석사학위논문

3D 실린더 모델기반 머리자세추정을 이용한 텔레프레즌스 화상회의시스템의 구현

2014年

한성대학교 대학원
정보통신공학과
정보통신공학전공
진 용 규

석 사 학 위 논 문

지도교수 조혜경

3D 실린더 모델기반 머리자세추정을 이용한 텔레프레즌스 화상회의시스템의 구현

A Cost-Effective Telepresence Video Communication Robot System
Delivering Speaker's Head Motion
by Employing 3D Cylinder Head Model

2 0 1 3 년 12 월 일

한성대학교 대학원

정 보 통 신 공 학 과

정 보 통 신 공 학 전 공

진 용 규

석 사 학 위 논 문
지도교수 조혜경

3D 실린더 모델기반 머리자세추정을 이용한 텔레프레즌스 화상회의시스템의 구현

A Cost-Effective Telepresence Video Communication Robot System
Delivering Speaker's Head Motion
by Employing 3D Cylinder Head Model

위 논문을 공학 석사학위 논문으로 제출함

2 0 1 3 년 12 월 일

한성대학교 대학원
정 보 통 신 공 학 과
정 보 통 신 공 학 전 공
진 용 규

국 문 초 록

3D 실린더 모델기반 머리자세추정을 이용한 텔레프레즌스 화상회의 시스템의 구현

한성대학교 대학원
정보통신공학과
정보통신공학전공
진 용 규

본 논문에서는 대화 상대의 머리 움직임을 모사하는 디스플레이장치를 가진 화상회의용 텔레프레즌스 로봇 시스템을 제안한다. 대화 상대의 머리자세는 동의나 부정을 나타내는 의사표시나 시선 방향을 통한 관심 표현의 단서를 제공하는 비언어적 소셜 신호 중 하나로, 이를 통해 원격지에 있는 사용자 간 영상기반 상호작용의 몰입감과 실존감을 증대시킬 수 있음이 알려져 있다. 이러한 실감형 텔레프레즌스 로봇 시스템을 구현하기 위해서는 영상 시퀀스로부터 화자의 머리 자세를 효과적으로 추정할 수 있는 알고리즘, 자연스러운 실재감을 유지하기 위한 매개체들 간의 실시간 통신, 합리적인 비용의 활용성이 높은 매개 로봇 장치가 필요하다.

본 논문에서는 먼저, 머리자세 추정 방법에 대한 선행 연구들을 분석하여 그 중에서 추정 정확도가 높고 실시간 환경에 적합하다고 알려져 있는 방법을 기반으로, 머리의 3D 모델과 특징점 추적을 결합한 효과적인 머리자세 추적 방법을 제안한다. 입력 영상으로부터 얼굴검출 시 피부색을 이용하는 방법을 제안함으로써, 얼굴 검출 모듈이 표정이나 각도 변화에 강인한 특성을

가지게 한다. 특히 온라인 참조 프레임 등록을 통하여 변화하는 얼굴 자세와 주변 환경의 변화에 적용할 수 있도록 한다. 매개 장치는 스마트기기와 저가의 범용 로봇 키트를 사용하여 누구나 간편하게 상호작용 효과가 증대된 텔레프레즌스 시스템을 구현할 수 있도록 한다.

제안된 시스템을 통해 간단한 구성으로 원격 환경 사이의 텔레프레즌스 화상회의 시스템을 구현할 수 있음에 따라, 고가의 제품들로 개인화가 힘들었던 텔레프레즌스 시스템의 보급이 확대되어, 일반 사용자들에게 교감형 텔레프레즌스 도구로서 활용될 수 있을 것으로 기대한다.

【주요어】 텔레프레즌스 로봇, 머리자세추정, 화상회의, Telepresence Robot, Head Pose Estimation, Video Conferencing, Telepresence System, Human Robot Interaction

목 차

Ⅰ. 서 론 ····· 1

1.1 연구의 배경 및 목표 ····· 1
1.2 연구의 방법 ····· 3
1.3 논문의 구성 ····· 4

Ⅱ. 이론적 배경 ····· 5

2.1 텔레프레즌스 로봇 ····· 5
2.2 모델 기반 3차원 머리 자세 추정 ····· 7
2.2.1 머리자세 ····· 7
2.2.2 머리자세 추정 방법 ····· 8
2.2.3 Harris Corner Detector ····· 11
2.2.4 Lucas-Kanade Tracking ····· 14

Ⅲ. 3차원 얼굴 추적을 이용한 머리 자세 추정 ····· 16

3.1 제안 시스템의 개요 ····· 16
3.2 레이블링을 이용한 얼굴 검출 방법 ····· 18
3.3 온라인 참조 프레임 등록 ····· 21
3.4 POSIT 알고리즘을 이용한 3차원 모델 ····· 22
3.5 요약 ····· 24

Ⅳ. 텔레프레즌스 화상회의 시스템 설계 ····· 25

4.1 통합 시스템의 구성 ····· 25
4.2 소프트웨어 아키텍처 ····· 27
1) 영상 및 음성 통신부 ····· 27
2) 머리자세 추정 및 로봇 제어부 ····· 32

4.3 구현 결과 ····· 33
1) 소프트웨어구현 ····· 33
2) 하드웨어구현 ····· 33

Ⅴ. 실험 결과 ····· 35

5.1 실험 환경 ····· 35
5.2 실험 결과 ····· 36

Ⅵ. 결 론 ····· 41

참 고 문 헌 ····· 42

ABSTRACT ····· 45

그 림 목 차

〈그림 1 〉 다양한 텔레프레즌스 시스템 ····· 1
〈그림 2 〉 텔레프레즌스 로봇 PRoP 시스템 ····· 5
〈그림 3 〉 팔의 제스처를 접목시킨 텔레프레즌스 로봇 Me-bot ····· 5
〈그림 4 〉 머리자세와 시선의 관계 ····· 7
〈그림 5 〉 머리 자세 각도 ····· 8
〈그림 6 〉 영상에서 나타난 코너 포인트 ····· 11
〈그림 7 〉 각 영역의 종류에 대한 화소값의 변화 ····· 12
〈그림 8 〉 3D 회전체에서의 각 축 방향으로 표시된 optical flow ····· 14
〈그림 9 〉 Block Diagram of Head Pose Estimation ····· 16
〈그림 10〉 얼굴 검출 과정 ····· 18
〈그림 11〉 YCrCb 색공간 ····· 19
〈그림 12〉 모델 포인트 에 대해 의 정사영 투영 ····· 22
〈그림 13〉 시스템 구성도 ····· 25
〈그림 14〉 소프트웨어 아키텍처 ····· 27
〈그림 15〉 소스요소의 구조 ····· 28
〈그림 16〉 필터요소의 구조 ····· 29
〈그림 17〉 디먹서요소의 구조 ····· 29
〈그림 18〉 싱크요소의 구조 ····· 29
〈그림 19〉 영상 송신 블록 ····· 30
〈그림 20〉 영상 수신블록 ····· 31
〈그림 21〉 로봇 명령 패킷 ····· 32
〈그림 22〉 로봇에 스마트기기가 장착된 모습 ····· 34
〈그림 23〉 얼굴 추적을 위한 BU dataset ····· 35
〈그림 24〉 얼굴검출 비교 ····· 36
〈그림 25〉 프레임별 특징점 검출 결과 ····· 37
〈그림 26〉 머리자세 추정 후 모델 시뮬레이션 ····· 38
〈그림 27〉 머리자세추적 성능: (a) Roll, (b) Pitch, (c) Yaw. ····· 39

I. 서 론

1.1 연구의 배경 및 목표

텔레프레즌스(Telepresence)이란 용어는 원격(tele)과 실재감(presence)의 합성어로 [1]에 의해 효시 되었으며, 텔레프레즌스 사용자들이 신호나 자극을 제공하는 매개체를 통해서 물리적으로 떨어진 환경을 실재하는 것처럼 느끼는 것이라 정의할 수 있다. 텔레프레즌스 기술이 활용되는 분야는 위험작업 활용분야, 인간 작업능력증대 활용분야, 장애인 및 재활 보조기구, 게임 보조기구, 가상현실, 체험학습 등 각종 산업 현장 및 훈련 교육에 사용되고 있으며 많은 영역으로 확장되고 있다.

텔레프레즌스 기술이 성공적으로 실현되기 위해서 중요시되는 요구기술은 텔레프레즌스 시스템의 활용 형태에 따라 다르다. 원격수술로봇, 원격운전 등 원격조작 형태의 응용에서는 실제로는 원격조종 로봇을 구동 하지만, 작업자가 마치 원격지에서 자신이 환경을 직접 조작하는 것과 같이 느낄 수 있도록 현장의 정보가 조작자의 오감을 통해 효과적으로 전달되어야 하며, 조작자 의도에 따라 제어되는 이동 장치와 목적에 적합한 표현 장치가 포함되어야 한다. 그리고 화상회의, 체험학습 도구 등의 원격 환경 경험자들 간의 실감형 상호작용이 중요시되는 형태의 텔레프레즌스 시스템에서는 매개체가 되는 장치를 인격체로 인지하고 자연스러운 의사소통을 할 수 있어야 한다[2].

텔레프레즌스의 다양한 응용 시스템 중 화상회의 중심의 텔레프레즌스 로봇 시스템은 출장 비용와 시간을 줄일 수 있는 미래 유망 아이템으로 주목을 받으면서 이미 상품화가 진행되고 있고, 다음 〈그림 1〉 처럼 세부적인 용도와 사양도 다양하다. Scitos나 RP-VITA 같은 원격 진단용 고가 모델이 시판중인 한편, Botiful 같이 미니멀한 이동 기능만 갖춘 스마트폰 거치대 형태의 제품도 소개되고 있다[3].

텔레프레즌스 화상회의 서비스는 실감형 영상을 포함하여 다양한 응용서비스가 결합된 회의 서비스를 의미한다. 기존의 영상회의 기술보다 영상 및 음

〈그림 1〉 다양한 텔레프레즌스 시스템

성 스트림의 코딩, 실감형 미디어 전송, 영상 디스플레이, 시선 처리, 음성 출력 및 조명 기술 등을 추가적으로 포함하는 기술을 총칭한다. 국외 경우, 시스코, 폴리콤, HP, Digital Video Enterprise, 비됴, 화웨이 등 국외 업체들은 이미 많은 상용화 제품을 출시한 상태이다. 반면, 국내 경우는 유프리즘, 해든브리지, 새하컴즈, 우암 등 자체적으로 개발한 영상회의 솔루션을 상용화하고 있으나, 실감형 텔레프레즌스 제품은 거의 없는 실정이다[4]. 또한 대부분의 텔레프레즌스 제품은 고가 모델이 시판중이여서 대중적으로 널리 쓰이기 어려운 실정이다.

본 논문에서는 화상회의와 같은 실감형 상호작용의 형태로 쓰일 수 있는 텔레프레즌스 화상회의 시스템 구현을 목표로, 필요한 요구사항을 검토하고 이에 대한 방안을 제시한다. 실감형 상호작용 형태의 텔레프레즌스 로봇 시스템을 구현하기 위해서는 감성 교감을 위한 비언어적인 정보의 이용, 자연스러운 실재감을 유지하기 위한 매개체들 간의 실시간 통신, 합리적인 비용의 활용성이 높은 매개 장치가 요구된다.

1.2 연구의 방법

본 논문에서는 상기의 요구사항을 만족시키기 위하여, 머리자세와 개인용 모바일 기기와 합리적인 비용의 범용 로봇 키트를 이용하여 텔레프레즌스 화상회의 시스템을 구현하고 실시간 환경에서의 활용 적합성을 평가한다.

먼저, 단안 카메라 기반의 머리 자세 추정 방법 중에서 추정 정확도가 높고 실시간 환경에 적합하다고 알려져 있는 3D 모델기반 추적방법을 선택하여 제안하였다. 구현된 머리자세 추정 프로그램의 추정 성능을 평가하기 위해 Boston University의 정형화된 dataset을 이용하여 머리자세의 Roll, Pitch, Yaw에 대하여 추정 오차를 측정하였다. 다음으로 원격간의 자연스러운 실재감을 위한 영상전달과 로봇 동작이 실시간으로 가능한지 무선 네트워크 환경에서 시스템 적합성을 평가한다. 본 논문에서는 시스템 구현을 위해 다음과 같이 세 가지 방안을 제안한다.

첫 번째, 실시간 환경에 적합한 머리자세 추정 방법을 제안한다. 사람들은 다른 사람과 의사소통을 할 때, 언어적인 수단과 비언어적인 수단을 사용한다. 본 연구에서는 비언어적인 수단 중에 많이 사용되는 머리자세를 텔레프레즌스 시스템에 적용시켜서 상호작용 효과를 증대시킬 것이다. 이를 위해 단안 카메라와 컴퓨터 비전기술을 이용한 머리 자세를 추정하는 방법들에 대해 알아보고, 그중 실시간 환경에서 적합하다고 연구되고 있는 3D 모델기반 추적 방법을 이용한 머리자세 추정 방법을 제안한다.

두 번째, 감성교감용 매개 장치를 제안한다. 아직까지 텔레프레즌스 제품들의 가격은 고가여서 많은 사용자들이 접하기 어렵다. 이에 널리 보급되어 있는 개인용 스마트 기기를 매개 장치로 활용하여, 스마트 기기 내에 장착되어 있는 장치(카메라, 디스플레이, 마이크, 스피커)를 사용해 영상표시와 로봇제어 기능이 있는 감성교감 매개 장치를 제안한다.

마지막으로, 화상회의용 텔레프레즌스 로봇 시스템을 제안한다. 실시간으로 동영상 신호와 함께 스마트기기와의 블루투스 통신을 활용해서 원격 로봇의 머리 자세 명령을 내릴 수 있는 로봇제어 기능이 포함된 시스템을 제안한다.

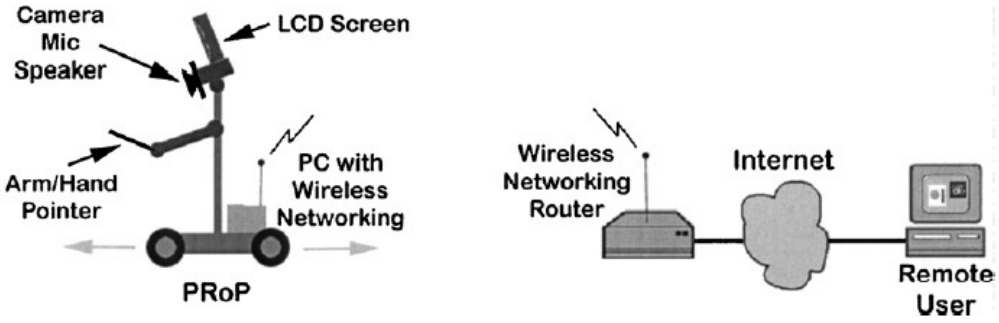
1.3 논문의 구성

본 논문의 2장에서는 선행연구들을 통해 텔레프레즌스 시스템과 소셜신호의 연관성, 얼굴자세 추정 방법들을 알아보고 적합한 얼굴자세 추정 방법을 제시한다. 3장에서는 3차원 얼굴 추적을 이용한 머리자세 추정 방법을 제안한다. 4장에서는 제안한 방법을 이용해서 전체 시스템을 설계한다. 5 장에서는 얼굴자세 dataset을 이용해 추정오차와 속도를 측정하여 구현된 시스템의 성능을 평가한다. 그리고 마지막 5장에서 결론을 정리한다.

Ⅱ. 이론적 배경

2.1 텔레프레즌스 로봇

텔레프레즌스의 사회적 측면을 탐구하기 위한 텔레프레즌스 로봇은 1990년대 중반부터 개발 되었다[5]. 다음의 〈그림 2〉처럼 물리적으로 떨어진 공간에서 인간의 언어 및 비언어적 의사소통을 위해 마이크, 스피커, 카메라, 바퀴, 무선네트워크 장치가 포함된 로봇을 구성하였다.



〈그림 2〉 텔레프레즌스 로봇 PRoP 시스템

텔레프레즌스 로봇은 다른 분야의 기술들이 결합되면서, 쇼핑몰이나 지하철과 같은 공공장소에서 서비스 로봇형태로의 활용[6], 노인 케어 커뮤니케이션[7], 의사소통과 치료 목적을 위한 로봇 테디베어[8] 등 다양한 형태로 발전되어 왔다. [9]에 연구에서는 아래의 〈그림 3〉 처럼 텔레프레즌스 시스템에 비언어적인 소셜 신호(social signal) 중 팔의 제스처를 접목시켰다.

〈그림 3〉 팔의 제스처를 접목시킨 텔레프레즌스 로봇 Me-bot

실험결과 제스처를 적용하지 않은 시스템 보다 상호작용면에서 심리적인 몰입도(psychological involvement), 참여도(engagement), 협력(cooperation), 즐거움(enjoyment) 등을 증대시키는 것으로 보고된 바 있다. 그러나 자유도 높은 팔을 구현하기 위해서는 상당한 비용이 추가 될뿐더러 조작자가 원격 로봇의 팔을 별도로 조정해야 한다는 부담을 수반한다.

사람의 몰입도를 나타내는 소셜 신호에는 시선(gaze), 얼굴 자세(head pose), 몸짓(body posture), 제스처(gestures)등이 있다. 그중에서 몰입도를 표현하는데 가장 좋은 신호는 시선이다. 하지만 시선 추적을 위해서는 머리에 장착하는 형태의 시선 추적기(eye-gaze tracker)를 필요로 한다. 시선 추적기를 통해서 빠른 속도로 정확한 얼굴 각도와 시선 추정 가능 하지만 불편하고 자연스러운 동작을 취하기가 어렵다.

2.2 모델 기반 3차원 머리 자세 추정

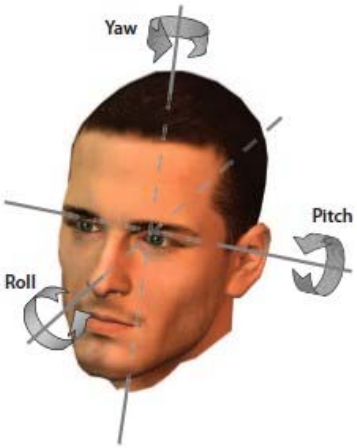
2.2.1 머리자세

머리자세는 비언어적인 의사소통 수단으로 사람들은 다른 사람 머리의 움직임과 방향을 통해 다른 사람의 의도를 파악할 수 있다. 사람이 특정 방향으로 고개를 돌리는 경우 그쪽 방향에 관심이 가는 사물이 있다는 것을 알 수 있다. 과장된 머리의 움직임은 손가락으로 가리키는 것과 같은 의미이고, 특정위치의 사람을 지시할 때도 쓰인다. 사람들은 개인의 다양한 감정을 전달하기 위해서도 머리 자세를 사용한다. 그들이 말한 것에 대한 동의를 나타내기 위해 머리를 끄덕이거나, 반대, 혼란, 배려 등의 의사를 전달하기위해 대화에서 제스처의 형태로 머리를 움직인다. 또한 머리 자세는 사람의 시선 방향을 판단하는데 큰 영향을 준다. 다음 〈그림 4〉를 예로 보면 두 그림에서 눈은 동일하지만 머리자세에 의해서 사람의 시선은 다른 방향을 향해 있다.

〈그림 4〉 머리자세와 시선의 관계

2.2.2 머리자세 추정 방법

머리자세의 추정은 3D 공간에서의 머리의 위치이동과 3축에 대한 머리 회전 각도를 추정하는 것을 의미한다. 다음 〈그림 5〉는 사람의 머리를 강체로 모델링 할 수 있다고 가정 했을 때 머리 회전방향의 정의를 보여준다.



〈그림 5〉 머리 자세 각도

머리자세를 추정하는 방법은 다양하게 연구 되어왔다. 이상적인 머리 자세 추정은 카메라 왜곡, 조명, 얼굴 표정, 다른 물체로 인한 폐쇄 등의 다양한 이미지 변화 요인에 대하여 불변성을 보여 주어야한다. 머리 자세를 추정하는 방법은 각각의 장점, 단점을 가지고 있으므로, 적용하려는 시스템의 환경과 목적에 따라 추정 방법을 선택하여야 한다.

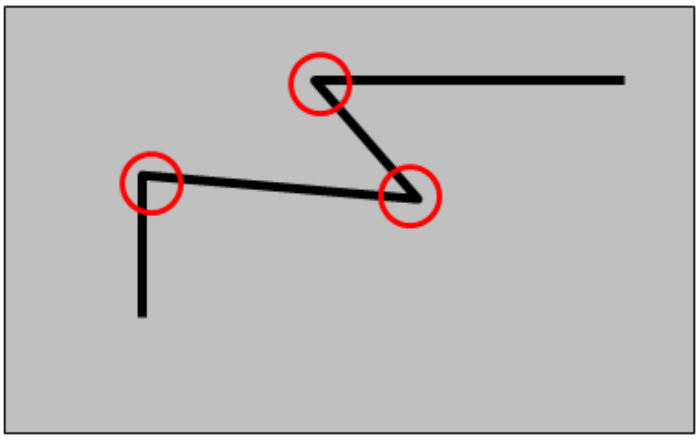
머리자세를 추정하는 방법에는 크게 외형 템플릿 방법, 기하하적 방법, 트래킹 기반 방법 그리고 얼굴 모델 기반 방법 등이 있다. 첫 번째 방법인 외형 템플릿 방법은 이미지를 기반으로 얼굴 자세를 판단하는 방법이다. 현재의 프레임 이미지 템플릿의 집합에 이미지를 비교하여 일치된 이미지에 자세를 부여한다. 외형 템플릿 방법의 장점은 시스템의 조건 변화에 따라서 적응할 수 있도록 언제든지 더 큰 집합의 템플릿으로 확장될 수 있다는 점이다. 그리고 낮은 해상도와 높은 해상도 이미지에서 모두 적합한 방법이다. 하지만 외형 템플릿 방법은 많은 단점을 가지고 있다. 연속적인 모든 자세를 추정하기 위

해 보간 방법을 사용하지 않으면 집합 템플릿에 한정되어 있는 별도의 자세만 추정가능 하다. 둘째, 기하학적 방법은 얼굴 특징의 구성으로 머리자세를 추정하는 방법이다. 다섯 얼굴의 점 (각 눈의 바깥 쪽 모서리, 입의 바깥 모서리와 코끝)을 사용하여 입의 중간점과 눈의 중간점 사이에 선을 연결하여 얼굴 대칭축을 찾을 수 있다. 얼굴 방향은 코의 3D 각도에서 약한 원근법 기하학적 구조에 따라 결정될 수 있다. 또 다른 자세 추정은 다른 구성의 다섯 점 (각 눈의 안쪽과 바깥 쪽 모서리와 코끝)을 이용해서 네 개의 눈 점을 동일 평면으로 가정한다. 이때 Yaw는 왼쪽과 오른쪽 눈 사이의 크기에서 차이에서 추정할 수 있고, Roll은 수평선과 눈 라인의 각도에서 찾을 수 있다. 마지막으로 피치는 코 끝과 눈 라인 사이의 거리를 비교하여 결정할 수 있다. 기하학적 방법의 장점은 몇 가지 얼굴 특징으로도 빠르고 간단하게 괜찮은 머리 자세 추정치를 얻을 수 있다는 점이다. 하지만 단점은 얼굴 특징 검출에서 높은 정밀도와 정확도를 필요로 한다는 점과 누락 된 특징점의 처리문제가 있다. 세 번째 방법인 추적 기반 방법은 비디오 영상의 연속적인 프레임 사이 머리의 상대적인 움직임을 추적하여 머리자세를 추정하는 방법이다. 이러한 시스템은 일반적으로 높은 수준의 정확도를 보인다. 하지만 머리 위치 초기화가 필수이며, 일반적으로 추적 대상은 시스템이 시작되기 전에 정면 자세를 유지해야 하고, 추적이 실패 될 때마다 다시 초기화해야 한다. 초기의 추적 방법은 6개의 특징점을 추적 하는 방법으로 머리의 움직임을 결정했다. 모델 기반 접근 방식을 사용하기도 하는데, 머리 자세 추정에서 머리의 강체 3D 모델을 사용하는 것이 일반적이다. 3D 모델 위에 머리의 이미지를 텍스처 매핑(texture mapping) 하여, 이전 프레임과 새로운 프레임 모델 사이의 외관 차이를 최소화 하는 변환의 개별 집합을 검색하여 머리 자세를 추정 할 수 있다. 추적 기반 방법의 장점은 비디오 프레임 사이의 작은 자세 변화가 있어도 높은 정확도로 머리를 추적 할 수 있다는 점이다. 그리고 동적으로 사람 머리의 원형을 구축하여 모양 변화로 인한 악영향을 피할 수 있다. 단점은 기존 모델을 적용하거나 새로운 모델을 생성할 때 정확한 초기화가 필요하다는 점이다. 이점을 해결하기 위해서 기존 모델의 초기화 위치에 근접했을 때 동적 템플릿을 사용하여 자동으로 초기화하는 방법을 사용한다. 이런 기술은

다양한 환경에서 실시간으로 정확하게 머리 자세 추적을 가능하게 한다[10]. 마지막으로 얼굴모델 기반 방법은 2D 평면 모델과 3D 모델을 사용하는 방법이 있다. 초기에는 2D 평면 모델을 주로 적용하였지만 2D 평면 모델은 정면 얼굴에 제한된다는 점과 다양한 얼굴 회전에 적용이 어렵다는 단점이 있다. 제한적인 2D 얼굴 추적의 단점을 극복하고자 3D 얼굴 모델을 기반으로 하는 연구가 활발히 진행되고 있다. 3D 모델 기반 방법에는 실린더(cylinder), 타원(ellipse) 또는 얼굴자체 등을 기반으로 하는 모델을 설정한다. 그 중에서 3D 실린더 모델은 [11]에 의하여 제안된 방식으로 3D 공간상의 얼굴을 3D 실린더로 근사하여 모델링하고 이를 2D 평면으로 원근 투영(prospective projection)하는 방법을 사용한다. 3D 공간에서 추정된 얼굴 각도의 변화가 원근 투영된 2D 영상에 반영하는 방법에 의해 얼굴 이동과 회전 등의 변화를 추적할 수 있다. 이러한 방법은 2D 텍스쳐 맵을 몇 개의 영상들의 선형 조합해서 형성하기 때문에 각 영상들을 정합하고 워핑하는 과정과 각 계수를 추정하는데 계산량이 많이 필요하다.

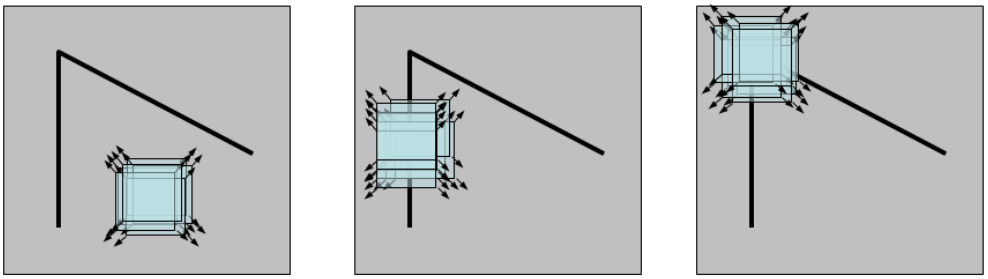
2.2.3 Harris Corner Detector

영상에서 코너 포인트는 직선과 더불어 영상 분석을 위한 중요한 정보를 제공하게 된다. 일반적으로 영상에서 코너 포인트는 특정한 객체와 다른 객체 또는 배경 사이에 존재하게 되므로 이러한 코너 포인트를 정확하게 검출하는 문제는 영상 내의 특정 객체의 위치와 형태에 대한 정보를 정확하게 구하는 것과 같다고 할 수 있다. 현재까지 코너 포인트를 찾기 위한 다양한 연구들이 진행되고 있으며 그 중 가장 대표적인 방법으로 Harris corner detector[12] 라고 할 수 있다. Harris corner detector 알고리즘은 기본적으로 영상 내 객체 내에 작은 직사각형 윈도우를 씌우고 상하 좌우로 움직이면서 윈도우 내의 화소값의 변화를 분석하여 코너 포인트를 찾는 방법을 이용하여 코너 포인트를 검출한다. 영상 내 객체의 밝기값의 변화가 없는 영역은 homogeneous 영역으로 추정하는데 이 때 윈도우를 상하좌우로 움직여도 화소값은 거의 일정할 것이다. 이와 달리 수평으로 이루어진 영상의 경계선이 있는 경우에는 윈도우가 좌우로 움직일 때 화소값의 변화가 크기 때문에 윈도우 내의 변화값이 크지만 상하 방향으로 움직이는 윈도우에 대해서는 화소값의 변화가 없다. 따라서 상하좌우 방향, 즉 모든 방향으로 움직이는 윈도우 내에서 임계값 이상의 화소값의 변화가 있는 경우를 코너 포인트라고 할 수 있다. 다음 〈그림 6〉에서 빨간 원으로 표시된 부분이 코너 포인트를 나타낸다.



〈그림 6〉 영상에서 나타난 코너 포인트

다음 〈그림 7〉은 편평한 영역, 경계선 그리고 코너 포인트에서 윈도우의 움직임에 따른 화소값의 변화값의 정도를 설명한 것이다. 편평한 영역 즉 homogeneous 영역에서는 화소값의 변화가 거의 없을 것이고 경계선에서는 윈도우가 경계선을 따라 움직일 때 화소값의 변화가 크고 그에 비해 코너 포인트들은 윈도우가 움직이는 모든 방향에서 화소값의 변화가 크다는 것을 알 수 있다.



〈그림 7〉 각 영역의 종류에 대한 화소값의 변화

Harris corner detector의 기본 알고리즘을 수학적으로 분석하기 위해 영상에서 임의의 한 픽셀값을 $I=(x,y)$ 이라 하고 이에 대한 위치 변화량을 $(\Delta x, \Delta y)$ 라 하면 윈도우 내에서 밝기 변화량을 다음과 같은 함수로 나타낼 수 있다.

$$E(u,v)=\sum_{x,y} w(x,y)[I(x+\Delta x, y+\Delta y)-I(x,y)]^2 \qquad (1)$$

위의 식(1)에서 윈도우 함수는 직각

픽셀이 위치 변화량, $(\Delta x, \Delta y)$ 만큼 움직인 영역을 Taylor 확장을 이용하여 나타내면 다음과 같다.

$$I(x+\Delta x, y+\Delta y) \approx I(x,y)+[I_x(x,y)\, I_y(x,y)]\begin{bmatrix}\Delta x \\ \Delta y\end{bmatrix} \qquad (2)$$

여기서 $I_x(x,y), I_y(x,y)$은 각각 x, y에 대한 미분값을 나타낸다. 위의 식(2)을 맨 처음 식(1)에 대입하여 정리하면 다음과 같다.

$$c(x,y) = [\Delta x \, \Delta y] \begin{bmatrix} \sum_W (I_x(x,y))^2 & \sum_W I_x(x,y) I_y(x,y) \\ \sum_W I_x(x,y) I_y(x,y) & \sum_W (I_y(x,y))^2 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \tag{3}$$

위의 식(3)에서 가운데 행렬을 $C(x,y)$ 라고 하고 이를 이용하여 코너의 정도를 판단하게 된다. 코너의 정도는 다음 식에 의하여 결정되고 각 픽셀에서 이 값이 0보다 큰 값을 가지게 되면 코너 포인트로 판단하게 된다.

$$O(x,y) = \det(C(x,y)) - k[trace(C(x,y))]^2 \tag{4}$$

$$HCM(x,y) = \begin{cases} 1 & \text{if } O(x,y) > 0 \\ 0 & otherwise \end{cases} \tag{5}$$

2.2.4 Lucas-Kanade Tracking

영상 내부의 모든 픽셀에서 속도를 구하는 것을 dense optical flow라고 하는데 이런 optical flow의 대표적인 알고리즘이 Lucas-Kanade 방법이다[13]. 〈그림 8〉은 3D 공간에서의 회전체의 optical flow를 각각 3차원과 2차원에서 나타낸 것이다.

Lucas-Kanade 알고리즘은 3가지 가정이 성립해야 적용할 수 있다. 첫째, 밝기 항상성(brightness constancy)으로 어떤 객체 상의 픽셀은 프레임이 바뀌어도 그 값이 변하지 않는다. 가령 그레이 영상의 경우 추적하고 있는 객체 픽셀의 밝기는 변하지 않는다는 것이다. 둘째, 시간 지속성(temporal persistence)인데 영상 내에서 움직임이 충분히 느리다고 가정하는 것이다. 즉 영상에서 객체의 움직임에 비해서 시간의 변화가 더 빠르게 진행되어 인접한 프레임 사이의 객체의 이동이 크지 않다는 가정이다. 셋째, 공간 일관성(spatial coherence)로서 공간적으로 서로 인접하는 점들은 동일한 객체에 속할 가능성이 높고 동일한 움직임을 갖는다는 가정이다. 이러한 가정들에 의해서 optical flow 방정식은 픽셀 p를 중심으로 하는 윈도우 내에서 모든 픽셀들에 대해서 유효하다고 할 수 있다.

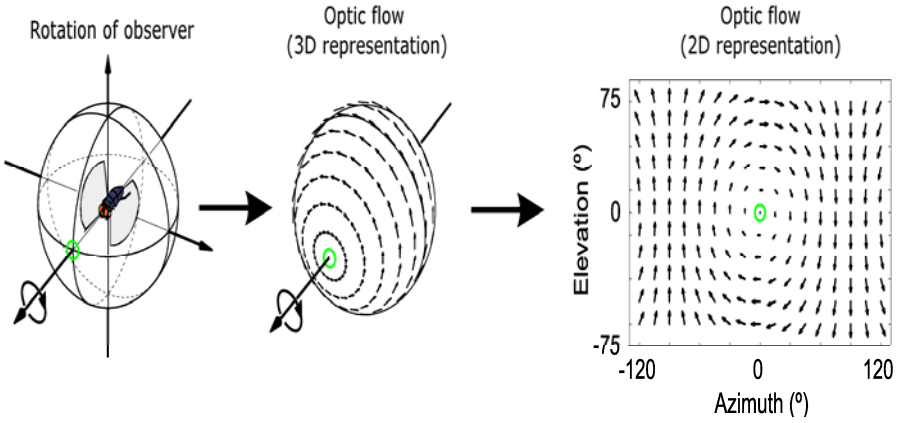


그림 8 3D 회전체에서의 각 축 방향으로 표시된 optical flow

2D 영상의 경우에 시간 t에서 (x, y)에 픽셀값을 $I(x, y, t)$라고 정의한다. δt 시간동안 픽셀의 밝기는 변하지 않고 위치의 변화, $(\delta x, \delta y)$만 일어난다고 가

정하고 그 때의 픽셀값을 $I(x+\delta x, y+\delta y, t+\delta t)$라고 하면 다음과 같은 식이 성립한다.

$$I(x, y, t) = I(x+\delta x, y+\delta y, t+\delta t) \tag{6}$$

위의 식(6)에서 오른쪽에 있는 항을 Taylor 급수 전개를 하면 다음과 같은 식이 된다.

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x, y, t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + \text{H.Q.T} \tag{7}$$

앞의 가정이 성립하기 위해서는 위의 식에서 미분식의 합이 0이 되어야 한다. 즉, 다음을 만족해야 한다.

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \tag{8}$$

위의 식(8)에서 시간축으로의 움직임의 변화를 구하기 위해서 시간축 편미분을 적용하면 다음과 같다.

$$\frac{\partial I}{\partial x}\frac{\delta x}{\delta t} + \frac{\partial I}{\partial y}\frac{\delta y}{\delta t} + \frac{\partial I}{\partial t}\frac{\delta t}{\delta t} = 0 \tag{9}$$

이때 $\delta x/\delta t, \delta y/\delta t$는 각각 x, y 방향의 속도 V_x, V_y이고 $\partial I/\partial x, \partial I/\partial y, \partial I/\partial t$는 각각 영상 I에 대한 각 방향에 대한 편미분 I_x, I_y, I_t 이다. 따라서 위의 식(9)은 다음과 같이 정리할 수 있다.

$$I_x V_x + I_y V_y = -I_t \tag{10}$$

이러한 관계를 영상에 각 픽셀에 적용하면 다음과 같이 행렬 방정식이 된다.

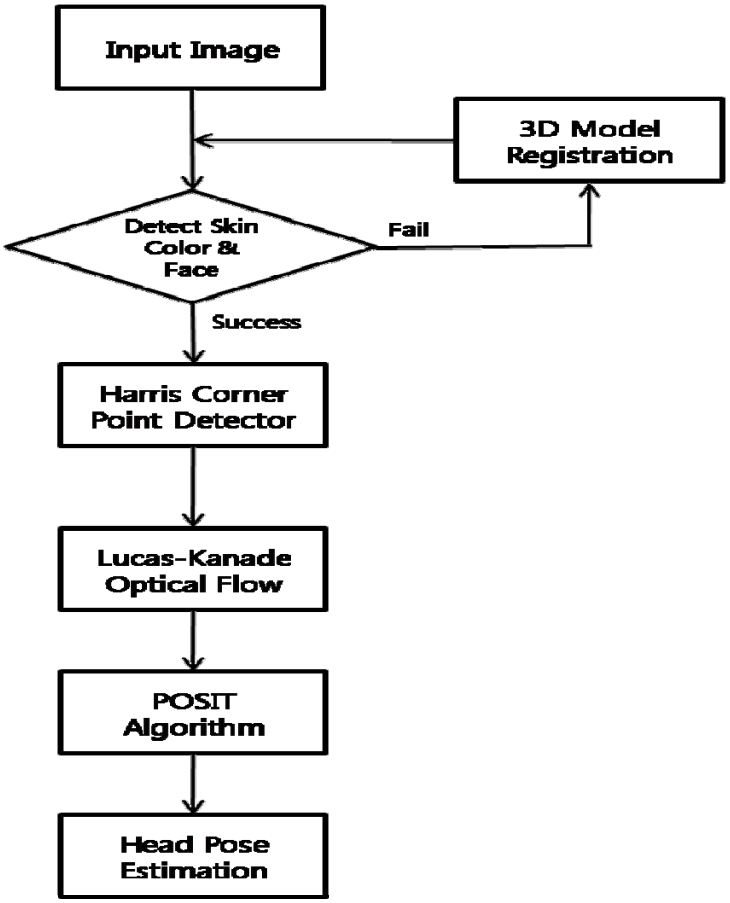
$$\nabla I^T \cdot \vec{V} = -I_t \tag{11}$$

위의 행렬 방정식을 최소 자승방법에 의해 풀면 움직임 벡터 $\vec{V}$를 구할 수 있다. 특히 Lucas-Kanade optical flow에서는 영상에서 각 픽셀마다 그 점을 중심으로 하는 윈도우를 설정하고 그 윈도우 단위로 위의 식들을 적용해서 움직임 벡터를 구하는 것이다.

Ⅲ. 3차원 얼굴 추적을 이용한 머리 자세 추정

3.1 제안 시스템의 개요

본 논문에서 제안한 방법은 특징점 기반 추적 방법에 의한 머리 자세 추정 알고리즘이다. 3차원에서의 움직임 벡터, $\mu = [\theta_x, \theta_y, \theta_z, t_x, t_y, t_z]$이라 하면 여기에서 $\theta_x, \theta_y, \theta_z$는 각 축에 대한 회전 정도를, t_x, t_y, t_z는 이동 정도를 나타내는 값이다. 이 움직임 벡터를 추정하는 것이 머리 자세 추정의 최종 목표이다. 다음 〈그림 9〉는 본 논문에서 제안하는 머리 자세 추정 알고리즘의 전체적인 블록도를 나타낸 것이다.



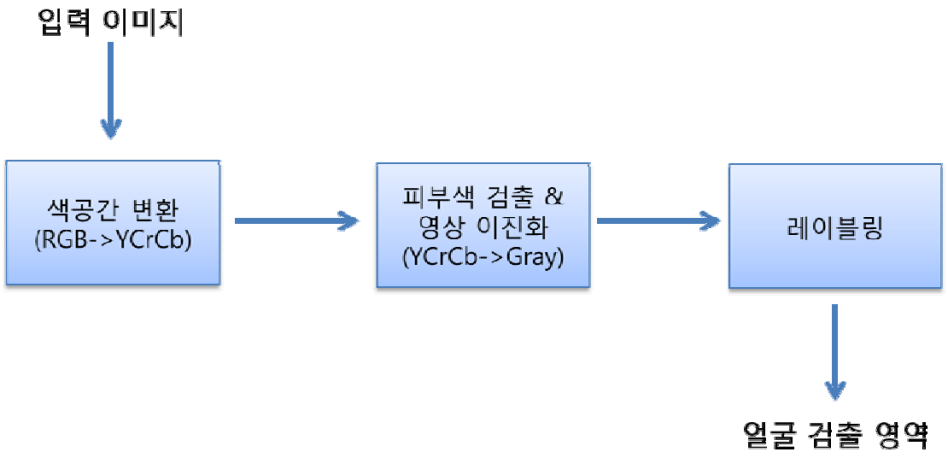
〈그림 9〉 Block Diagram of Head Pose Estimation

웹캠이나 모바일 기기로 촬영한 비디오 스트림으로부터 프레임 단위로 입력 영상을 추출하고 각 프레임은 설정되어 있는 3D 모델 포인트를 이용하여 머리 자세를 추정한다. 우선 입력 영상으로부터 관심 영역인 얼굴 영역을 검출하기 위해서 스킨 컬러를 이용한 검출 방법을 이용한다. 기존의 Viola-Jones[14] 방법은 정면 얼굴 검출에는 비교적 좋은 성능을 보여 주지만 얼굴의 각도에 매우 민감하기 때문에 사용자가 다양한 포즈를 취할 경우에 엄굴 검출이 되지 않는 경우가 많았다. 이에 본 논문에서는 스킨 컬러를 이용해 얼굴을 검출하여 다양한 자세에 적용할 수 있었다. 그리고 얼굴을 검출하지 못한 경우에는 포즈에 큰 변화가 있는 경우로 간주하여 그 때 입력 프레임 기반으로 다시 3D model의 특징점을 갱신하여 다음 프레임 영상들에 적용할 수 있도록 하였다. 다음으로 검출된 얼굴 영역에서 Harris Corner Detector를 적용하여 특징점들을 추출하고 Lucas-Kanade optical flow 방법을 이용하여 참조 프레임 사이의 모션을 추출하였다. tracking 과정에서 추출된 대응점들을 POSIT 알고리즘을 이용하여 3D model 포인트에 대응하는 3차원 점들을 구하여 회전 행렬(rotation matrix)와 이동 행렬(translation matrix)를 구한다.

본 논문에서 제안한 방법에서는 스킨 컬러를 이용한 얼굴 검출 방법과 검출 결과에 따라 트래킹에 사용하는 3D 모델과 참조 영상을 갱신함으로서 보다 강인한 얼굴 자세 추정 알고리즘을 구축했다고 할 수 있다.

3.2 레이블링을 이용한 얼굴 검출 방법

얼굴의 특징점을 찾기 위해서는 얼굴검출의 단계가 선행되어야 한다. 얼굴 검출 방법에는 다양한 방법이 있으나, 본 논문에서는 그중에서 피부색을 이용한 얼굴 검출 방법을 이용하였다. 피부색 검출 방법은 다른 얼굴 검출 방법보다 비교적 영상처리 단계가 간단하다. 그리고 적절한 전처리 과정을 거친다면 빠르고 높은 정확도로 얼굴검출을 구현할 수 있어 실시간 환경에 적합한 방법으로 알려져 있다. 본 논문에서 제시하는 방법은 프레임 영상이미지에서 피부색과 다른 색의 화소를 분류하고, 분류된 이미지에서 레이블링(labeling)을 통해서 얼굴영역을 검출한다. 자세한 과정은 다음 〈그림 10〉과 같이 세 단계로 나눌 수 있다.

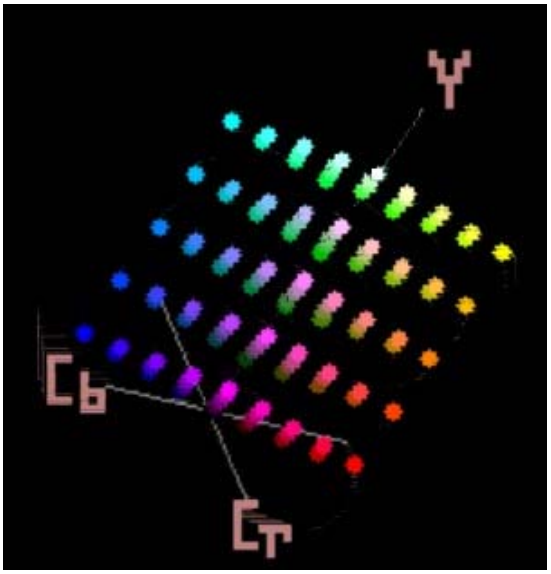


〈그림 10〉 얼굴 검출 과정

먼저 하나의 프레임 영상에서 피부색 영역을 얻기 위해서는 피부색에 해당하는 색공간의 범위를 제한하여야 하는데, 이때 피부색을 찾기 위해 어떤 색공간을 선택하느냐가 중요하다. [15]는 피부색을 이용한 얼굴 검출에 적합한 색공간을 선택하기 위해여, 1100가지의 실험 이미지를 대상으로 주로 사용하는 대표적인 색공간인 RGB(Red/ Green/Blue), HSV(Hue/Saturation/Value), YCrCb(luminance/Chroma-red/Chroma-blue) 별로 피부색을 이용해 얼굴을 검출하는 실험을 하였다. 그 중 YCrCb 색공간이 얼굴 검출율 83.91%의 정확도로 다른 색공간(RGB 56.46%, HSI 82.27%) 보다 높았으며 영상처리 속

도 또한 빨랐다. 이에 본 논문에서도 피부색 검출에 적합한 YCrCb 색공간을 이용하여 피부색을 검출하였다.

YCrCb 색공간은 다음 〈그림 11〉과 같이 Y, Cr, Cb로 구성된 3차원 색공간이다. 여기서 Y는 명도를 의미하며 색상과는 관련이 적은 밝기 성분을 나타내며, Cr과 Cb는 각각 붉은색 채도, 푸른색 채도의 색차 성분을 나타낸다. 3차원의 색공간 중에 피부색 판단에 영향을 많이 끼치는 부분은 Cr, Cb 차원의 영상이다. 카메라를 통해 전달된 RGB영상을 YCrCb 영상으로 변환 후, 두 1차원 영상을 기준으로 잘 알려져 있는 적합한 범위 133〈Cr〈173 , 77〈Cb〈127를 만족하는 화소의 영상만 남겨두고 다른 화소는 레이블링을 위해서 검정색으로 값을 변경하여 구분한다.



〈그림 11〉 YCrCb 색공간

레이블링은 인접한 화소들에게 같은 번호(Label)를 붙여서 연결되지 않은 다른 성분과 구분을 하는 작업으로, 영상에서 객체를 검출할 때 주로 사용된다. 레이블링 작업시 필요한 이진화 영상은 이전에 피부색이 검출된 YCrCb 이미지를 Gray 색공간으로 이진화 시킨 후 사용한다. 레이블링 작업의 순서는 화소검색 단계, 분류 단계, 그룹선택 단계로 나눌 수 있다. 먼저, 이진화 영상 이미지를 기반으로 이미지 전체 영역에서 흰색 값을 갖는 화소를 중심

으로 인접한 영역에 흰색 화소들이 있는지 판별 한다. 그리고 인접화소들 간에 같은 번호를 붙여서 분류한 후, 같은 번호를 가지는 그룹에 대한 정보를 저장한다. 마지막으로 저장되어 있는 그룹별 사각형 영역(x, y, width, height)의 정보를 이용해 영역의 위치와, 영역의 넓이를 기준으로 얼굴 영역 후보들을 선택하였다.

3.3 온라인 참조 프레임 등록

온라인으로 트래킹하는 과정에서 참조 프레임은 현재 프레임에서 머리 자세 추정에 필요한 중요한 기준점들을 제공하기 때문에 매우 중요하다. 특히 머리 자세가 급격하게 변할 경우에 현재 참조 프레임은 다음 비디오 시퀀스에 적절하지 못할 수도 있고 그 결과로 트래킹에 실패하여 잘못된 추정 결과가 나오게 된다. 이러한 경우에 강인한 머리 자세 추정 시스템을 구축하기 위해 본 논문에서 참조 프레임을 트래킹하는 과정에서 다시 선정하여 보다 정확한 결과를 얻을 수 있었다.

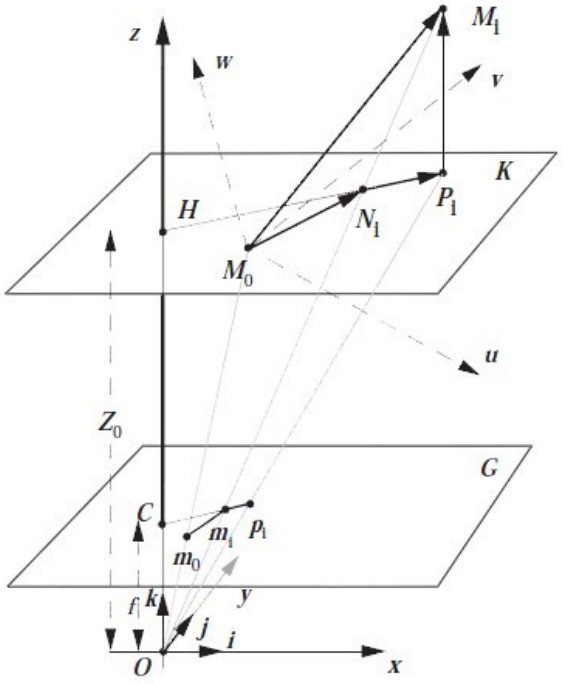
비디오 입력이 들어오면 첫 번째 프레임은 자동적으로 참조 프레임으로 등록되고 그 다음 프레임들에 대해 기준이 되는 3D 모델 포인트와 특징점들을 가지게 된다. 즉 참조 프레임을 T_i 라하면 프레임 단위로 3D 모델 포인트들, M_j와 이를 2차원 영상으로 투영한 점들, m_j를 가진다. 이러한 특징점들은 Harris Corner Detector를 이용해서 추출하며 이 점들의 유효성을 높이기 위해서 RANSAC (RANdom SAmple COnsensu)를 이용해서 선택하는 방법을 부분적으로 적용하였다.

온라인 참조 프레임 등록을 이용할 때 고려해야 하는 경우가 있는데 이는 머리 자세 추정이라는 목표의 관점에서 얼굴의 표정이 바뀌는 경우와 머리 자세가 바뀌는 경우 모두 기존의 참조 프레임의 특징점들을 다시 갱신해야 하는 결과로 나온다는 것이다. 머리 자세 추정은 얼굴의 표정에는 무관하게 단지 머리의 위치와 관련이 있는 것이기 때문에 이러한 경우에 추정이 잘못될 수 있다.

또한 본 논문에서는 온라인 참조 프레임 등록만이 아닌 트래킹하는 과정에 특징점들을 추가하여 현재 참조 프레임을 유지하면서 좀 더 많은 정보를 제공하고 특징점 수를 특정하지 않고 프레임 간의 특성에 따라 보다 유연하게 운영하도록 했다.

3.4 POSIT 알고리즘을 이용한 3차원 모델

POSIT 알고리즘은 단일 영상으로부터 객체의 위치를 구하는 방법[16]이다. 이 방법에서는 사용자가 영상으로부터 단일 평면위에 존재하지 않는 4개 이상의 특징점들과 그들 사이의 상대적인 기하학적 구조를 알고 있다고 가정한다. 또한 이 방법은 두 가지 방법의 결합으로 볼 수 있는데 첫 번째 방법은 POS(Pose from Orthography and Scaling)이고 두 번째 방법은 POSIT(POS with ITerations) 이다. POS 알고리즘은 확대 또는 축소된 정사영의 투시 평면을 근사하여 선형 시스템의 해를 구하는 방법으로 객체의 회전 행렬 (rotation matrix) 과 이동 벡터 (translation vector) 를 구하는 방법이다. 이에 비해 POSIT 알고리즘은 위의 과정을 수회 반복하여 정확한 포즈 값에 수렴하도록 한다.



〈그림 12〉 모델 포인트 M_0에 대해 m_0의 정사영 투영

〈그림 12〉에서는 초점 거리(focal length) f와 영상의 중심점을 알고 있다는 가정하에서 정사영된 중심점 O와 영상 평면을 가진 핀홀 카메라 모델을 보여주고 있다. 카메라 프레임의 단위 벡터를 i, j, 그리고 k 라 하고 3D 모델의 특징점들을 $M_0, M_1 ..., M_i, ...M_n$이라 할 때 모델 좌표 프레임은 M_0를 중심으로 하며 단위벡터 u, v, 그리고 w를 가지게 된다. 모델 포인트 M_i은 모델 프레임에서는 알고 있는 좌표인 (U_i, V_i, W_i) 가 되고 카메라 프레임에서는 (X_i, Y_i, Z_i) 라는 미정의 좌표를 가진다. 이러한 M_i의 정사영 좌표를 m_i라 하면 주어진 시스템에서 m_i는 계산할 수 있으며 영상 평면에서 이 점은 (x_i, y_i) 좌표값을 가지게 된다. 이 때 pose 행렬은 다음과 같이 정의할 수 있다.

$$P = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} i_u & i_v & i_w & T_x \\ j_u & j_v & j_w & T_y \\ k_u & k_v & k_w & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

위에서 R은 모델 프레임에 대한 카메라 프레임의 원점의 회전 변환을 나타내는 회전 행렬이고 T는 이동 변환을 나타내는 이동 벡터이다. 회전 행렬 R은 각 행이 단위 벡터 (i, j, k)를 모델 프레임 좌표 시스템으로 변환한 좌표이다. 즉 회전 행렬 R은 모델 좌표를 카메라 시스템에서 정의된 좌표로 변환시킨다. 회전 행렬을 계산할 때 i, j 축 값을 구하면 $k = i \times j$ 이므로 행렬값을 모두 구할 수 있다. 이동 벡터 T는 벡터 OM_o가 (X_0, Y_0, Z_0)를 가지도록 정해지며 X_0, Y_0 값들을 알고 있으므로 Z_0만 구하면 된다. 정사영 투영 모델에서 3D 포인트 (X_i, Y_i, Z_i) 은 영상에서 (x_i, y_i) 좌표로 투영되는데 본 논문에서 객체의 깊이값 (depth) 은 카메라에서 객체까지의 거리에 비해 매우 작고 시각적으로 인지할 수 있는 점들은 광학적인 축에 가깝다는 가정에 근거하여 2차원 투영 좌표를 다음과 같이 구할 수 있다고 가정한다.

$$(x_i, y_i) = (\frac{f}{(1+\epsilon)}\frac{X_i}{T_Z}, \frac{f}{(1+\epsilon)}\frac{Y_i}{T_Z}) \tag{13}$$

위에서 ϵ 값은 실험에 의해 결정된다.

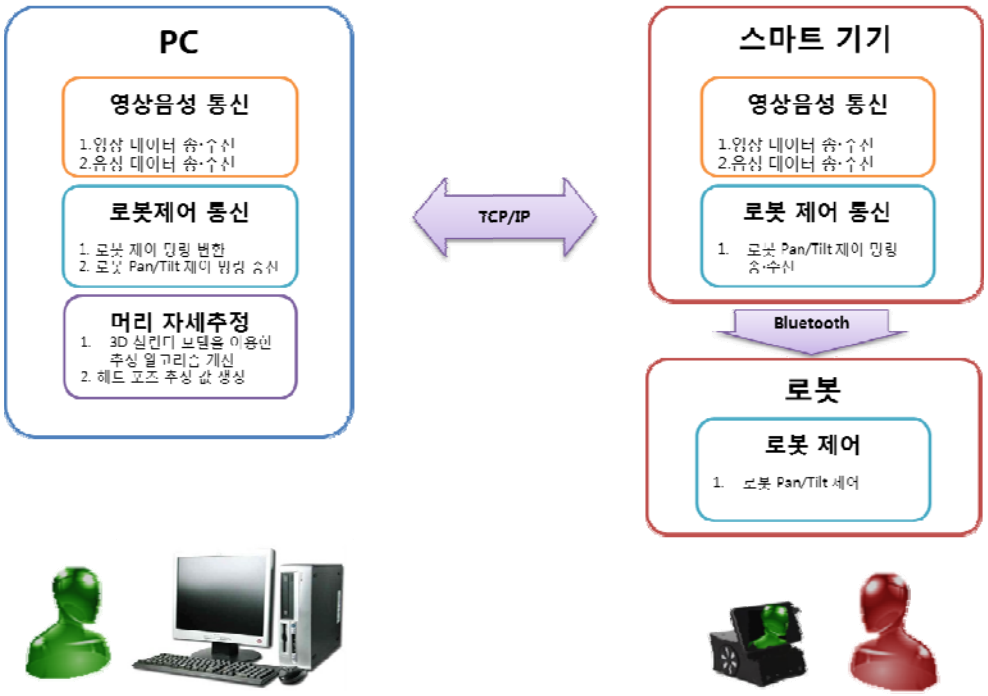
3.5 요약

본 장에서는 텔레프레즌스 시스템을 구현하기 위해서 비디오 시퀀스를 입력으로 한 머리 자세 추정 알고리즘을 제안하였다. 입력된 비디오 시퀀스를 프레임 단위로 처리하여 현재 프레임과 참조 프레임을 중심으로 특징점 기반 트래킹 방법을 이용하여 머리 자세 추정 알고리즘을 구현하였다. 입력 영상에서 얼굴 검출 단계는 전체 시스템의 수행에 큰 영향을 주기 때문에 스킨 컬러를 이용하여 다양한 머리 자세에 적용할 수 있게 하였다. 종래의 방법에 비해 얼굴 검출 능력이 표정이나 각도에 강인한 특성을 가지게 되었다. 특히 온라인 참조 프레임 등록을 통하여 변화하는 얼굴 자세와 주변 환경의 변화에 적용할 수 있도록 하였다. 향후 좀 더 다양한 조명 변화로 인한 스킨 컬러의 변화와 일시적으로 얼굴이 가려진다는 등의 경우를 고려한 알고리즘 개발이 필요하다.

Ⅳ. 텔레프레즌스 화상회의 시스템 설계

4.1 통합 시스템의 구성

교감형 텔레프레즌스 화상회의 시스템을 구성하기 위하여 다음 〈그림 13〉과 같이 전체 시스템을 설계하였다. 물리적으로 떨어진 원격공간의 두 환경에 위치한 사용자들은 각각 PC와 스마트기기+로봇 형태의 장비를 텔레프레즌스 매개체 장비로 이용한다. PC와 스마트기기는 무선 네트워크로 연결되어 PC와 스마트기기 간에 영상, 음성 데이터와 로봇명령을 주고받음으로써 쌍방향의 사용자들에게 원격 환경에 대한 현장감을 줄 수 있도록 구성하였다.



〈그림 13〉 시스템 구성도

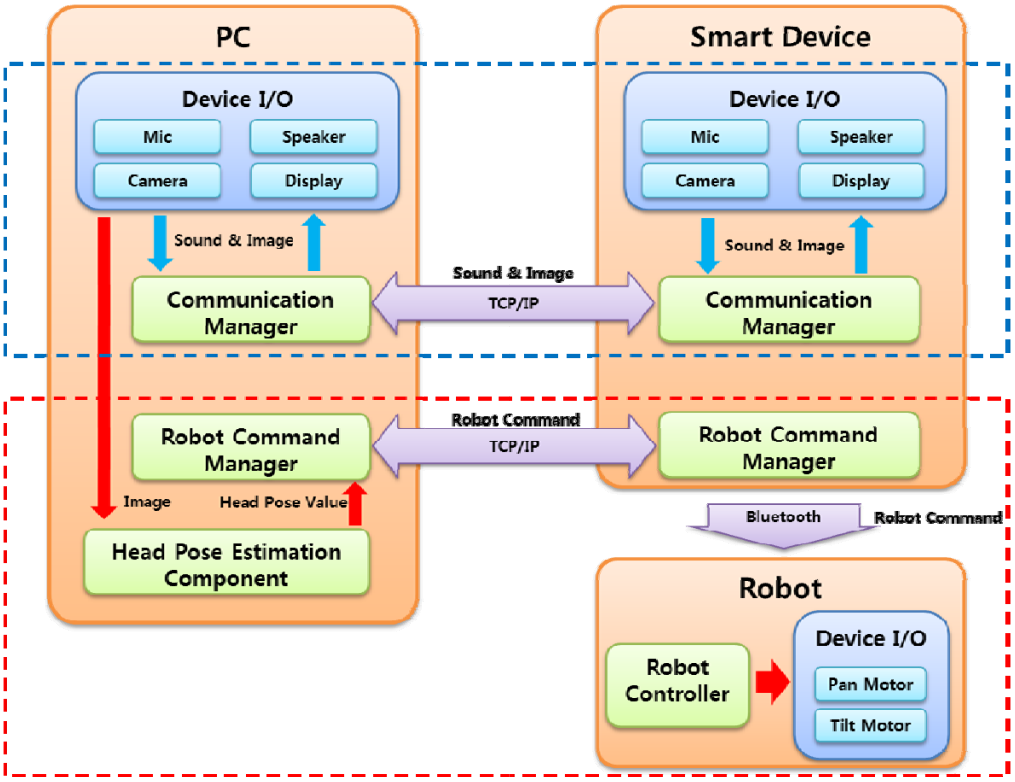
PC와 스마트 기기에는 공통적으로 서로 간의 영상, 음성 송수신 기능을 가지고 있다. 기기간의 통신은 TCP/IP 통신을 이용한다. PC 매개체 장비에서는

사용자의 머리자세를 추정하는 기능과 로봇제어명령을 스마트기기로 전달하는 기능이 있다. 이렇게 PC의 사용자는 원격에 위치한 사용자에게 영상, 음성정보로 전달할 수 있는 실재감에 추가된 비언어적인 제스처로서 머리자세를 전달할 수 있다.

스마트 기기에는 받은 로봇제어명령을 이용해서 로봇을 제어하는 기능이 있다. TCP/IP 통신을 이용해 PC와는 로봇제어명령을 수신만하고 로봇과는 bluetooth 통신으로 송신만 한다. 실제 로봇의 제어기는 스마트기기에서 제어값을 받은 그대로 모터를 제어하는 역할만 한다.

4.2 소프트웨어 아키텍처

본 시스템의 소프트웨어는 목적에 따라 영상 및 음성 통신부와 머리자세 추정 및 로봇제어부로 나누어 〈그림 14〉와 같이 설계하였다.



〈그림 14〉 소프트웨어 아키텍처

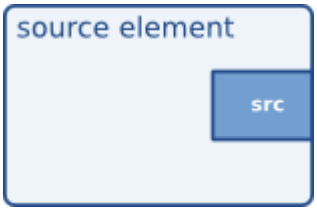
1) 영상 및 음성 통신부

PC와 스마트기기 간에 영상(image)과 음성(sound) 데이터를 주고받는 부분이다. 주로 각 매개체 장비들과 장비 내부의 I/O(camera, mic, speaker, display)들 간의 sound, image data를 주고받는 기능을 설계한다. 영상 송수신은 영상을 H.264 코덱으로 압축하여 RTSP(Real Time Streaming Protocol) 프로토콜로 전송한다. H.264 코덱은 영상 압축 표준의 하나로, 현재 고선명 비디오의 녹화, 압축, 배포를 위한 가장 일반적인 포맷 가운데 하

나이다. 매우 높은 데이터 압축률을 자랑한다[17]. RTSP 프로토콜은 RTP프로토콜(Real-time Transport Protocol)에 미디어 데이터를 제어하는 명령이 추가된 통신 규약이다. RTP 프로토콜은 IP 네트워크를 통해 오디오와 비디오를 전달하기 위한 표준화된 패킷 포맷을 정의한다. RTP는 스트리밍 미디어를 포함한 커뮤니케이션 및 엔터테인먼트 시스템에 널리 쓰이고 있다. RTP는 국제 인터넷 표준화 기구(IETF)의 오디오 비디오 트랜스포트 워킹 그룹이 개발하였으며 RTC 1889로 1996년 처음 출판되었으며 2003년에 RFC 3550가 RTP를 대신하게 되었다. 이러한 RTP 프로토콜에 미디어를 제어하는 명령을 추가한 프로토콜이 바로 RTSP이다[18, 19].

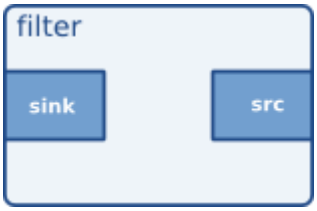
본 논문에서 구현한 영상 송수신방식은 요소(element)와 파이프라인을 기반으로 하여 구성되어 있다. 요소는 전체 영상송수신 단계에서 하나의 역할을 하는 블랙박스로서 디코더, 인코더, 디먹서, 비디오 혹은 오디오 출력 등을 말할 수 있다. 요소에는 다음과 같은 종류가 있다[20].

- 소스요소 : 소스요소는 데이터 입력을 받지 않고, 단지 데이터를 생성하는 요소이다. 영상의 카메라 혹은 마이크로부터 들어오는 데이터가 바로 소스요소의 대표적인 예이다. 소스요소는 아래와 같이 표시될 수 있다.



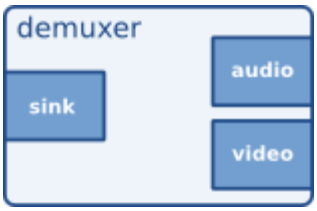
〈그림 15〉 소스요소의 구조

- 필터요소 : 필터요소는 입력 및 출력을 모두 가지고 있는 요소이다. 이 요소는 수신된 데이터에 어떠한 처리과정을 한 후 이를 출력하는 기능을 가지고 있으며, 이러한 요소의 예로는 볼륨소자, 영상 스케일러, 컨버터 등이 있다. 필터 요소는 소스요소와 싱크요소 사이에 여러개가 존재할 수 있으며 아래와 같이 표시될 수 있다.



〈그림 16〉 필터요소의 구조

- 디먹서요소 : 둘 이상의 출력 패드를 가지고 있는 요소를 디먹서요소라고 한다. 디먹소요소의 대표적인 예가 들어온 입력 스트림으로부터 오디오와 비디오 신호를 분리하는 것으로서 아래의 그림과 같이 표현할 수 있다.

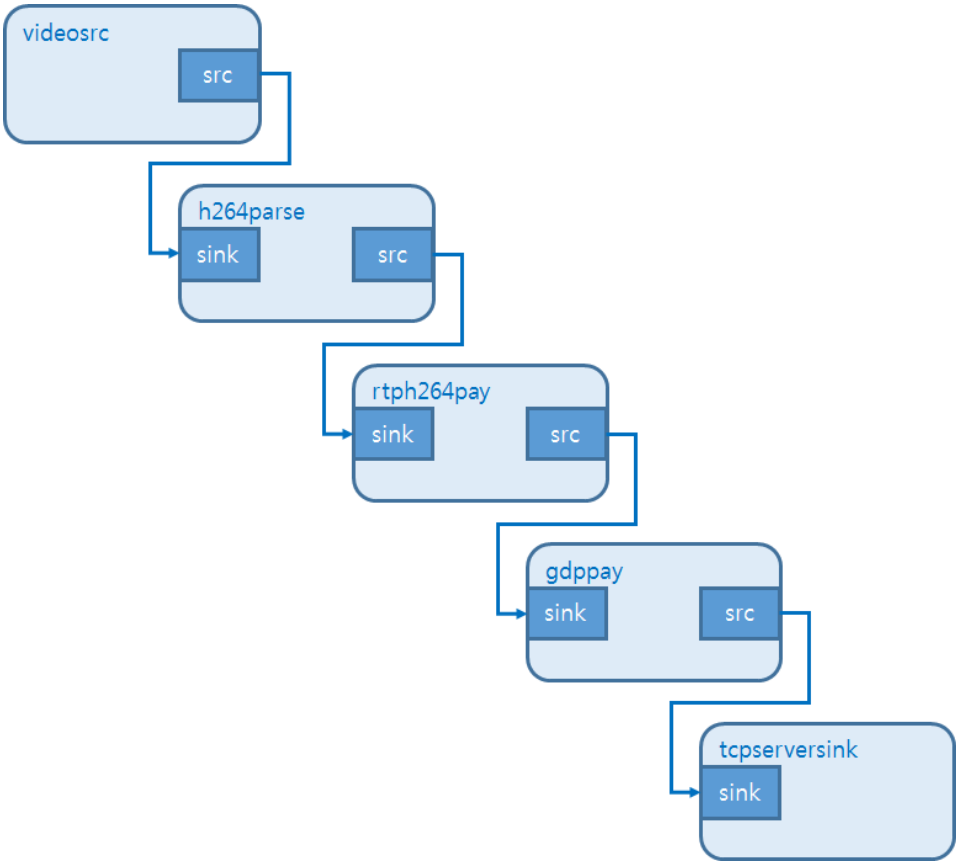


〈그림 17〉 디먹서요소의 구조

- 싱크요소 : 싱크요소는 미디어 파이프라인에서의 마지막 엔드포인트이다. 이 요소는 데이터를 수신하지만 아무것도 생산하지 않는다. 싱크요소의 예는 디스크 기록, 사운드 카드 재생, 비디오 출력 등이 있으며, 아래의 그림과 같이 표현할 수 있다.

〈그림 18〉 싱크요소의 구조

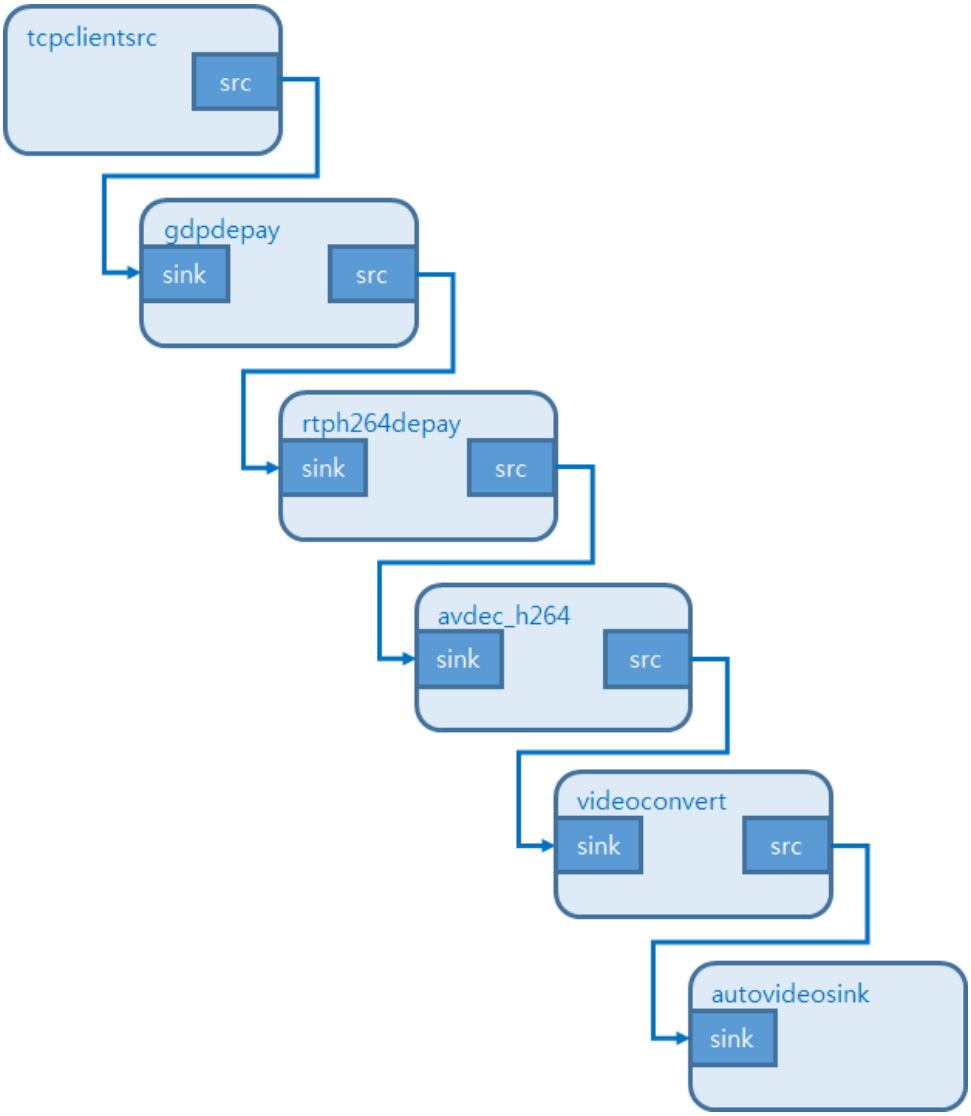
그리고 다음 〈그림 19〉와 〈그림 20〉은 영상의 송신과 수신 과정에서 각각의 요소들의 연결 관계를 나타낸 그림이다.



〈그림 19〉 영상 송신 블록

영상 송신 과정에서 각 요소들의 역할은 다음과 같다.

- videosrc : 카메라에서 영상 획득
- h264parse : 영상을 H.264 포맷으로 인코딩
- rtph264pay : H.264 영상을 RTP 패킷에 추가
- gdppay : 데이터 스트리밍 버퍼에 영상 추가
- tcpserversink : RTP 영상을 TCP 방식으로 전송



〈그림 20〉 영상 수신블록

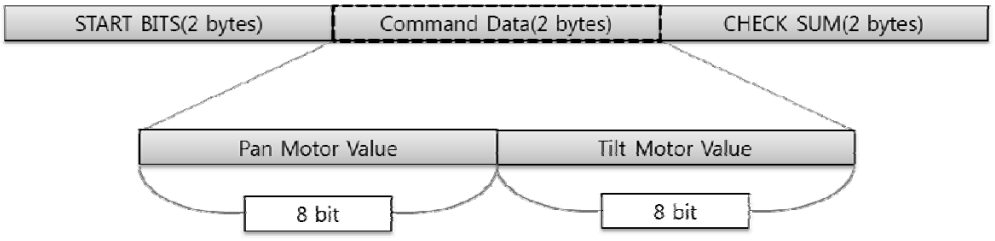
영상 수신과정에서 각 요소들의 역할은 다음과 같다.

- tcpclientsrc : 영상 스트리밍 수신
- gdpdepay : 데이터 스트리밍 버퍼에서 영상을 수신
- rtph264depay : RTP 패킷에서 H264 영상을 수신
- avdec_h264 : H264 영상을 디코딩
- videoconvert : 영상의 색공간 변환
- autovideosink : 영상을 출력장치로 표시

2) 머리자세 추정 및 로봇 제어부

PC에서 추정한 머리자세를 이용해 스마트기기에 제어명령을 전달하여 로봇을 제어하기 위한 부분이다. PC 쪽은 머리자세추정 컴포넌트(head pose estimation component)와 로봇명령매니저(tobot command manager)로 구성되며, 스마트기기 쪽은 로봇명령매니저(robot command manager)로 구성하였다. 본 논문에서 제안한 3D 실린더 모델 기반의 머리자세 추정 방법을 이용해서 웹캠에서 받아온 영상정보에서 머리자세를 추정하는 연산에 관련된 부분을 구분하여 PC 쪽의 머리자세추정 컴포넌트에 설계하였다. 그리고 머리자세추정 컴포넌트로부터 얻어지는 머리자세 값을 이용해서 실제 로봇의 모터를 제어하는 값으로 변환하는 기능과 변환된 값을 6 Byte 패킷으로 묶어 값을 보내는 기능, TCP/IP 통신의 서버기능을 PC쪽의 로봇명령매니저에 구분하여 설계하였다. 그리고 스마트기기 쪽의 로봇명령매니저에는 PC로부터 받아온 로봇제어명령 값을 받는 기능, TCP/IP의 클라이언트 기능, 블루투스 통신으로 로봇제어기에 받은 명령을 보내는 기능을 구분하여 설계하였다.

다음 〈그림 21〉은 PC - 스마트기기 - 로봇제어기 간에 전달하는 로봇명령 패킷이다. 6 byte의 패킷 하나로 Pan/Tilt 모터를 움직일 수 있도록 하였으며, 실제 로봇제어 명령은 2 byte를 사용한다. 그러므로 각각의 모터가 8bit(0~255)의 제어 값 해상도를 가진다. Pan/tilt의 최대, 최소 각도를 제한하여 사용 한다면 그만큼 모터의 정밀도를 더 높일 수 있을 것이다. 다음의 프로토콜을 기준으로 PC에서는 추정된 머리자세를 로봇제어명령으로 변환하는 기능을, 로봇명령 데이터의 종착지인 로봇의 제어기에는 스마트 기기에서 전송한 패킷을 해석하여 Pan/Tilt 모터를 제어하는 기능을 구현하였다.



〈그림 21〉 로봇 명령 패킷

4.3 구현 결과

1) 소프트웨어구현

본 시스템은 카메라 영상에서 얼굴을 검출하고 머리자세를 추정하는 과정에서 OpenCV와 OpenGL을 활용하였다. OpenCV(Open Source Computer Vision)는 인텔사에서 개발한 오픈소스 컴퓨터 비전 C 라이브러리이다. 기초적인 영상처리부터 고급 수준의 영상처리 알고리즘들이 함수로 구현 되어있고, 실시간 영상처리에 중점을 둔 라이브러리이다. 그리고 OpenGL(Open Graphics Library)은 2차원, 3차원 그래픽스 표준 API 규격으로 3차원 영상을 생성, 제어하는데 주로 이용한다. 소프트웨어 구현에 OpenCV와 OpenGL을 사용함으로써 영상처리 기능을 개발하는데 시간을 단축시킬 수 있었다.

PC 쪽 프로그램은 C++ 언어를 기반으로 개발하였고, 안드로이드는 Java 언어를 사용하였다. 머리자세 추정 과정에서 얼굴을 검출하기 위한 색공간 변환, 피부색 검출과 프레임마다 웹캠을 통해서 영상을 가져와 영상처리 하는 부분, 프레임 마다 특징점을 찾는 부분은 OpenCV라이브러리를 활용하였다. 그리고 OpenGL은 Posit알고리즘과 3D모델을 생성하여 추정 결과를 시뮬레이션 결과를 랜더링 하는 기능에 활용하였다.

스마트기기 쪽은 Android SDK와 Java언어를 기반으로 로봇을 제어하는 프로그램은 백그라운드로 상태에서도 동작하도록 하여서, 영상프로그램과 동시에 실행 가능하도록 구현하였다. 로봇제어기 쪽 소프트웨어는 임베디드C 언어로 구현하였다.

2) 하드웨어구현

PC쪽은 마이크 기능이 있는 웹캠을 추가한 형태로 구성하였고, 스마트기기는 갤럭시 S2 모델을 사용하였다. 로봇의 구성은 로보티즈사의 CM-530제어기와 Dynamixel AX-12 모터 2개, 블루투스 모듈을 사용하여 다음 〈그림 22〉과 같이 외형을 구성하였다. 모터 2개는 각각 pan/tilt 회전이 가능하도록

구성하였고, tilt 모터의 위치에는 스마트기기르 거치할 수 있는 형태로 거치대를 만들었다.

〈그림 22〉 로봇에 스마트기기가
장착된 모습

Ⅴ. 실험 결과

5.1 실험 환경

제안한 시스템의 적합성을 평가하기 위하여, 실시간으로 사용자의 얼굴추정과 원격지의 로봇 제어가 가능한지 실험하였다. 머리자세추정 프로그램의 성능평가 실험을 위해 Boston University의 BU dataset의 영상을 사용하였다. 이 영상들은 다음 〈그림 23〉에서처럼 각 사람들 마다 머리자세를 측정하는 장치를 장착하고 200프레임에 걸쳐서 자유롭게 움직인 영상들이다. 실험에서는 이중 7개의 영상을 기준으로 실험하였다.

〈그림 23〉 얼굴 추적을 위한 BU dataset

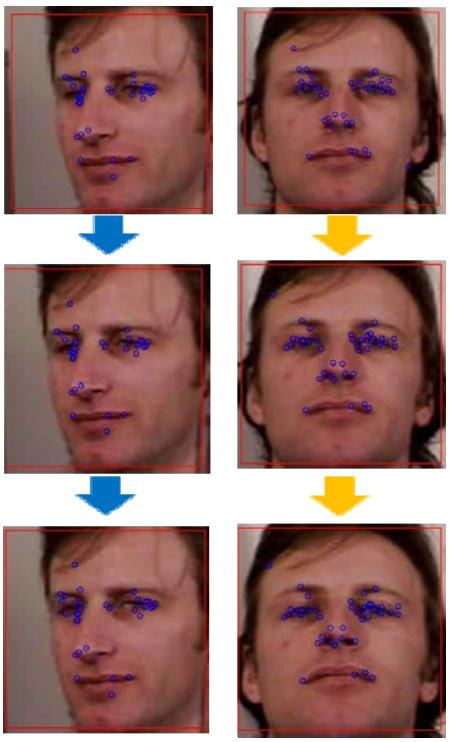
5.2 실험 결과

먼저 본 논문에서 머리자세 추정은 간단하게 얼굴검출, 특징점 검출, Posit을 이용한 자세 추정의 단계로 머리자세를 추정한다. 그래서 머리자세를 연속적으로 잘 추정하기위해서는 최초로 영상마다 얼굴검출이 잘 되어야 한다. 다음 〈그림 24〉는 구현된 프로그램의 얼굴 검출 단계에서 Viola-Jones방법과 논문에서 제안한 피부색 검출 방법을 이용하여 얼굴을 검출한 결과이다.

〈그림 24〉 얼굴검출 비교: 왼쪽-Viola-Jones 검출에서 실패한 경우, 오른쪽- 피부색 검출방법으로 검출된 경우.

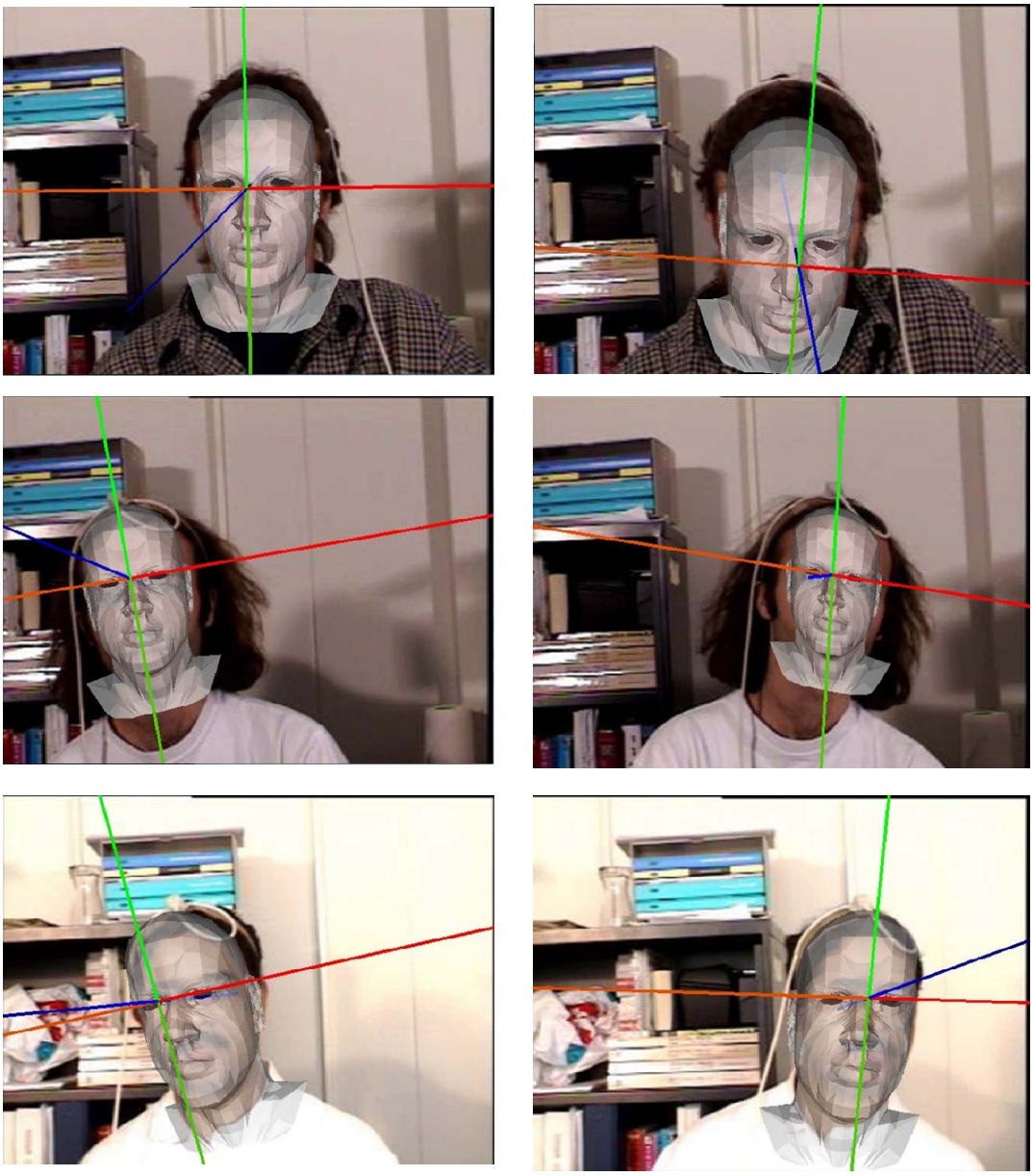
얼굴 검출에 성공하였으면 얼굴이 검출된 영역을 빨간색 사각형으로 표시하도록 구현하였다. 먼저 Viola-Jones방법을 적용하였을 때, 정면의 얼굴 영상에서는 얼굴을 잘 찾았으나 얼굴 각도가 중심에서 멀어질 경우 얼굴을 못 찾는 경우가 발생하였다. 이는 Viola-Jones의 얼굴 검출 방법이 정면 얼굴이외에 각도 변화에 민감하기 때문이다. 반면에 본 논문에서 제안한 피부색 검출 방법을 사용하였을 때에는 얼굴 각도 변화가 크더라도 얼굴 영역을 잘 검출하는 것을 볼 수 있었다.

다음 〈그림 25〉는 얼굴 영역이 잘 검출된 상황에서 매 프레임 마다 특징점을 찾는 단계에 대한 결과 그림이다. 그림에서 파란색 점은 현재 얼굴 영역에서 찾은 특징점의 위치를 표시한 것이고, 빨간색 선은 이전 프레임에서 검출된 특징점과 현재 프레임에서 동일하다고 판단된 특징점을 기준으로 그은 선이다. 연속적으로 일정 수 이상의 특징점들이 검출되었고, 이전 프레임과 현재 프레임 간의 동일한 특징점들도 연속적으로 잘 찾은 것을 볼 수 있다.



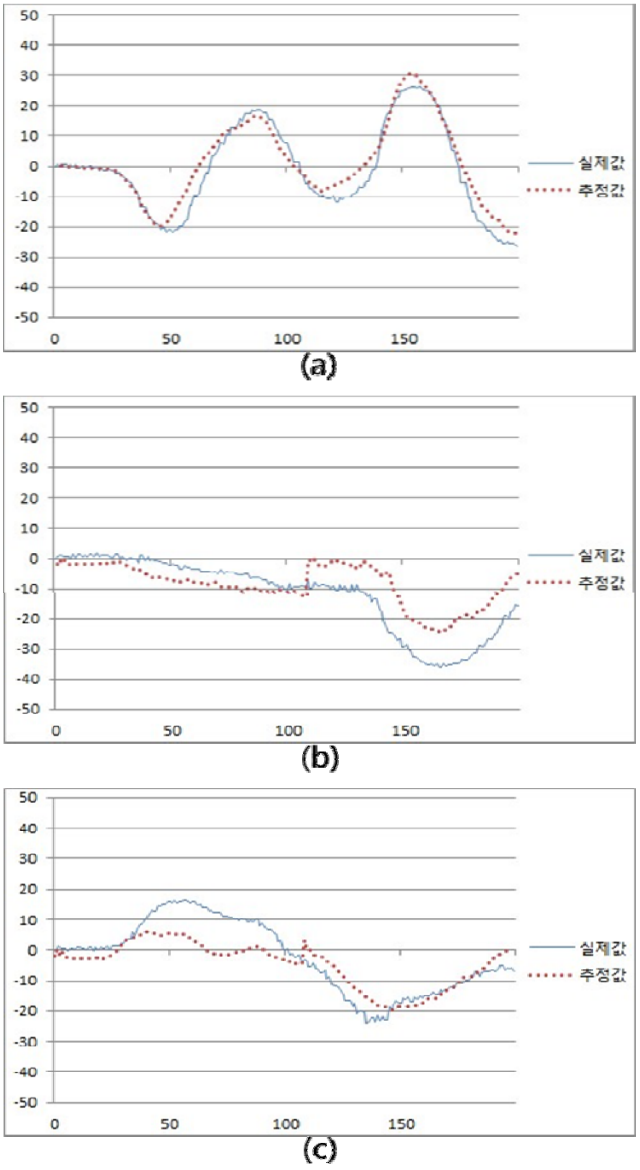
〈그림 25〉 프레임별 특징점 검출 결과

마지막으로 다음 〈그림 26〉은 특징점을 Posit 알고리즘에 적용시켜 머리자세를 추정한 후, 머리위치에 3D 모델을 시뮬레이션한 결과이다. 시뮬레이션한 모델의 스케일의 문제가 있었으나, 매 프레임 영상의 머리자세를 잘 추정하는 것을 확인할 수 있었다.



〈그림 26〉 머리자세 추정 후 모델 시뮬레이션

다음은 더 정밀한 머리자세의 추정 정확도를 측정하기 위해서 BU dataset의 영상의 한 프레임마다 실제 Roll, Yaw, Pitch 값에 대한 ground truth data와 머리자세 프로그램에서 추정한 값을 비교해 보았다. 다음 〈그림 27〉은 영상들 중에서 각각의 Roll, Yaw, Pitch 값의 변화가 빈번한 영상을 대상으로 실제값과 추정값을 비교한 그래프이다.



〈그림 27〉 머리자세추적 성능: (a) Roll, (b) Yaw, (c) Pitch

dataset의 7개 영상에 대하여 실험한 결과 평균 제곱근 오차(RMSE: Root Mean Square Error) 가 Roll, Yaw(pan), Pitch(tilt)의 각도에서 각각 4.38° , 8.97° , 10.42° 의 평균 추정 오차가 있었다. 특히, 머리 자세 추정과정에서 실제 머리자세의 각도가 클수록 추정의 정확도가 떨어지는 것을 확인할 수 있었다.

Ⅵ. 결 론

본 연구에서는 화상회의와 같은 교감형 텔레프레즌스 시스템의 구현에 있어서 요구사항을 만족하기위해, 텔레프레즌스 시스템에 로봇과 인간의 의사소통 과정에서 비언어적인 수단으로 많이 쓰이는 머리자세를 접목시킴으로써 화상회의에 활용할 수 있는 텔레프레즌스 시스템을 제안하였다. 단안 카메라와 컴퓨터 비전기술을 이용한 머리자세 추정 방법을 이용해 실시간 환경에서 적합한 머리자세 추정 방법을 제안하였다. 3D 실린더 모델의 머리 자세 추정 방법은 초기 머리 위치에 강인한 장점을 가지는 방법으로, 실린더 모델을 기반으로 매 프레임마다 얼굴 검출영역에서 특징점을 찾아서 Posit 알고리즘을 이용해 머리자세를 추정하였다. Posit 알고리즘으로 머리자세를 추정하는 과정에서도 머리자세DB에 참조 템플릿을 갱신하는 과정을 통해 다양한 머리자세를 수용할 수 있었다.

향후 연구 과제에서는 조명 변화와 같은 다양한 환경에서도 잘 동작하면서도 계산비용을 줄이는 알고리즘 측면의 개선과 얼굴 DB크기와 검색 시간 간의 적정한 균형을 찾는 연구가 필요하다. 또한 모바일 기기에 좀 더 최적화 시키기 위해, 머리자세추정 속도와 영상통신의 통신 속도 개선에 대한 연구가 필요할 것이다.

제안된 시스템을 통해 널리 보급된 스마트기기와 저가의 로봇 키트의 구성으로 원격 환경 간의 텔레프레즌스 화상회의 시스템을 구현할 수 있음에 따라, 텔레프레즌스 시스템을 접하기 힘든 일반 사용자들에게 교감형 텔레프레즌스 도구로서 활용될 수 있을 것으로 기대한다.

참 고 문 헌

[1] Minsky, M. “Telepresence,” Omni 2, 9 (June 1980), pp. 45-51.

[2] 유지환, “원격조종로봇(Telerobot)과 원격 현장감 (Tele presence)의 연구 동향,” 2008.

[3] “Rise Of The Telepresence Robots,” Forbes, July 2013.

[4] 이종화, 강신각. “텔레프레즌스 기술 표준화 동향,” 한국통신학회지 (정보와 통신) 29.12 (2012): 25-30.

[5] E. Paulos and J. Canny, “Social Tele-embodiment: Understanding Presence,” Autonomous Robots, vol. 11, pp. 87-95, 2001.

[6] F. Michaud, P. Boissy, H. Corriveau, A. Grant, M. Lauria, D. Labonte, R. Cloutier, M. Roux, M. Royer, and D. Iannuzzi, “Telepresence robot for home care assistance,” Proceedings of AAAI, 2006.

[7] T. Tsai, Y. Hsu, A. Ma, T. King, and C. Wu, “Developing a telepresence robot for interpersonal communication with the elderly in a home environment,” Telemedicine and e-Health,vol. 13, no. 4, pp. 407-424, 2007.

[8] W. Stiehl, J. Lieberman, C. Breazeal, L. Basel, R. Cooper, H. Knight, L. Lalla, A. Maymin, and S. Purchase, “The huggable: a therapeutic

robotic companion for relational, affective touch," in 3rd IEEE Consumer Communications and Networking Conference, 2006. CCNC 2006, vol. 2, 2006.

[9] Sigurdur Orn Adalgeirsson & Cynthia Breazeal, "MeBot: A Robotic Platform for Socially Embodied Telepresence," Proceedings of the HRI 2010.

[10] E. Murphy-Chutorian and M. M. Trivedi, "Head pose estimation in computer vision: A survey," IEEE Trans. Pattern Anal. Machine Intell., vol. 31, no. 4, pp. 607-626, 2009.

[11] M. L. Cascia, S. Sclaroff, and V. Athitsos, "Fast, reliable head tracking under varying illumination: An approach based on registration of texture-mapped 3d models," IEEE Trans. Pattern Anal. Machine Intell., vol. 22, no. 4, pp. 322-336, 2000.

[12] C. Harris and M. Stephens. "A combined corner and edge detector," In Proc. Alvey Conf., pp. 189-192, 1988.

[13] B. Lucas and T. Kanade. "An iterative image registration technique with an application to stereo vision," Int'l Joint Conf. on Artificial Intel., pp. 674-679, 1981.

[14] Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features," Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. Vol. 1. IEEE, 2001.

[15] Singh, Sanjay Kr, et al. “A robust skin color based face detection algorithm,” Tamkang Journal of Science and Engineering, 2003, pp. 227-234.

[16] D. DeMenthon and L. S. Davis, “Model-based object pose in 25 lines of code,” in European Conference on Computer Vision, 1992, pp. 335 – 343.

[17] Wikipedia. (2013). H.264 [Online]. Available: http://en.wikipedia.org

[18] Wikipedia. (2013). RTP [Online]. Available: http://en.wikipedia.org

[19] Wikipedia. (2013). RTSP [Online]. Available: http://en.wikipedia.org

[20] Gstreamer. (2013). Gstreamer Library [Online]. Available: http://gstreamer.freedesktop.org

ABSTRACT

A Cost-Effective Telepresence Video Communication Robot System Delivering Speaker's Head Motion by Employing 3D Cylinder Head Model

Jin, Yong-Gyu
Major in Information & Communication Engineering
Dept. of Information & Communication Engineering
The Graduate School
Hansung University

It was reported that a telerobot that communicates more than simply audio or video but also expressive gestures, body pose and proxemics allows for a more engaging and enjoyable interaction. To provide such social expression at a reasonable cost, this thesis proposes a cost-effective telepresence robot system for video communication which can deliver speaker's head motion through it's display stanchion. Head gestures such as nodding and head-shaking can deliver crucial information during conversation, and we can assume the eye-gaze of the speaker, which is known as one of the key non-verbal signals for interaction, from his/her head pose.

In order to develop an efficient head tracking method, a 3D cylinder-like head model is employed and the Harris corner detector is combined with the Lucas-Kanade optical flow to be suitable for extracting 3D motion information of the model. Especially, a skin color-based face detection algorithm is proposed to achieve robust performance upon variant directions while maintaining reasonable computational cost. The performance of the proposed head tracking algorithm is verified through the experiments using BU's standard data sets.

A design of the robot platform is also described in detail, as well as the design of supporting systems such as video transmission and robot control interfaces. Through some in-person experiments between apart sites it was found that the suggested telerobot system can deliver smooth head motion for more natural interaction.

【keywords】Telepresence Robot, Head Pose Estimation, Video Conferencing, Telepresence System, Human Robot Interaction