송하여 제어하는 것이다. 회전각을 측정하는 방법에는 자이로 센서나 자기 센서를 이용하는 방법, 그리고 PC카메라로부터 얻은 이미지를 컴퓨터 비전 기술에 적용하는 방법이 있다.[4] 첫 번째로 자이로센서를 HMD에 부착하여 사용자의 머리의 방향각을 검출하는 방법은 자이로센서의 특성상 사용자의 자세를 감지할 뿐 위치를 감지할 수는 없는 한계를 가지며, 또한 HMD에 장착해야 하기 때문에 HMD가 무거워지고, 사용자의 움직임을 감지하는 속도가 느린 단점을 가지고 있다. 두 번째로 자기센서를 이용하는 방법이 있다. 자기센서는 사용자 머리의 움직임을 정밀하게 측정하는 것이 가능하나, 자이로센서의 경우와 마찬가지로 자기센서를 HMD에 장착하기 때문에 HMD가 무거워지고 주변에 금속성의 물체가 존재할 때 이로 인한 측정의 왜곡이 생길수가 있다. 세 번째로 컴퓨터 비전 기술을 이용한 기존 방법은 HMD에 카메라를 장착하고 카메라가 비추는 벽에 마커를 부착하여 이를 인식함으로써 사용자의 위치 및 자세를 감지하는 것이다. 그러나 이 방법은 앞에서 소개한 방법과 마찬가지로 HMD의 무게를 증가시키고 실내에 마커를 부착해야 하기 때문에 특별히 꾸며진 공간이외의 장소에서는 기능을 수행할 수 없다. 본 논문에서는 앞에서 소개한 방법들의 단점을 극복하기 위해 HMD에 특징점을 표시한 후 PC카메라로부터 들어오는 영상신호를 처리하고 회전각 측정 알고리즘을 적용하여 HMD에 어떤 장치도 부착하지 않고 방향각을 측정하고 있다. 앞에서 언급하였듯이 본 논문에서는 원격 비전 시스템을 구성하기 위한 초기 단계로 PC 카메라로 HMD의 3-D Pose Estimation, 즉 HMD의 roll, pitch, yaw 회전각을 실시간으로 측정하는 방법을 사용하였다. 그리고 카메라를 이용하여 방향각을 측정하는 알고리즘은 여러 가지가 있다. 그 중에서 Caltech에서 유사한 알고리즘을 이용하여 회전각을 측정할 수 있는 matlab toolbox를 제공하고 있는데 이는 각각의 이미지를 저장하여 하나씩 분석한 후 회전각을 측정하고 있다. 본 논문에서는 PC 카메라에 의해 얻은 이미지를 초당 30프레임을 처리하여 실시간으로 회전각을 측정할 수 있도록 연구하였다. 또한 카메라의 이용하여 회전각을 측정하는 부분에서는 보통 스테레오 카메라나 두 대 이상의 카메라를 사용하여 측정하는 연구들은 많이 존재한다. 하지만 본 논문에서는 한 대의 PC 카메라를 이용하여 HMD의 회전각을 측정하고자 하였다. 본 논문에서 측정 알고리즘을 간략하게 설명하면 우선적으로 회전각 측정에 필요한 8개의 특징점의 좌표를 획득하기 위해서 PC카메라로부터 얻은 영상을 이진화, 라벨링 등 영상처리 알고리즘을 적용하였다. 그리고 회

전각 측정은 획득한 8개의 특징점의 좌표를 사용하였고 cross-ratio의 불변성[5]과 중심사영[6]을 이용한 회전각 측정 알고리즘을 적용시켜 roll, pitch, yaw를 측정할 수 있었다. 본 논문의 구성은 다음과 같다. 본문에서는 좌표 획득을 위한 영상처리 알고리즘과 회전각 측정을 위한 알고리즘, 실험결과에 대해 소개하고 결론에서는 제안된 알고리즘의 유용성에 대한 언급할 것이다.

2. 본 론

2.1 좌표검출을 위한 영상처리 알고리즘

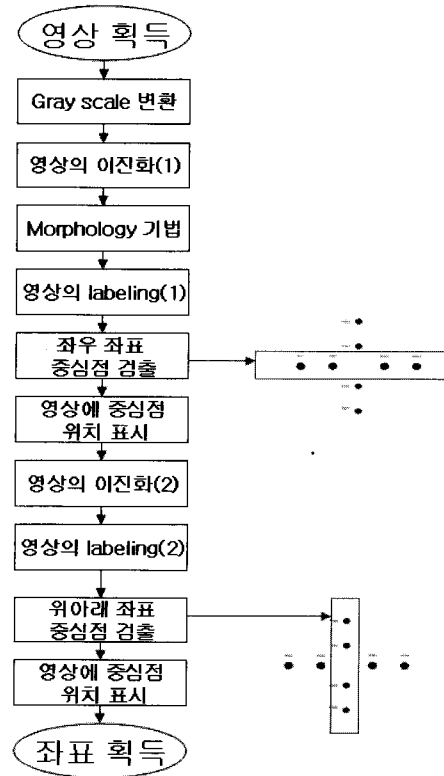


그림 3 좌표 검출 알고리즘  
Fig. 3 The Algorithm for detecting coordinates

그림 3은 좌표 검출을 위한 알고리즘이다. 영상을 획득하기 위해 Microsoft Life Cam1.4 PC카메라를 사용하였고 영상 스케일은 320×240을 사용하였다. 첫 번째로 PC카메라로부터 얻어지는 영상은 컬러 영상이므로 이를 흑백명암도의 밝기 값(Gray scale)으로 변환을 해야 한다. 변환된 그레이스케일 영상에서 이진화를 이용하여 8개의 특징점을 획득할 수 있다. 이진화 된 이미지는 배경(background)과 물체(foreground)에 노이즈를 가지고 있는데 이 노이즈를 제거하기 위해 모폴로지(Morphology) 기법을 이용하였다. 이진화시키고 노이즈를 제거한 영상은 8개의 특징점만 존재하게 되고 이들의 좌표를 얻기 위해서 이 점들을 각각 다른 라벨 값을 넣어주어야 한다. 그래서 라벨링 알고리즘을 사용하여 각각의 특징점에 라벨값을 넣어주고 좌표를 검출하였다.

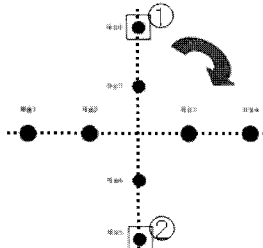


그림 4 실험을 위한 점들의 위치  
Fig. 4 The point position for an experiment

여기서 좌우와 위 아래로 위치한 점들은 각각 한 일직선 상에 구성되어 있기 때문에 그림 4를 보듯이 머리가 회전하게 되면 라벨링 되는 순서가 바뀌어서 좌표의 위치가 변하게 된다. 즉 그림 4와 같이 회전을 했을 때 점1과 점2는 라벨링 되는 순서가 변하므로 좌표의 위치도 바뀌게 된다는 의미이다. 이 에러를 제거하기 위해서 라벨링 처리를 두 번 실시하였다. 첫 번째 라벨링은 영상을 좌우로 처리한 후 중심좌표를 얻을 수 있는 알고리즘을 사용하여 좌표1~4까지 획득하였고 두 번째 라벨링은 좌표1~4까지의 그레이 값을 이진화를 적용하여 제거한 뒤 위 아래로 라벨링을 실시하여 좌표5~8을 얻을 수 있었다. 라벨링의 방법에는 Glassfire 알고리즘과 반복문을 사용한 영역 라벨링 방법이 있다.[7] Glassfire 알고리즘은 이진화 된 영상을 라벨링하는 가장 간단한 방법으로 마른 잔디(glass)에서 불(fire)이 번져나가는 모양과 비슷하게 화소를 라벨링하기 때문에 붙여진 이름이다. 이 방법은 자기호출을 이용하여 모든 인접화소를 차례로 검사하면서 라벨링하는 방법이다. 반복문을 사용한 영역 라벨링 방법은 스택을 직접 설계하여 사용하므로 스택의 크기를 프로그램 내에서 사용자가 정의하는 것이 가능하다. 이 방법은 고속으로 동작하면서 아주 큰 영역을 라벨링하는 것도 가능하다. 본 논문에서는 Glassfire 알고리즘을 사용하였다.

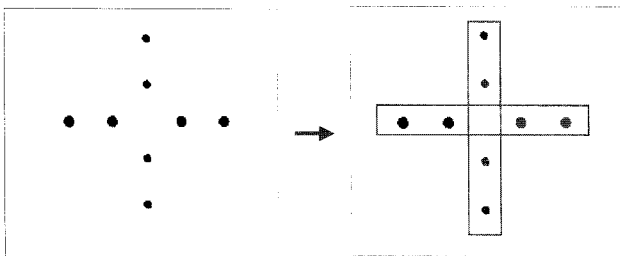


그림 5 1차 라벨링 영상  
Fig. 5 First labeling image

그림 5는 본 연구에서 필요한 좌표를 검출하기 위해 첫 번째 라벨링을 실행하였을 때 출력되는 영상이다. 첫 번째 라벨링은 그림 5를 보듯이 왼쪽에 오른쪽으로 실행하였다. 1차 라벨링 알고리즘을 적용함으로써 8개 특징점에 라벨값을 각각 지정할 수 있고 이 각각의 라벨값을 이용해서 8개의 특징점 좌표를 모두 검출할 수 있다. 하지만 물체가 roll 각도로 회전하였을 때 그림 5 오른쪽 영상의 세로로 된 직사각형 안에 표시된 점들은 라벨링 되는 순서가 바뀌므로

좌표 값이 변하게 된다. 그래서 1차 라벨링에서는 roll 방향으로 회전을 하여도 좌표 값이 변하지 않는 가로로 된 직사각형 안에 있는 점들의 좌표만 검출하였다.

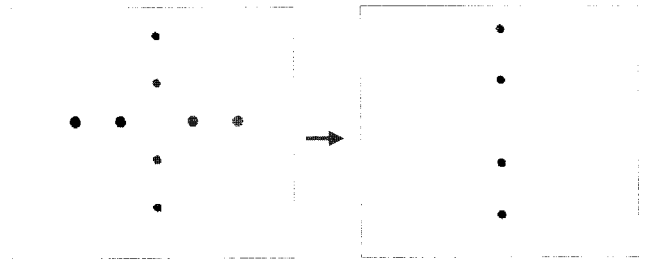


그림 6 1차 라벨링 영상의 이진화  
Fig. 6 Binarization of first labeling image

1차 라벨링에서 가로축에 존재하는 점들의 좌표를 검출하였고 2차 라벨링에서 세로축의 점들의 좌표를 검출하기 위해서 그림 6과 같이 1차 라벨링 된 영상에 이진화 알고리즘을 적용하였다. 2차 라벨링은 세로축으로 형성된 점들의 좌표만 구하기 때문에 가로축에 있는 점들은 그레이 값을 배경(background)과 같은 255를 부여해서 이미지 상에서 제거하였다.

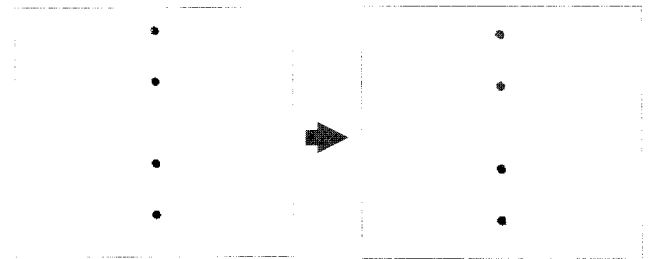


그림 7 2차 라벨링 영상  
Fig. 7 Second labeling Image

그림 7는 가로축의 점들을 제거한 뒤 2차 라벨링 알고리즘을 적용시킨 영상이다. 2차 라벨링은 이미지 상에서 아래에서 위로 실행하였고 이 점들에 라벨값을 부여함으로써 세로축의 점들의 좌표를 구할 수 있다.

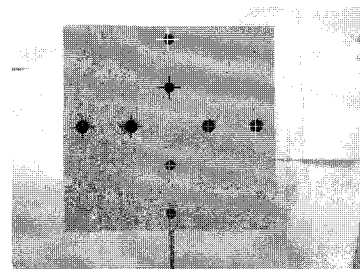


그림 8 중심좌표를 검출한 영상  
Fig. 8 Image of detecting center coordinate

그림 8은 PC카메라로부터 얻은 영상을 영상처리 알고리즘과 중심좌표 검출 알고리즘을 적용하여 얻은 이미지이다.

2.2 방향각 측정 알고리즘

2.2.1 방향각의 구성

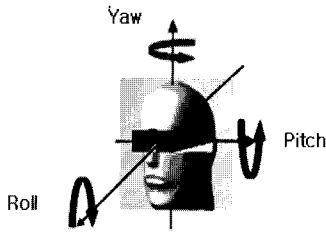


그림 9 롤, 피치, 요  
Fig. 9 Roll, Pitch, Yaw

그림 9는 방향각의 구성을 나타내고 앞의 영상처리 알고리즘을 이용하여 검출한 8개 특징점의 좌표들을 가지고 Roll, Pitch, Yaw를 측정하였다.

2.2.2 Roll Estimation Algorithm

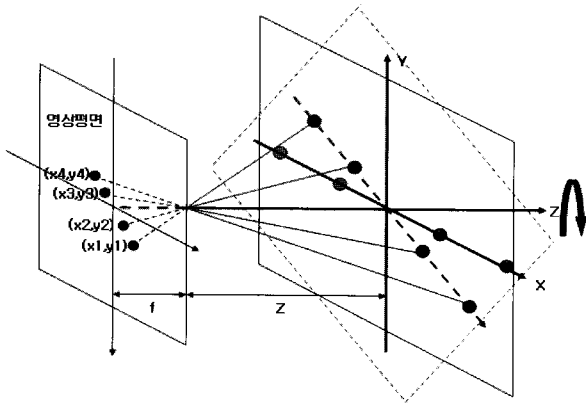


그림 10 Z축을 중심으로 회전각 측정  
Fig. 10 Roll estimation

그림 10을 보면 roll은 Z-axis의 회전각이다. 이 roll은 영상처리 알고리즘에서 얻어진 x와 y 좌표만으로 쉽게 구할 수 있다.

$$\gamma = \arctan(\Delta y / \Delta x) \tag{1}$$

$$\Delta y = y_4 - y_1$$

$$\Delta x = x_4 - x_1$$

식(1)을 적용하면 쉽게 roll을 측정할 수가 있다. 하지만 하드웨어적으로 이 점들은 정확히 하나의 일직선상에 있다고 볼 수 없다. 그러므로 미소하지만 약간의 오차가 존재할 것이고 이 오차를 줄이고자 다음의 방법을 이용하여 Z축의 회전각을 측정하였다.

그림 11에서 x축 점들의 좌표 4개를 이용해서 roll에 대한 회전각 5개식을 다음과 같이 유도할 수 있다.

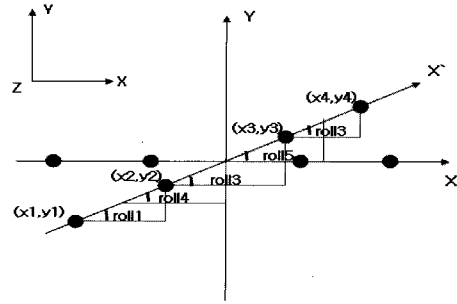


그림 11 Roll 측정 알고리즘  
Fig. 11 The algorithm for roll estimation

$$roll1 = \arctan\left(\frac{y_2 - y_1}{x_2 - x_1}\right), roll2 = \arctan\left(\frac{y_3 - y_2}{x_3 - x_2}\right), roll3 = \arctan\left(\frac{y_4 - y_3}{x_4 - x_3}\right)$$

$$roll4 = \arctan\left(\frac{((y_1 + y_2)/2 - (y_2 + y_3)/2)}{((x_1 + x_2)/2 - (x_2 + x_3)/2)}\right) \tag{2}$$

$$roll5 = \arctan\left(\frac{((y_2 + y_3)/2 - (y_3 + y_4)/2)}{((x_2 + x_3)/2 - (x_3 + x_4)/2)}\right)$$

식(2)에서 구한 다섯 개의 roll각들을 합하여 구한 roll각 개수만큼 나누어 주면 평균 roll각을 구할 수 있다.

$$roll(\gamma) = (roll1 + roll2 + roll3 + roll4 + roll5) / 5 \tag{3}$$

2.2.3 Yaw Estimation Algorithm

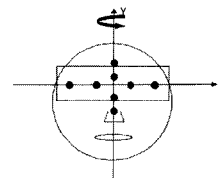


그림 12 Y축을 중심으로 회전각 측정  
Fig. 12 Yaw estimation

Yaw 측정 알고리즘은 그림 12와 같이 Y축을 중심으로 회전하는 각을 검출하는 것이다. 즉 머리가 좌우로 회전할 때를 의미한다. 이 Yaw각의 측정에는 중심사영과 cross-ratio의 불변성을 이용하였다.

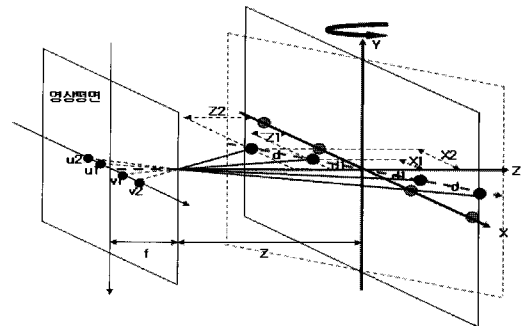


그림 13 Y축을 기준으로 회전할 때의 점들의 중심사영  
Fig. 13 The perspective projection of points while the head is rotating about Y-axis

그림 13에서 보듯이 물체의 점과 핀 홀, 영상평면위의 점은 일직선상에 있다. 위와 같은 사영을 중심 사영 (perspective projection)이라 한다. 이 중심사영을 이용하여 아래 4개의 식들을 얻을 수 있다.

$$u_2 = \frac{fX_2}{Z+Z_2} = \frac{f(d+d_1)\cos\beta}{Z+(d+d_1)\sin\beta} \quad (4)$$

$$u_1 = \frac{fX_1}{Z+Z_1} = \frac{fd_1\cos\beta}{Z+d_1\sin\beta} \quad (5)$$

$$v_1 = \frac{fX_1}{Z-Z_1} = \frac{fd_1\cos\beta}{Z-d_1\sin\beta} \quad (6)$$

$$v_2 = \frac{fX_2}{Z-Z_2} = \frac{f(d+d_1)\cos\beta}{Z-(d+d_1)\sin\beta} \quad (7)$$

위의 식 중에서 식(5)을 이용해서 다음과 같이 유도할 수 있다.

$$u_1 = \frac{fX_1}{Z+Z_1} \Rightarrow Z+Z_1 = \frac{fX_1}{u_1} \quad (8)$$

$$Z = \frac{fX_1}{u_1} - Z_1 = Z_1 \left( \frac{fX_1}{u_1 Z_1} - 1 \right)$$

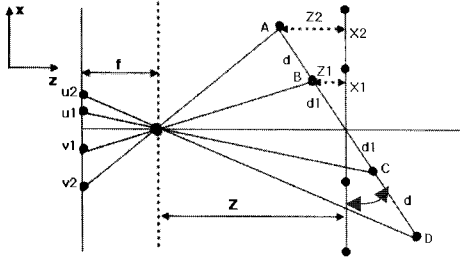


그림 14 Yaw( $\beta$ ) 계산  
Fig. 14  $\beta$  calculation

여기서 Y축을 기준으로 회전한 각은  $\beta$ 이다. 그림 14를 보면  $\frac{Z_1}{X_1} = \tan(\beta)$ 임을 알 수 있다.

그리고 식 (8)에서  $(\frac{fX_1}{u_1 Z_1} - 1) = S$ 라고 하면 이 식으로부터 Y축을 기준으로 회전하는 각( $\beta$ )을 얻을 수 있다.

$$S = \frac{f}{u_1 \tan(\beta)} - 1 \Rightarrow \beta = \text{atan}\left(\frac{f}{u_1(S+1)}\right) \quad (9)$$

식(9)에서 f는 카메라의 초점거리이므로 알 수 있다. 하지만  $u_1$ 과 S는 구해야 하는 변수이다. 이 두 변수는 X축에 존재하는 중심사영된 4개의 점들의 상대적인 거리로부터 결정할 수 있다. 그러므로 식(4)~식(7)을 이용하여 S를 아래와 같이 유도할 수 있다.

$$\frac{u_2 - u_1}{v_1 - v_2} = \frac{\frac{f(d+d_1)\cos\beta}{Z+(d+d_1)\sin\beta} - \frac{fd_1\cos\beta}{Z+d_1\sin\beta}}{\frac{fd_1\cos\beta}{Z-d_1\sin\beta} - \frac{f(d+d_1)\cos\beta}{Z-(d+d_1)\sin\beta}} \quad (10)$$

식(10)에서 우측에 있는 분자의 식을 A, 분모의 식 B라고 할 때 이를 정리하면 아래의 식(11)들을 획득할 수 있다.

$$A = \frac{fZd\cos\beta}{\{Z+(d+d_1)\sin\beta\}\{Z+d_1\sin\beta\}} \quad (11)$$

$$B = \frac{-fZd\cos\beta}{\{Z-d_1\sin\beta\}\{Z-(d+d_1)\sin\beta\}}$$

$$\frac{A}{B} = -\frac{(Z-d_1\sin\beta)\{Z-(d+d_1)\sin\beta\}}{(Z+d_1\sin\beta)\{Z+(d+d_1)\sin\beta\}}$$

식(8)에서  $Z = Z_1 S$ 임을 알 수 있으므로 위 식을 다음과 같이 정리할 수 있다.

$$\frac{A}{B} = -\frac{(S - \frac{d_1\sin\beta}{Z_1})\left\{S - \frac{(d+d_1)\sin\beta}{Z_1}\right\}}{(S + \frac{d_1\sin\beta}{Z_1})\left\{S + \frac{(d+d_1)\sin\beta}{Z_1}\right\}} \quad (12)$$

그림 14에서 보면  $\frac{Z_1}{d_1} = \sin\beta$  이고  $Z_1 = d_1 \sin\beta$ 임을 알 수 있고 이 식을 식(12)에 대입하면 아래와 같은 식이 유도된다.

$$\frac{A}{B} = -\frac{(S-1)\left\{S - \left(\frac{d}{d_1} + 1\right)\right\}}{(S+1)\left\{S + \left(\frac{d}{d_1} + 1\right)\right\}} \quad (13)$$

그림 14에서 cross-ratio invariance을 이용하여 식(14)를 유도할 수 있다.

$$\frac{(u_2 - u_1)(v_1 - v_2)}{(u_2 - v_1)(u_1 - v_2)} = \frac{d^2}{(2d_1 + d)^2} = C_{ratio} \quad (14)$$

$$\frac{d}{(2d_1 + d)} = \sqrt{C_{ratio}} \Rightarrow \frac{2d_1}{d} = \frac{1}{\sqrt{C_{ratio}}} - 1$$

여기서  $\frac{2d_1}{d} = H$ 라고 정의하자. 그러면  $\frac{d_1}{d} = \frac{H}{2}$ 라고 할 수 있고, 식(13)에 대입하면 식(15)와 같이 S에 대한 2차 방정식을 얻을 수 있다.

$$\begin{aligned} a &= \Delta u + \Delta v & (15) \\ b &= (\Delta u - \Delta v)\left(\frac{2}{H} + 2\right) \\ c &= (\Delta u + \Delta v)\left(\frac{2}{H} + 1\right) \\ aS^2 + bS + c &= 0 \end{aligned}$$

앞에서 Y축을 기준으로 회전한 각  $\beta$ 를 구하기 위해 필요한 변수 S를 구하였고, 이제 남은 변수  $u_1$ 을 결정해야 한다.

이  $u_1$ 을 구하기 위해서 다른 두 개의 cross-ratio 불변성을 이용하였다. 그림 14를 이용하여 다음의 두 식을 얻을 수 있다.

$$\frac{(u_2 - u_1)v_1}{(u_1 - v_1)u_2} = -\frac{dd_1}{2d_1(d+d_1)} = C_{ratio1} \quad (16)$$

$$\frac{(v_1 - v_2)u_1}{(u_1 - v_1)v_2} = -\frac{dd_1}{2d_1(d+d_1)} = C_{ratio2}$$

위 두 식을 보면  $C_{ratio1}$ 와  $C_{ratio2}$ 가 같음을 알 수 있다. 식(16)에서 두 번째 식을  $u_1$ 에 대하여 정리하면 아래의 식을 획득한다.

$$u_1 = \frac{C_{ratio2}(u_1 - v_1)v_2}{(v_1 - v_2)} \quad (17)$$

식(17)에서  $u_1$ 을 구하기 위해서는  $v_2$ 를 알아야 하므로 위의 식을 다음과 같이 정리할 수 있다.

여기서  $\Delta v = v_1 - v_2$ ,  $\Delta u = u_2 - u_1$ 라고 하면 식(18)을 유도할 수 있다.

$$v_2 = \frac{\{\Delta v \Delta u + \Delta v u_1 - \Delta u v_1\}}{C_{ratio2}(u_1 - v_1) - \Delta u} \quad (18)$$

식(18)에서  $v_2$ 에 관한 식을 얻었고 이 식을 식(17)에 대입하면 식(19)를 얻는다.

$$u_1 = \frac{C_{ratio2}(u_1 - v_1)\{\Delta v \Delta u + \Delta v u_1 - \Delta u v_1\}}{\Delta v\{C_{ratio2}(u_1 - v_1) - \Delta u\}} \quad (19)$$

$$u_1 = \frac{C_{ratio2}(u_1 - v_1)\Delta v \Delta u - C_{ratio2}^2(u_1 - v_1)\left\{\frac{\Delta v u_1}{C_{ratio2}} - \frac{\Delta u v_1}{C_{ratio2}}\right\}}{\Delta v\{C_{ratio2}(u_1 - v_1) - \Delta u\}}$$

식(19)에서  $u_1$ 과  $v_1$ 을 제외한 나머지는 이미지 상에서 구할 수 있는 상대적인 거리이다.  $u_1$ 과  $v_1$ 을 제거하기 위해서 식(16)의 두 식을 이용하여 다음의 식을 획득한다.

$$\frac{\Delta u v_1}{C_{ratio2}} = (u_1 - v_1)u_2, \quad \frac{\Delta v u_1}{C_{ratio2}} = (u_1 - v_1)v_2 \quad (20)$$

식(20)을 식(19)에 대입하면 영상에서 얻을 수 있는 점들의 상대적인 거리만을 이용해서  $u_1$ 을 결정할 수 있다.

$$u_1 = \frac{(u_1 - v_1)\Delta v \Delta u C_{ratio2} - (u_1 - v_1)^2(u_2 - v_2)C_{ratio2}^2}{\Delta v(u_1 - v_1)C_{ratio2} - \Delta v \Delta u} \quad (21)$$

앞에서 구한 S와  $u_1$ 을 식(9)에 대입하면 Y축에 기준으로 회전한 각  $\beta$ 를 얻을 수 있다.

### 2.2.4 Pitch Estimation Algorithm

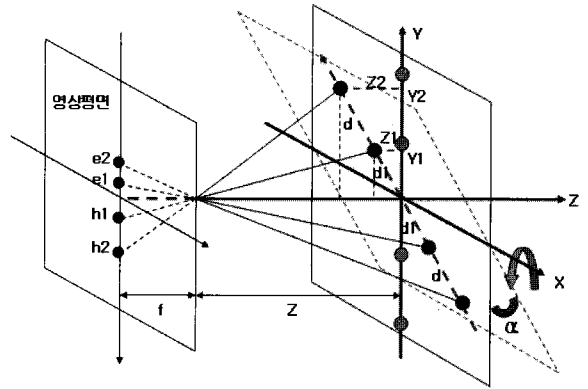


그림 15 X축을 중심으로 회전각 측정

Fig. 15 Pitch estimation

Pitch 측정 알고리즘은 그림 15와 같이 X축을 중심으로 회전하는 각을 검출하는 것이다. 즉 머리가 위 아래로 회전할 때를 의미한다. 이 Pitch각의 측정도 Yaw각 측정과 마찬가지로 중심사영과 cross-ratio의 불변성을 이용하였다. 그림 15에서 보듯이 물체의 점과 핀 홀, 영상평면위의 점은 일직선상에 있고. 위와 같은 중심 사영(perspective projection)을 이용하여 아래의 4개의 식들을 얻을 수 있다.

$$e_2 = \frac{f Y_2}{Z + Z_2} = \frac{f(d+d_1)\cos\alpha}{Z + (d+d_1)\sin\alpha} \quad (22)$$

$$e_1 = \frac{f Y_1}{Z + Z_1} = \frac{f d_1 \cos\alpha}{Z + d_1 \sin\alpha} \quad (23)$$

$$h_1 = \frac{f Y_1}{Z - Z_1} = \frac{f d_1 \cos\alpha}{Z - d_1 \sin\alpha} \quad (24)$$

$$h_2 = \frac{f Y_2}{Z - Z_2} = \frac{f(d+d_1)\cos\alpha}{Z - (d+d_1)\sin\alpha} \quad (25)$$

위의 식 중에서 식(23)을 정리하면 다음의 식을 획득한다.

$$\alpha = \text{atan}\left(\frac{f}{e_1(P+1)}\right) \quad (26)$$

식(26)에서  $f$ 는 카메라의 초점거리이고  $e_1$ 과  $P$ 는 X축을 기준으로 회전할 때 구해야 하는 변수이다. 이 두 변수는 영상평면의 y축에 존재하는 중심사영된 4개의 점들의 상대적인 거리로부터 결정할 수 있다. 그리고 yaw각과 마찬가지로 pitch각을 구하기 위한 두 개의 변수  $P$ 와  $e_1$ 에 관한 식을 다음과 같이 얻을 수 있다.

$$a = \Delta e + \Delta h \quad (27)$$

$$b = (\Delta e - \Delta h)\left(\frac{2}{T} + 2\right)$$

$$c = (\Delta e + \Delta h)\left(\frac{2}{T} + 1\right)$$

$$aP^2 + bP + c = 0$$

$$e_1 = \frac{(e_1 - h_1)\Delta h \Delta e C_{ratio2} - (e_1 - h_1)^2 (e_2 - h_2) C_{ratio2}^2}{\Delta h (e_1 - h_1) C_{ratio2} - \Delta h \Delta e} \quad (28)$$

여기서 앞에서 구한 P와  $e_1$ 을 식(26)에 대입하면 X축을 기준으로 회전한 각을 구할 수 있다.

### 2.3 실험 결과

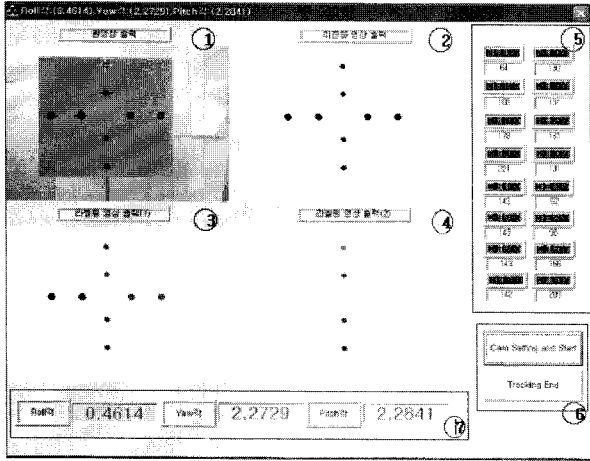


그림 16 회전각 측정 프로그램의 구성  
Fig. 16 The composition of program for rotation angle estimation

그림 16은 대상물체의 회전각을 측정하기 위한 프로그램의 구성도이다. 그림 16에서 창1번은 PC카메라에서 받은 영상을 영상처리알고리즘을 이용하여 점들의 중심좌표를 검출한 이미지를 나타낸다. 창2는 영상처리알고리즘의 한 부분인 이진화를 적용시켜 출력한 이미지이다. 창3은 1번에서 4번까지의 좌표를 검출하기 위해 좌우로 라벨링 된 이미지 출력하고 있다. 창4는 5번에서 8번까지의 좌표를 검출하기 위해 이미 검출된 좌표의 점들은 제거하고 위아래로 라벨링 된 이미지를 출력하고 있다. 창5는 영상에서 점들의 x와 y 좌표값들을 나타낸다. 창6은 영상처리 하기 전 PC카메라를 설정 할 수 있는 버튼과 종료 버튼으로 구성되어 있다. 창7은 회전각측정 알고리즘을 이용하여 검출한 roll, pitch, yaw를 나타낸다.

표 1 Roll 측정 결과

Table 1 Roll estimation results

Roll: 5.1771 실제 회전각: 5°	Roll: 10.2887 실제 회전각: 10°	Roll: 15.4910 실제 회전각: 15°	Roll: 20.2165 실제 회전각: 20°
Roll: -5.4045 실제 회전각: -5°	Roll: -10.7523 실제 회전각: -10°	Roll: -15.1060 실제 회전각: -15°	Roll: -20.5941 실제 회전각: -20°

표 1은 roll 측정 실험결과이다. 위의 표를 보듯이 실제 회전각과 측정값이 약 0.1°에서 0.8°가량의 오차가 존재한다. 이 오차는 실제 회전각을 볼 때 소수점까지 파악하지 않았고 측정도구인 PC카메라도 정확하게 수평의 위치에 있다고 볼 수 없기 때문이다. 그리고 pan-tilt를 제어하는데 있어서 yaw각과 pitch각을 제외하고 roll각에 대한 회전은 영향을 미치지 않는다.

표 2 Yaw 측정 결과

Table 2 Yaw estimation results

Yaw: 4.1934 실제 회전각: 5°	Yaw: 11.1204 실제 회전각: 10°	Yaw: 14.3861 실제 회전각: 15°	Yaw: 20.1560 실제 회전각: 20°
Yaw: -4.2314 실제 회전각: -5°	Yaw: -11.6531 실제 회전각: -10°	Yaw: -15.1515 실제 회전각: -15°	Yaw: -19.0363 실제 회전각: -20°

표 2는 실시간으로 좌우 회전을 했을 때의 실험 결과이다. 실제 회전각과 측정값이 약 1°에서 2°가량의 오차가 나는 것을 볼 수 있다. 이 오차는 하드웨어적으로 점의 위치나 카메라의 위치가 정확하게 정교하지 않아서 발생할 수 있다. 이미지 사이즈는 320×240을 사용하였는데 픽셀수가 더 정교한 이미지를 사용한다면 오차는 더욱 줄어들 것이다. 그리고 측정되는 거리가 멀어질수록 특징점 사이의 거리에 존재하는 픽셀의 수는 점점 줄어들어서 오차의 크기가 더욱 커지게 된다. 이 오차 역시 이미지 사이즈가 큰 영상을 사용한다면 줄일 수 있다.

표 3 Pitch 측정 결과

Table 3 Pitch estimation results

Pitch: 4.5202 실제 회전각: 5°	Pitch: 10.4411 실제 회전각: 10°	Pitch: 14.4324 실제 회전각: 15°	Pitch: 19.2269 실제 회전각: 20°
Pitch: -4.0235 실제 회전각: -5°	Pitch: -10.6484 실제 회전각: -10°	Pitch: -14.6840 실제 회전각: -15°	Pitch: -21.7818 실제 회전각: -20°

표 3은 위 아래로 회전했을 때의 실험 결과이다. Pitch 측정 원리는 Yaw의 측정원리와 비슷하다. 표에서 보듯이 실제 회전각과 측정값을 비교했을 때 오차는 약 1°~2°가량이나는 것을 볼 수 있고 이 측정 역시 카메라의 위치나 대상 물체의 위치에 따라 약간의 오차가 존재할 수 있다.

표 4 Yaw와 Pitch 측정 결과

Table 4 Yaw & Pitch estimation results

Yaw Pitch	-15	-10	-5	0	5	10	15
-15							
측정값Yaw	-15.1515	-11.4322	-5.6828	0.0000	5.5108	10.9585	14.6840
측정값Pitch	-14.9547	-14.9547	-14.9547	-14.6840	-14.6840	-16.5430	-14.6840
-10							
측정값Yaw	-15.7384	-11.6531	-5.7931	0.0000	5.7221	9.4437	16.8469
측정값Pitch	-9.6254	-11.3722	-11.3722	-11.3722	-11.3722	-11.3722	-11.3722
-5							
측정값Yaw	-10.3713	-10.2280	-6.0270	0.0000	5.9503	10.0102	15.8288
측정값Pitch	-5.9503	-4.0235	-5.9503	-5.9503	-6.0713	-5.9503	-6.0713
0							
측정값Yaw	-15.1515	-10.6631	-6.2805	0.0000	4.1934	10.4269	14.3861
측정값Pitch	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5							
측정값Yaw	-15.8259	-11.1365	-4.3235	0.0000	4.2838	10.8793	14.9942
측정값Pitch	-6.7033	-6.7033	-4.4196	-4.4196	-4.4196	-4.4196	-4.4196
10							
측정값Yaw	-14.0974	-9.2371	-4.5202	0.0000	4.4763	11.1204	15.6547
측정값Pitch	-11.3890	-9.2371	-9.2371	-11.6531	-9.2371	-9.2371	-9.2371
15							
측정값Yaw	-14.4324	-9.4554	-4.7356	0.0000	4.6880	9.2687	14.0068
측정값Pitch	-14.4324	-14.4324	-14.4324	-14.4324	-14.4324	-14.4324	-14.4324

표 4는 Yaw( $\beta$ )와 Pitch( $\alpha$ )를 함께 회전했을 때의 실험결과이다. 위의 표를 보듯이 각각 회전했을 때의 오차의 범위가 Yaw와 Pitch를 함께 회전했을 때의 오차의 범위는 차이가 나지 않는 것을 볼 수 있다. 이 논문에서의 목적은 pan-tilt를 제어하는 것이기 때문에 roll각은 영향을 미치지 않으므로 표 4의 측정치에서 제외하였다.

표 5 Yaw와 Pitch로 회전할 때의 오차거리

Table 5 When rotating yaw & pitch, the distance error

Yaw( $\beta$ ) Pitch( $\alpha$ )	-15	-10	-5	0	5	10	15
-15	0.1581	1.4329	0.6843	0.3160	0.6006	1.8167	0.4469
-10	0.8280	2.1484	1.5849	1.3722	1.5506	1.4807	2.3009
-5	1.6684	1.0028	1.3992	0.9503	1.4320	0.9504	1.3545
0	0.1515	0.6631	1.2805	0	0.8066	0.4269	0.6139
5	1.8930	2.0476	0.8914	0.5804	0.9218	1.0536	0.5804
10	1.6565	1.0789	0.9012	1.6531	0.9251	1.3555	1.0053
15	0.8027	0.7866	0.6262	0.5676	0.6477	0.9257	1.1439

표 5는 yaw와 pitch로 회전할 때의 오차거리이다. 즉 yaw와 pitch로 함께 회전했을 때에 실제값과 측정값의 차이를 나타내고 있다. 표에서 보듯이 오차거리는 약 2°이하임을 볼 수 있다.

$$\text{오차거리} = \sqrt{(\beta_{\text{실제값}} - \beta_{\text{측정값}})^2 + (\alpha_{\text{실제값}} - \alpha_{\text{측정값}})^2} \quad (29)$$

### 3. 결론

본 논문은 원격제어 비전 시스템을 구축하기 위한 연구로서 HMD를 착용한 사용자의 머리 회전각을 측정하는 알고리즘을 제안하였다. PC카메라로부터 얻은 회전각을 네트워크를 통해 이동로봇에 장착된 pan-tilt에 전송하여 제어하고자 하는 목적으로 영상처리 알고리즘으로부터 얻는 8개의 특징점의 좌표를 이용하여 실시간으로 회전각을 측정하였다. 실험결과 본 논문에서 제안한 영상처리 알고리즘과 회전각 측정 알고리즘의 유용성이 입증되었다. 회전각의 오차는 0~2°정도이고 이 오차는 카메라의 위치와 대상물체의 위치가 정교하게 설정된 게 아니기 때문에 발생할 수 있다. 그리고 PC카메라로부터 받은 영상을 처리하므로 조명, 주위 환경 등 외란의 영향을 많이 받았다. 향후 과제로는 조명 등의 외란에 극복할 수 있는 영상처리 알고리즘이 추가되어야 할 것이다.

### 감사의 글

본 연구는 2006년 울산 대학교 교내 연구비, 산업자원부 지정 울산대 RIC(NARC) 및 교육인적자원부 지정 울산대 BK21대학원(EVERDEC) 지원에 의한 것입니다. 지원에 감사드립니다.

### 참고 문헌

- [1] 손재범, 정완균, 염영일, "원격제어 시스템", 제어·자동화시스템 학회지, pp.42~60, 1996.
- [2] Ren C.Luo, Tse Min Chen and Chih-Chen Yih, "Intelligent Autonomous Mobile Robot Control Through the Internet." IEEE, Vol. 1, pp. 6-11, December 2000.
- [3] 심교만, "시간지연 안정성을 고려한 이동로봇의 무선 원격조정", 울산대학교 석사학위 논문, pp.1-6, 2002.12.
- [4] 정기수, "이동로봇 원격조작을 위한 인터넷 기반 제어 시스템에 관한 연구", 대한전기학회 하계학술대회 논문집(D), pp.1983-1986, 2006.07.
- [5] Olivier Faugeras, "Three-Dimensional Computer Vision A Geometric Viewpoint", The MIT press, pp.7-30, 1993.
- [6] 저/조강현·유범재 공역, "3차원 비전", 대영사, pp.19-38, 2000.
- [7] 저/강동중, 하종은, "Visual C++을 이용한 디지털 영상 처리", 사이텍미디어, pp.247-270, 2003.
- [8] Olivier Faugeras and Quang-Tuan luong, "The



Geometry of Multiple Images”, The MIT press, pp1-14, 2001.

- [9] C. Kambhamettu Y. Zhang, "Robust 3d head tracking under partial occlusion. Automated Face and Gesture Recognition", IEEE Computer Society Washington. DC. USA, pp176, 2000.

## 저 자 소 개



### 노 영 식 (盧 瑩 植)

1959년 1월 1일생. 1981년 연세대학교 전기공학과 졸업. 1987년 동 대학원(공학 박사). 1987년~1991년 삼성전자생산기술연구소 선임연구원. 1991년~현재 울산대학교 전기전자정보시스템공학부 교수.

Tel : 052) 259-2190

E-mail : ysro@mail.ulsan.ac.kr



### 윤 승 준 (尹 升 俊)

1980년 10월 3일생. 2007년 울산대학교 전기전자정보시스템공학부 졸업. 2007년~현재 동 대학원 석사과정.

Tel : 052) 259-1642

E-mail : ysj7447@mail.ulsan.ac.kr



### 강 희 준 (姜 熙 峻)

1961년 10월 1일생. 1985년 서울대학교 기계공학과 졸업. 1991년 Texas Austin 대학(공학박사). 1992년~현재 울산대학교 전기전자정보시스템공학부 교수

Tel : 052) 259-2207

E-mail : hjkang@mail.ulsan.ac.kr



### 서 영 수 (徐 榮 秀)

1967년 10월 6일생. 1990년 서울대 공대 제어계측과 졸업. 1992년 동 대학원 제어계측과 졸업. 1997년 동경대학교 계수공학과 졸업(공학박사). 2000~현재 울산대학교 전기전자정보시스템공학부 부교수

Tel : 052) 259-2196

E-mail : suh@ieee.org