

論文2002-39CI-3-6

## 얼굴 방향과 손 동작 인식을 통합한 가상 공간에 존재하는 Agent들과의 상호 작용

(Interaction with Agents in the Virtual Space Combined by Recognition of Face Direction and Hand Gestures)

曹綱鉉\*, 金聖恩\*\*, 李仁浩\*\*\*

(Kang-Hyun Jo, Sung-Eun Kim, and In-Ho Lee)

### 요 약

본 논문에서는 인간의 행동을 컴퓨터에게 인식시켜 가상의 공간에 존재하는 에이전트(agent)들과 상호 작용이 가능한 시스템을 구현하였다. 이 시스템은 크게 행동을 인식하는 인식 시스템과 인식 정보를 통해 미리 구성된 가상 공간에 존재하는 여러 에이전트간의 상호 작용을 하는 시스템으로 구성되어있다. 인식 시스템은 동작자의 연속적인 행동을 CCD카메라로부터 입력 받아 각각의 프레임에 대해 머리와 손의 특징을 추출한다. 그리고, 추출된 정보를 연속적인 시간의 흐름에 대해 해석을 한 후, 동작을 인식한다. 상호 작용 시스템을 위해 동작자의 분신인 아바타(avatar), 자율적으로 행동하는 퍼피(puppy), 그리고 비자율적인 객체인 탁자, 문, 창문, 공과 같은 이동이 가능한 오브젝트(object)들이 존재하는 가상 공간을 구현하였다. 인식된 동작은 상호 작용 시스템을 통해 가상 공간의 아바타에게 전달이 된다. 아바타의 동작 천이는 상태 천이도를 바탕으로 이루어진다. 상태 천이도는 각각의 동작이 노드로 정의되고, 그 노드들을 종속적으로 연결한 그래프로 구성된다. 아바타는 문과 창문을 여닫고, 오브젝트를 잡거나 이동할 수 있다. 또 퍼피에게 명령을 내리거나 퍼피의 동작에 대한 응답을 할 수 있다.

### Abstract

In this paper, we describe a system that can interact with agents in the virtual space incorporated in the system. This system is constructed by an analysis system for analyzing human gesture and an interact system for interacting with agents in the virtual space using analyzed information. An implemented analysis system for analyzing gesture extracts a head and hands region after taking image sequence of an operator's continuous behavior using CCD cameras. In interact system, we construct the virtual space that exist an avatar which incarnating operator himself, an autonomous object (like a puppy), and non-autonomous objects which are table, door, window and object. Recognized gesture is transmitted to the avatar in the virtual space, then transit to next state based on state transition diagram. State transition diagram is represented in a graph in which each state represented as node and connect with link. In the virtual space, the agent link an avatar can open and close a window and a door, grab or move an object like a ball, order a puppy to do and respond to the puppy's behavior as does the puppy.

**Key Words** : HCI(Human-Computer Interaction), gesture recognition, state transition diagram

\* 平生會員, 蔚山大學校 制御計測工學科  
(Dept of Control and Instrumentation Eng., University of Ulsan)

\*\* 正會員, ETRI 假想現實研究센터 연구원  
(Virtual Reality Center, ETRI)

\*\*\* 正會員, ETRI 컴퓨터소프트웨어研究所  
(Computer Software Laboratory, ETRI)

※ 본 연구논문은 정보통신우수시범사업의 연구지원을 받았으며, 이에 감사드립니다.

接受日字:2001年5月28日, 수정완료일:2002年4月11日

## I. 서론

인간의 의사 전달 표현은 지역과 문화에 따라 차이가 있지만, 서로간의 통신 수단에서 아주 중요한 역할을 한다. 인간의 의사 전달에 관한 연구는 오랜 시간동안 연구되어 왔고 컴퓨터의 발전과 더불어 최근에는 컴퓨터와의 의사 전달을 위해 인간의 행동을 인식하고자 하는 연구가 많이 진행되고 있다.<sup>[1~35,11,16]</sup> 논문에서는 행동에 대한 본질적인 의미를 기계에게 전달함으로써 동작자가 친밀감을 느낄 수 있는 지능적인 시스템을 구현하고자 한다. 그리고 인간과 기계와의 상호 작용을 위해 간단한 가상 공간을 구현한다. 가상 공간에는 아바타(avatar)로 불리는 인간의 동작을 직접적으로 수행하는 에이전트(agent)와 자율적으로 움직이는 퍼피(puppy)가 존재한다. [16]은 자율적 객체와 아바타간의 상호 작용에 대한 좋은 예가 된다. 본 연구에서는 동작자가 마치 실제 세계에서 행동하듯이 자연스러운 동작으로 인식 시스템에게 명령을 내려 다양한 동작을 인식시키고, 또한 인식된 동작을 이용하여 가상 공간에 존재하는 에이전트들과 상호 작용이 가능하도록 한다.

이러한 실세계에 가까운 시스템을 구현하기 위해 고려되어야 할 것은 크게 인간 중심의 인터페이스의 구현과 현실감이 느껴지는 가상 공간의 구현으로 나눌 수 있다. 실세계와의 상호 작용을 위해 구현된 가상 공간이 현실감을 가지기 위해서는 그 공간에 존재하는 에이전트들이 실세계에 존재하는 것들과 최대한 유사한 형태를 가질 필요가 있다. 특히 인간과 대화가 이루어지는 에이전트의 얼굴 표정은 현실감을 결정짓는 중요한 변수이다. 마치 실제 인물과 대화하듯이, 가상 공간내의 에이전트는 주기적으로 눈을 깜박이거나 눈썹을 움직이고, 얼굴 근육의 움직임도 자연스럽게 이루어져야 한다. 현실감이 느껴지는 얼굴 표정을 얻기 위해, 그물 구조를 가지는 얼굴 모델을 만든 후, 얼굴 근육의 동작에 대한 변형이나 움직임을 제어하는 방법<sup>[17,18]</sup>이나 레이저 스캐너를 이용해 얼굴 근육의 움직임에 대한 3차원 데이터를 얻는 방법<sup>[9]</sup>이 많이 사용된다. 본 연구에서는 인간 중심의 인터페이스 구현에 비중을 둔다. 인간은 일반적으로 손, 다리, 얼굴 표정, 목소리, 그리고 머리의 방향을 이용해 의사 전달을 한다. 머리의 방향에 따른 시선의 움직임을 해석하기 위한 도구로서 HMD(Head Mounted Display)가 일반적으로 사용이 된

다. 또한, 이 장비는 실세계와 가상 공간을 결합시켜 몰입감을 최대화 시켜주는 장점이 있다. 손의 동작은 다양성 때문에 그 자체로써 하나의 언어(수화)가 될 수 있다.<sup>[23]</sup> 지금까지 손의 동작 인식을 위한 연구는 Data Glove에 의해 많이 이루어졌다. 그러나, HMD나 Data Glove와 같은 장비는 정밀도는 뛰어나지만, 항상 착용해야 하고 또한, 고가이기 때문에 인간 행동의 흐름을 해석하기 위한 장치로는 불편하다. 행동의 흐름을 해석하는 것은 영화 또는 애니메이션 분야에서 사용되는 모션캡처(motion capture) 시스템과 달리 아주 뛰어난 정밀도를 요구하지 않는다. 그러므로, 앞서 발표되었던 [21]에서는 CCD카메라를 이용해 아무런 장비를 부착하지 않은 동작자의 동작을 해석하였다. 여기서는 이 시스템을 이용해 얻은 결과를 가지고 동작자의 동작을 인식하고, 에이전트와 상호 작용할 수 있는 시스템을 구현하고자 한다.

인식된 동작은 현재의 에이전트의 상태에 따라 선택적으로 수행된다. 예를 들어, 동작자가 가상 공간내의 문을 열고자 할 경우, 동작자의 동작을 직접적으로 수행하는 아바타는 문 앞에 있어야 한다. 그렇지 않으면 명령은 무시된다. 가상 공간 속의 자율적인 객체인 퍼피는 스스로 동작자에게 무엇을 요구할 수 있다. 그리고 동작자는 퍼피의 요구에 대한 긍정 또는 부정의 답을 줄 수가 있다.

얼굴의 방향과 손 동작에 대한 명령을 인식하기 위해 상태 천이도를 이용한다.<sup>[11]</sup> 동작자의 모든 동작은 반드시 이전의 동작을 기반으로 한 연속된 움직임으로 이루어진다. 이러한 동작의 흐름은 그래프 형태를 가지는 상태 천이도로 표현할 수 있다.

## II. 얼굴 방향과 손 동작 해석

이 장에서는 동작의 인식을 위해 동작자의 얼굴 영역과 손 영역을 해석하는 법에 대해 간략히 설명하고자 한다. 이 내용은 [21]에 보다 자세히 기술되어 있다. 동작자의 동작은 그림 9(a)와 같이 동작자의 전방과 전방 상단에 설치된 두 대의 CCD카메라에 의해 입력된다. 동작은 동작자의 머리와 손의 움직임에 의해 이루어지므로 먼저, 동작자의 머리와 손의 영역을 추출한다. 입력 영상에서 피부색 영역만을 취함으로써 머리와 손의 영역을 찾는다. 피부색은 칼라 영상의 RGB분포를 실험적으로 계산함으로써 얻을 수 있다.

피부 영역이 추출되면, 이 영역으로부터 인식에 사용될 특징을 추출하기 위해 각 영역의 기하학적인 정보를 계산한다. 각각의 피부 영역들은 근사적인 타원의 형태로 되어 있다고 가정하고 타원의 방정식을 이용해 모델링한다. 그 결과 각 영역들의 중점과 타원의 장축이 수평축과 이루는 기울기 정보를 얻을 수 있다. 영상으로부터 추출된 각 피부 영역들의 모델링 작업이 끝나면, 이 정보를 이용해 머리와 손의 움직임을 해석할 수 있다.

각 영역들에 대해 머리와 두 손의 구분은 다음과 같이 이루어진다. 동작이 시작되기 전의 초기 자세로부터 머리와 두 손의 첫 시작 위치를 초기 설정한다. 동작자의 초기 자세는 의자에 앉은 상태에서 두 손을 무릎에 올린 자세이다. 각각의 입력된 영상에서 피부색 영역으로 검출된 영역들 중에서, 머리는 영상의 상단에 존재하는 것으로 설정하고, 영상의 하단에 존재하는 두 영역 중 왼쪽에 존재하는 영역을 오른손, 오른쪽에 존재하는 영역을 왼손으로 설정한다. 이렇게 초기 자세에서 각 영역이 결정되면, 이후의 시간부터 입력되는 영상은 칼만 필터(Kalman filter)를 이용해 각 영역의 이동을 추적하게 된다.<sup>[6,7,14,15]</sup>

## 1. 얼굴 영역의 해석

### 1) 얼굴의 특징점 추출 및 신뢰도 검사

머리로 간주된 영역으로부터 얼굴의 방향 정보를 얻는다. [21]에서는 하나의 2차원 영상으로부터 3차원 방향을 추정할 수 있는 방법을 제안하였다. 머리는 3차원 좌표를 가지는 타원체로 가정하고 타원체의 표면에 얼굴의 특징인 두 눈과 코, 입이 존재한다고 가정한다. 얼굴 특징점들의 중점을 알 수 있을 경우 타원체의 중심과 얼굴 특징점들의 중심이 이루는 각이 얼굴의 방향이 된다. 머리의 회전 방향은 동작자가 바라보는 방향을 기준으로 수평축에 대해  $\alpha$ , 수직축에 대해  $\beta$ , 시선의 방향축에 대해  $\gamma$ 로 설정하였다. 얼굴 특징점의 위치를 얻기 위해 템플릿 상관법(template matching)을 이용한다.

하나의 템플릿에 대해 얼굴의 영역을 검사하면, 그 템플릿과 가장 유사한 얼굴의 영역에서 가장 큰 상관값을 가지며 그 값을 중심으로 점점 일치하는 정도가 감소해서 전혀 일치하지 않을 경우, 아주 낮은 상관값을 가지는 분포가 만들어진다. 그 중 가장 크게 일치하는 부분이 찾고자 하는 얼굴 특징점의 위치이다. 그러

나, 찾아진 특징점들은 가장 일치하는 곳에서 최대의 상관값을 가지지만 그 인근의 점들도 특징점으로 인정할 수 있는 큰 상관값을 가지므로 현재 찾아진 특징점이 얼마만큼 유효한 특징점으로 인정할 수 있는가에 대한 신뢰도를 검사할 필요가 있다.

신뢰도를 검사하기 위해, 템플릿 상관에 의한 상관값의 분포에서 최대 상관값을 가지는 점과 상관값이 감소하기 시작해 최소값을 가지는 점들을 포함하는 영역을 그림 1(a)와 같이 얻는다. 그리고, 이 상관값의 분포는 근사적으로 정규 분포를 가진다고 가정하고 그 분포에서 특징점으로 인정할 수 있는 상관값의 범위를 선택한다(그림 1(b)). 이 영역의 값을 2차원 평면 상에 투영시키면 그림 1(c)와 같이 근사적으로 타원의 형태로 나타나게 된다. 이 영역을 타원 방정식을 이용해서 모델링하면 타원의 중점과 장단축의 비율, 장축이 수평에 대해 기울어진 각, 찾아진 특징점과 타원의 중점 사이의 거리를 얻을 수 있다.

신뢰도는 타원 방정식의 모델링에서 얻은 값을 비교해서 구할 수 있다. 얼굴의 각 특징들은 너비가 높고 긴 형태로 구성되어 있기 때문에 2차원 평면상에 투영된 타원의 장축의 각이 수평을 기준으로  $0^\circ$ 에 가까울수록 '신뢰도는 큰 값을 가진다. 상관값이 정규 분포로 되어있다고 가정하였으므로 상관값의 가장 높은 값을 가지는 특징점의 위치와 타원의 중점과의 거리가 가까울수록 신뢰도는 큰 값을 가진다. 또한, 타원의 장

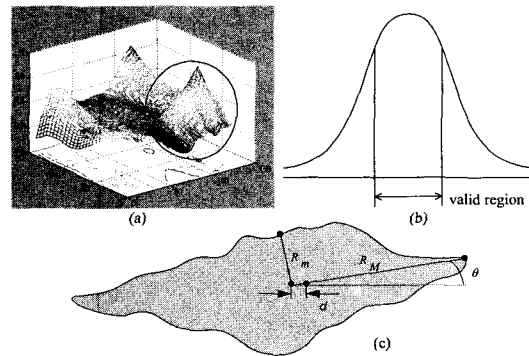


그림 1. 얼굴 특징점의 신뢰도 추정: (a) 템플릿 상관 후의 코의 상관 분포, (b) 영역의 최대값에 대한 히스토그램, (c) 유효 영역의 평면도

Fig. 1. Credibility estimation of feature point: (a) matching distribution of nose after template matching, (b) histogram of maximum value region in (a), (c) projection of valid region.

축과 단축의 거리가 가까울수록 타원의 중점은 안정된 위치에 놓이게 되므로 신뢰도는 큰 값을 가진다. 이러한 조건을 만족하는 신뢰도는 식 (1)과 같이 정규식으로 표현할 수 있다.

$$C(R_M, R_m, \theta, d) = \frac{1}{2} \cdot \frac{\frac{R_m}{R_M} + |\cos \theta|}{\frac{d}{R_m} + 1} \quad (1)$$

여기서  $R_M$ 는 장축의 길이,  $R_m$ 는 단축의 길이,  $\theta$ 는 수평축에 대해 장축이 기울어진 각 그리고  $d$ 는 특징점과 타원의 중점사이의 거리를 가리킨다.

식 (1)에 의해, 신뢰도는 찾아진 특징점과 타원의 중점이 일치하고, 타원의 중점이 수평으로 놓여져 있고, 장축과 단축의 거리가 같을 경우 최대값인 1의 값을 가진다. 그렇지 않을 경우, 조건에 따라 0과 1사이의 값을 가진다.

2) 얼굴의 방향 해석

타원체의 표면에 존재하는 얼굴 특징점들은 그림 2(a)와 같이 특징점의 위치와 신뢰도를 함께 표현할 수 있다. 그림 2(a)에서  $x$ 와  $y$ 축은 입력 영상의 좌표축을 가리키고, 점  $p1, p2, p3, p4$ 는 각각 오른쪽 눈, 왼쪽 눈, 코, 입의 위치를 가리킨다. 그리고, 이 점들은 각각  $C_R, C_L, C_N, C_M$ 으로 기술된 신뢰도를 포함한다. 특징점들의 중점은 그림 2(b)와 같이 각 축에 대한 신뢰도 그래프로 바꾼 후 1차 모멘트 식을 이용해 구한다. 이 과정은 식 (2)와 같으며, 이때  $A$ 는 영역의 넓이다.

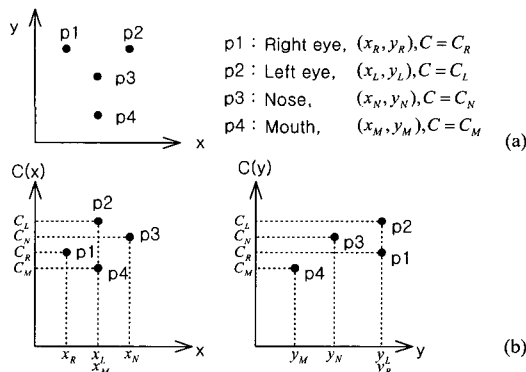


그림 2. 특징점의 추출: (a) 신뢰도가 포함된 특징점의 위치, (b) 각 축에 대한 신뢰도 그래프

Fig. 2. Extraction of feature points with credibility: (a) position of points with credibility in image axis, (b) credibility graph of each axis.

머리 회전각의 계산은 식 (3)과 같다. 그림 3과 식 (3)에서,  $R_s$ 와  $R_t$ 는 보는 관점에 따른 타원체의 반지름,  $(x_c, y_c, z_c)$ 는 타원체의 중점,  $(x_o, y_o)$ 는 특징점들의 중점,  $(d_x, d_y)$ 는  $(|x_c - x_o|, |y_c - y_o|)$ ,  $d_{x-eye}$ 는  $x$ 축 방향의 두 눈 사이의 거리,  $d_{eye-eye}$ 는 두 눈 사이의 거리를 가리킨다. 타원체를 옆에서 보았을 경우 수평축을 기준으로 하는  $\alpha$ 를 구할 수 있다. 이때  $R_s$ 는 타원체의 수직축의 반지름이다. 타원체를 위에서 보았을 경우, 수직축을 기준으로 하는  $\beta$ 를 구할 수 있다. 이때  $R_t$ 는 타원체의 수평축의 반지름이다.  $\gamma$ 의 경우 두 눈 사이의 기울기를 계산함으로써 구할 수 있다.

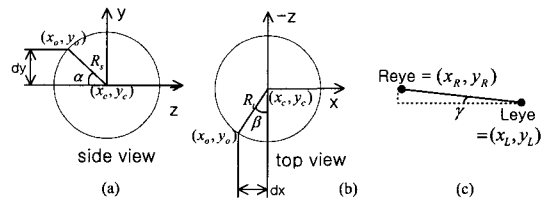


그림 3. 머리 모델에 의한 얼굴의 방향 추정: (a)  $\alpha$ , (b)  $\beta$ , (c)  $\gamma$ .

Fig. 3. Estimation of face direction by head model: (a)  $\alpha$  based on horizontal axis, (b)  $\beta$  based on vertical axis, (c)  $\gamma$  base on optical axis.

$$x_o = \frac{\sum_i x_i c(x_i)}{A_x}, \quad y_o = \frac{\sum_i y_i c(y_i)}{A_y}, \quad \text{where } i = R, L, N, M \quad (2)$$

$$\alpha = \sin^{-1} \left( \frac{d_y}{R_s} \right), \quad \beta = \sin^{-1} \left( \frac{d_x}{R_t} \right), \quad \gamma = \sin^{-1} \left( \frac{\|d_{x-eye}\|}{\|d_{eye-eye}\|} \right) \quad (3)$$

2. 손 영역의 해석

전처리 단계에서 얻어진 손의 영역은 손의 정보뿐만 아니라 팔의 정보도 포함될 수 있다. 이 경우, 손목을 찾아 손과 팔의 영역을 구분한다. 손목은 기하학적 특성상 팔 너비가 감소에서 증가하는 부분에 존재한다. 손목이 찾아지면 손목에서부터 손끝까지의 영역을 팔의 영역이 제외된 순수한 손의 영역으로 간주한다.

1) 손의 특징 해석

팔의 영역이 포함되지 않은 손의 영역을 해석하기 위해 손의 영역을 근사적인 타원으로 간주하고 타원

방정식을 이용해서 모델링하고, 또한 손의 위치, 크기, 형상 정보를 추출한다. 손은 펼쳐진 손바닥, 모어진 손바닥, 주먹, 엄지 손가락에 의한 지시, 집게 손가락에 의한 지시와 같은 5가지 형태를 구분한다.

영역의 중점에서 손 영역의 모든 경계 점과의 거리의 평균을 반지름으로 하는 가상의 원을 손 영역의 중심을 기준으로 그릴 경우, 이 원의 외부에 나타나는 손 영역을 해석함으로써 손가락을 추출할 수 있다. 손가락은 원의 외부에 길고 좁은 형태로 나타나게 되고 손가락이 아닌 영역은 짧고 넓은 형태로 나타난다. 두 개 이상 붙어 있는 손가락은 손 영역의 중점에서 경계점 간의 가장 긴 거리와 유사한 길이를 가지는 형태로 나타난다.

그림 4에 손가락을 검출하는 과정을 보인다. 손목선의 두 끝점을 손 영역의 경계점을 따라 확장하여 원의 둘레와 처음 만나는 두 점을 손가락 검사의 시작점과 끝점으로 설정한다. 시작점에서부터 끝점 사이에 포함된 손 영역의 경계점을 검사하여 원의 외부에 있는 영역 중 영역의 길이에 대한 너비의 비가 실험적으로 0.5보다 작은 값을 가지는 긴 형태로 나타날 경우 손가락으로 간주한다. 손가락이 손의 진행 방향을 가리키는 축의 끝점을 포함하면 그 손가락은 지시의 의미를 가지는 집게 손가락으로 간주한다.

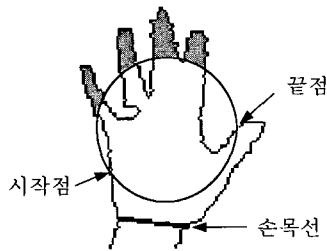


그림 4. 손가락 추출과정  
Fig. 4. Finger extraction.

표 1은 찾아진 특성을 비교하여 손의 형상을 결정하는 조건을 나타낸다. 표 1에서 'X'는 don't care를 가리킨다. 예를 들어 펼쳐진 손바닥이 되기 위한 조건은 첫째, 타원의 장축에 대한 단축의 비율은 어떠한 값이라도 상관없다. 둘째, 손의 중점으로부터 경계점까지의 최대 거리에 대한 최소 거리의 비가  $Th_{HandRatio}$ 보다 작아야 한다. 셋째, 엄지 손가락의 검출 여부는 상관하지 않는다. 넷째, 집게 손가락의 수와 나머지 손가락의

수가 각각 1개와 1개 이상이거나 또는 집게 손가락은 검출되지 않고 나머지 손가락이 2개 이상 검출되어야 한다. 다섯째, 손의 진행 방향은 장축이든지 단축이든지 상관하지 않는다. 위의 다섯 가지 조건이 만족되면 손의 형상은 펼쳐진 손바닥으로 인식된다. 실험적으로 구한  $Th_{HandRatio}$ 와  $Th_{EllipseRatio}$ 의 값은 [21]에 기술되어 있다. 손가락이 검출되면 손가락의 끝점과 손 영역이 원의 둘레와 만나는 두 교점의 중점을 이용해서 손가락이 가리키는 방향을 얻을 수 있다.

표 1. 손의 형태 추정 조건 (X : don't care)  
Table 1. Estimate condition of hand shape (X : don't care)

	ellipse ratio	hand length ratio	number of thumb	number of index finger	number of other finger	hand direction
palr. (opened finger)	X	< $Th_{HandRatio}$	X	1 0	> 0 2	X
palr. (closed finger)	X $Th_{EllipseRatio}$	< $Th_{HandRatio}$	X	0 0	1 0	X
fist	$Th_{EllipseRatio}$	$Th_{HandRatio}$	0	0	0	X
point (thumb)	< $Th_{EllipseRatio}$	< $Th_{HandRatio}$	1	0	0	minor axis
point (index finger)	X	< $Th_{HandRatio}$	X	1	0	X

3. 머리와 손의 움직임 추적

손은 공간상에서 자유롭게 움직이기 때문에, 동작 중에는 오른손과 왼손을 구별해서 추적해야 한다. 또한, 머리와 두 손의 영역에서 겹침이 발생할 경우에도, 각 영역의 움직임은 구별되어야 한다. 이러한 문제를 해결하기 위해 칼만 필터를 이용한다.

머리나 손의 3차원 움직임은 6개의 자유도 - 3개의 이동(translation) 요소와 3개의 회전(rotation) 요소 - 를 가진다. CCD카메라로부터 입력된 영상에서는 이동에 관한 요소는 쉽게 얻을 수 있지만, 회전에 관한 요소는 얻기가 힘들다. 그래서, 본 연구에서는 머리와 손의 3차원 움직임을 추적하기 위해 이동 요소만을 고려한다.

외부의 추가적인 힘이 작용하지 않는다면 이전 프레임과 현재 프레임간의 영역의 이동 속도는 변함없다고 가정을 하고, 아래와 같이 각 영역의 운동을 위치 이동에 대한 상태 벡터(state vector)로 표현하면 위치  $p = [px, py, pz]^T$ 와 속도  $v = [vx, vy, vz]^T$ 와 같다. 상태 벡터의 시간에 따른 변화를 모델링한 역학적 모델에

따라 상태 벡터를 예측하면 다음과 같다.

$$\hat{s}_i^- = f(\hat{s}_{i-1}^+) \quad (4)$$

여기서,  $s_i^-$  는 이전 상태 벡터(priori state vector),  $s_i^+$  는 다음 상태 벡터(posteriori state vector), 그리고  $f(.)$ 는 등속도 운동일 때를 가정한 상태 전이 행렬이다.

각 영역의 위치 이동을 표현하는 상태 벡터는 식 (5)와 같다.

$$s = [b^T \ v^T]^T \quad (5)$$

예측된 이전 상태 벡터의 역학적 모델은

$$\hat{s}_i^- = F \cdot \hat{s}_{i-1}^+ \quad (6)$$

와 같고, 상태 전이 행렬은

$$F = \begin{bmatrix} I_3 & I_e \\ 0_3 & I_3 \end{bmatrix} \quad (7)$$

여기서  $I_3$ 는  $3 \times 3$  단위 행렬,  $0_3$ 는  $3 \times 3$  영행렬이다. 즉, 바로 이전 시간  $t-1$ 에서 얻은 이동 속도가 등속도로 유지된 값이 현재 시간에서의 예측치가 된다.

예측치를 포함하거나 영역의 이동 방향으로 예측치와 가장 인접한 곳에 위치한 영역이 예측치가 가리키는 영역이 된다. 머리와 두 손에 대한 각 영역이 예측치에 의해 결정되면, 초기 자세 해석에서와 같이 각 영역의 특징을 계산하기 위해 타원 방정식을 이용해 각 영역을 모델링한다. 각 영역의 실제 이동된 값은 영상으로부터 손 영역의 중점을 구하는 방법으로 얻을 수 있다.

각 영역의 동작의 경신은 실제 이동된 값과 상태 벡터간의 관계를 기술한 관측(measurement) 모델에 의해 이루어진다. 관측 모델은

$$z_i = h(s_i) + \eta_i \quad (8)$$

로 표기한다. 이 식에서 함수  $h(.)$ 는 관측 함수이며,  $\eta_i$ 는  $\eta_i \sim N(0, R_i)$ 의 특성을 갖는 정규 랜덤 노이즈이다.

관측치의 예측은

$$\hat{z}_i = H \cdot \hat{s}_i^- \quad (9)$$

로 얻으며 관측 행렬  $H$ 는 식 (10)과 같이 간단한 형태

가 된다.

$$H = [I_e \ 0_3] \quad (10)$$

관측 오차(measurement residual)는 관측 모형에 의해 예측된 관측치와 실제 이동된 값과의 차이이다. 관측 오차에 칼만 이득(Kalman gain)행렬을 가중치로 반영한 수정(innovation) 항을 예측치에 더해 새로운 상태 벡터로 갱신한다.

$$s_i^{(+)} = \hat{s}_i^{(-)} + K_i(z_i - H \cdot \hat{s}_i^{(-)}) \quad (11)$$

위 식에서 칼만 이득은

$$K_i = P_i^{(-)} H_i^T [H_i P_i^{(-)} H_i + R_i]^{-1} \quad (12)$$

와 같다.  $P^{(-)}$ 는 예측 오차 공분산(a priori estimate error covariance) 행렬,  $R$ 은 관측 잡음 공분산(measurement noise covariance) 행렬이다.

머리와 두 손의 각 영역에 대해 칼만 필터를 적용할 경우, 상태 벡터  $s_i^{(+)}$ 에 의해 시간에 따른 위치 이동을 추적할 수 있고, 각 영역간에 겹침이 발생했을 경우에도 영역들의 이동 방향과 예측치를 알기 때문에 추적이 가능하다. 특정 시간  $t$ 에 대해, 추출된 피부색 영역의 전처리 과정이 끝나면 머리 영역에 대해 얼굴의 특징점을 찾은 후 얼굴의 방향을 구한다. 그리고, 손 영역에 대해 손의 크기, 위치, 형상 정보를 해석한다.

#### 4. 특징 추출에 대한 인식률

입력된 영상의 특징 추출을 위해 첫번째 과정으로 전처리 과정을 거쳤다. 전처리 과정에서 동작자의 피부 영역을 찾고, 이 영역으로부터 머리 영역과 손 영역을 구별한 뒤, 찾아진 각 영역을 타원의 방정식을 이용해 모델링한 후, 머리 영역에서 얼굴의 방향 정보를, 손 영역에서 손의 특징을 찾았다.

손이나 팔의 움직임에 의해 얼굴 특징점들의 일부가 보이지 않을 경우, 보이지 않는 특징점은 이전의 영상에서 추정된 예측치를 이용하여 그 점의 위치를 결정하였다. 두 눈과 코, 입에 해당하는 4개의 특징점들 중에서 2개 이상이 보이지 않을 경우, 얼굴의 방향은 이전 영상에서 추출된 얼굴의 방향과 동일하다고 간주하였다. 그리고, 1개의 특징점만 보이지 않을 경우, 이전 영상과 비교하여 나머지 특징점들의 이동 변위의 평균 값을 얻고, 이 값을 이용하여 보이지 않는 특징점의 위

치를 추정된 뒤, 얼굴의 방향을 계산하였다. 그 결과, 얼굴 방향의 오차는  $\pm 3^\circ$  내의 값을 가졌다.

손의 형상 정보는 표 1과 같이 많은 요소들이 복합적으로 비교되어 결정된다. 형상 정보에 대한 인식률을 계산하기 위해 전방과 전방 상단에 설치된 두 대의 카메라로부터 연속된 동작에 대한 각각 400개의 영상을 입력 받은 뒤, 실험을 하였다. 주먹과 모어진 손바닥 형태의 경우 손의 형태가 단순하기 때문에 98% 이상의 인식률을 보였다. 펼쳐진 손바닥과 지시형태에서는 손가락의 추출 여부에 따라 인식이 결정되었다. 집게 손가락은 타원의 장축의 끝에 존재하기 때문에 98% 이상의 높은 인식률을 보였다. 엄지 손가락은 뻗어나가는 방향이 다른 손가락들과 틀리기 때문에 손의 방향이나 여러 형태에 따라 엄지 손가락이 영상 내에서 보이지 않는 경우와 그림 4와 같이 엄지 손가락의 영역이 가상의 원과 만나지 않는 경우가 종종 발생하였다. 엄지 손가락은 91%의 인식률을 보였다.

그림 8(a)와 10(a)는 추출된 얼굴의 방향과 손 동작의 결과를 보여준다. 각각의 그림에서 위쪽 영상은 전방 상단에 설치된 카메라에서, 아래쪽 영상은 전방에 설치된 카메라에서 입력된 결과이다. 그림 8(a)에서 동작자의 얼굴 방향 중 수평 회전 값인  $\beta$ 는 오른쪽으로  $25.9^\circ$ , 수직 회전 값인  $\beta$ 는  $19.3^\circ$ 로 추출됨을 알 수 있다. 손의 크기 정보는 손 영역의 픽셀의 수를 계산함으로써 구할 수 있다. 이 크기는 손과 카메라의 거리에 따라 같은 형상을 하고 있는 손이라 하더라도 크기가 다르게 나타난다. 크기 정보는 연속된 동작에서 손의 이동이나 형상의 변화가 있을 때 나타나는 상대적인 크기의 변화를 검사하는 데 사용된다. 그림 8(a)에서 손의 형태는 오른손은 손바닥, 왼손은 주먹으로 추출됨을 알 수 있다. 그림 10(a)는 손의 영역에 손 정보뿐 아니라 팔의 정보까지 포함된 경우를 보여준다. 이 경우는 손목선 추출 알고리즘을 이용해 손과 팔의 영역으로 구분한 뒤 손의 정보를 추출하게 되는데, 그 결과 오른손은 지시 형태, 왼손은 손바닥 형태임을 추출할 수 있었다.

### III. 인 식

#### 1. 동작 인식을 위한 컨텍스트 기반 접근법

제안하는 휴먼 인터페이스(HI) 시스템은 동작자의 동작 영상을 처리하고, 이들을 해석한 후, 가상 공간과 상

호 작용하도록 하는 기능을 한다. HI 시스템으로 입력되는 동작자의 동작은 머리를 좌우상하로 움직이고, 팔을 여러 방향으로 움직이고, 손을 폈다, 쥐었다 하는 것들이다. 이러한 동작들은 긍정, 부정에 대한 의미, 누군가를 부르거나, 쓰다듬어 주고, 명령을 내리고, 물건을 들고, 움직이는 의미로서 최종적으로 가상 공간에 전달된다.

이러한 동작들은 의미를 가지는 행동들의 시퀀스로 구성된다. 본 연구에서는 행동의 시퀀스 중 의미있는 각 부분들을 하나의 상태(state)로 간주한다. 각각의 상태들은 행동의 의미에 따라 서로간에 연결되어 있고, 각 상태는 다른 상태 또는 자신의 상태로 전이될 수 있다. 즉, 인간의 행동은 상황에 따라 고려되는 상태들의 흐름으로 표현 가능하다. 각 상태는 종속적으로 연결되어 있기 때문에 현재의 상태에서 다음 상태로의 추정이 가능하고 정의되지 않은 동작에 대해 시스템의 안정성을 유지할 수 있다.

또한, 각 상태의 천이시 필요한 행동은 숙련된 동작자의 동작을 데이터 베이스화시킨 표본 동작을 바탕으로 한다. 동작 인식을 위해 일반적으로 많이 사용되어 온, 시간의 흐름에 따라 순차적으로 동작을 해석해 나가는 방법([20])의 경우, 앞부분의 동작이 인식되지 않으면 뒷부분의 동작 인식에도 영향을 미치게 된다. 제시하는 인식법은 각 상태마다 천이가 일어날 때는 동작자의 입력 동작을 표본 동작과 비교함으로써 움직임이 조금 틀리거나, 동작 데이터의 일부가 유실되는 경우에도 인식이 가능하다.

#### 1) 상태 천이도

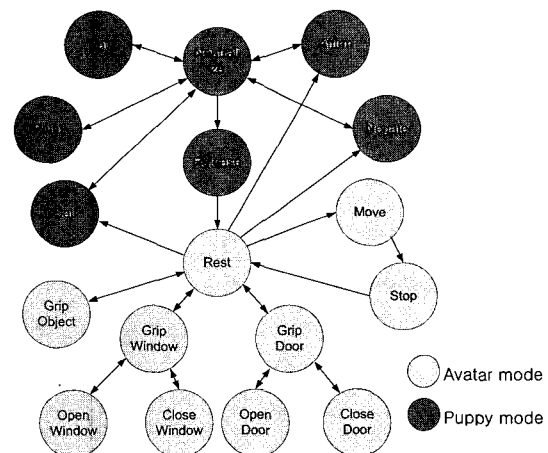


그림 5. 상태 천이도

Fig. 5. State transition diagram.

본 연구에서 사용된 상태 천이도를 그림 5에 보인다.

이것은 의미없는 동작의 상태를 가리키는 "Rest"를 기준으로 아바타 모드와 퍼피 모드로 나누어진다. 아바타 모드는 아바타의 직접적인 행동들에 대한 상태들의 집합이다. 그러므로, 아바타 모드에서는 아바타 자신이 직접 이동을 하여 공과 같은 오브젝트를 잡고, 그 오브젝트를 이동시킬 수 있으며, 가상의 공간에 존재하는 창문이나 문을 열거나 닫을 수 있다. 퍼피 모드는 가상 공간에 존재하는 퍼피와 상호 작용할 수 있는 행동들에 대한 상태들의 집합이다. 아바타 모드에서 퍼피 모드로의 전환은 아바타가 퍼피를 부름으로써 가능해진다. 아바타가 퍼피를 부를 경우, 자율적으로 움직이던 퍼피의 동작 제어권은 아바타에게 넘어간다. 퍼피 모드에서 아바타는 퍼피에게 오브젝트를 가져오도록 명령을 내리거나, 퍼피를 쓰다듬어 주는 행동, 또한 퍼피의 행동에 대해 긍정이나 부정의 답을 할 수 있다. 퍼피 모드에서 아바타 모드로의 복귀는 퍼피를 자유롭게 놓아준다는 의미인 "Release"동작을 취함으로써 이루어진다.

상태 천이도에서 두 손과 머리의 동작은 독립된 상태를 가진다. 즉, 두 손은 각각 독립된 상태를 가질 수 있기 때문에, 한손은 퍼피 모드, 다른 손은 아바타 모드에 존재할 수 있다. 그러므로, 한 손으로는 퍼피에게 명령을 내리고, 다른 한 손 또는 머리를 이용해서 아바타가 직접 어떠한 동작을 취하게 할 수 있다. 그러나, 두 손과 머리는 독립적으로 동작을 하기 때문에 각 상태가 인정 가능 하더라도 동시에 발생할 수 없는 상반된 동작이 발생할 수 있다. 상태의 천이가 발생하기 전에 각 상태를 검사하고 비교함으로써 이런 상황을 예방한다.

각 상태의 구체적인 행동 표현을 표 2에 보인다. 상태 천이도의 각 상태의 의미를 기술하면 다음과 같다. "Rest"는 의미없는 동작 상태 또는 연관성이 없는 동작을 연결하는 상태를 가리키며 아바타 모드와 퍼피 모드를 구별하는 상태이다. 의미를 가지지 않는 동작자의 움직임 또는 정의되지 않은 움직임들이 여기에 속한다. 아바타 모드에서, "Move"는 아바타를 이동시키고, "Stop"은 아바타의 이동을 멈추기 위해 "Move"다음에 연결된다. 이 명령이 인정되면 명령 수행 후 자동으로 "Rest"로 천이된다. "Grip Object"는 오브젝트를 잡는 명령이다. 아바타는 이 동작을 이용해 오브젝트를 잡은 뒤, "Move"명령으로 직접 가상 공간을 이동하여, 오브젝트를 옮기는 복합 동작이 가능하다. 창문이나 문을

"Grip Window", "Grip Door"로 손잡이를 잡은 후, "Open/Close Window", "Open/Close Door"를 이용해 열거나 닫을 수 있다. 퍼피 모드에서, "Call"은 퍼피를 부르는 상태이다. 퍼피에게 명령을 내리기 위해 반드시 이 명령이 요구된다. 명령이 인식되면 명령을 취한 손의 상태는 퍼피 모드로 천이된다. "Point"는 아바타가 가리키는 장소로 퍼피를 이동시키거나 가상 공간 내에 존재하는 오브젝트를 가져오도록 명령을 내리는 상태이다. "Pat"는 퍼피의 행동에 대한 응답으로 퍼피를 쓰다듬어 주는 행동을 표현하는 상태이다. "Affirm"과

표 2. 각 상태의 구체적 행동  
Table 2. Concrete actions of each state.

모드	상태(명령)	의미
Avatar	Rest	의미 없는 동작 상태 또는 연관성이 없는 동작을 연결하는 상태를 가리킨다. Avatar모드와 Puppy모드를 구별하는 상태이다.
	Move	avatar를 이동시킨다.
	Stop	avatar의 이동을 멈춘다. 이 상태는 Move명령 다음에 연결된다. 이 명령이 인정되면 명령 수행 후 자동으로 Rest상태로 천이된다.
	Grip Object	가상 공간 내의 물건을 잡는다.
	Grip Window	가상 공간 내의 창문 손잡이를 잡는다. 창문을 열거나 닫기 위해 필요한 상태이다.
	Grip Door	가상 공간 내의 문 손잡이를 잡는다. 문을 열거나 닫기 위해 필요한 상태이다.
	Open Window	창문을 연다. 반드시 이 동작은 창문 손잡이를 잡은 후에 이루어진다. 이 동작이 인식되면 천이 되었다가 동작이 멈추면 다시 Grip Window 상태로 자동으로 천이 된다.
	Close Window	창문을 닫는다. Open Window와 동일한 조건을 가진다.
	Open Door	문을 연다. Open Window와 동일한 조건을 가진다.
	Close Door	문을 닫는다. Open Window와 동일한 조건을 가진다.
Puppy	Call	Puppy를 부른다. Puppy에게 명령을 내리기 위해 반드시 이 명령이 요구된다. 명령이 인식되면 명령을 취한 동작자의 손의 상태는 puppy모드로 천이 된다.
	Point	Puppy에게 명령을 내린다. 동작자가 가리키는 장소로 puppy를 이동시키거나 가상 공간 내에 존재하는 물건을 가져오도록 명령을 내린다.
	Pat	Puppy의 행동에 대한 응답으로 puppy를 쓰다듬어 주는 행동에 대한 명령이다.
	Affirm	Puppy가 현재 수행 중인 명령에 대해 긍정의 답을 한다.
	Negate	Puppy가 현재 수행 중인 명령에 대해 부정의 답을 한다.
	Neutralize	Puppy모드에 속한 상태들을 연결하여 주는 상태이다. Puppy모드의 모든 명령은 인식이 된 후 명령에 대한 처리가 끝나면 자동으로 Neutralize상태로 천이한다.
Release	Puppy모드에서 avatar모드로 돌아온다.	



"Negate"는 퍼피가 현재 수행중인 명령에 대해 긍정, 부정의 응답을 표현하는 상태이다. "Neutralize"는 퍼피 모드에 속한 상태들을 연결하는 주는 상태이다. 아바타 모드의 "Rest"와 같이 의미없는 동작이나 이전 동작과 연관성이 없는 동작은 "Neutralize"로 친이된다. 또한, 퍼피 모드의 모든 명령은 인식이 된 후 명령에 대한 처리가 끝나면 자동으로 "Neutralize"로 친이한다. "Release"는 퍼피 모드에서 아바타 모드로 돌아오는 명령을 나타내는 상태이다.

## 2) 입력 동작과 표본 동작과 비교

시간의 흐름에 따라 입력되는 동작자의 동작은, 매시간마다 표본 동작의 비교되어 그 일치 정도를 백분율로 표기된다. 표본 동작과 입력 동작은 동작이 이루어지는 시간이 틀리기 때문에 직접적으로 비교를 할 수 없고 이것을 위해 전처리 과정을 거친다.

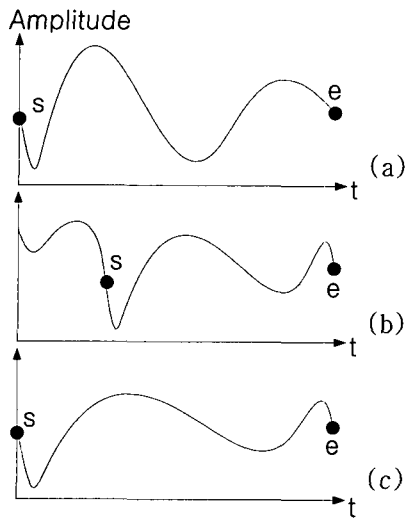


그림 6. (a) 표본 동작의 시작('s')과 끝('e')에 대한 기준점, (b) 입력 동작과 표본 동작의 기준점 매칭 결과, (c) (b)에서 얻어진 움직임을 표본 동작의 시간 거리로 보간한 결과.

Fig. 6. (a) Start('s') and end('e') points of a sample motion, (b) the matching result between each basis points of an input motion and a sample motion, (c) the interpolating result of the obtained motion in Fig 6(b) by the same time-length of the sample motion.

동작의 속도에 독립적인 값을 만들기 위해, 첫째, 그림 6(a)의 점 "s", "e"와 같이 표본 동작에서 기준점을

설정한다. 이 기준점은 동작의 시작과 끝에 존재하며, 가중치가 높은 특징에 부여한다. 둘째, 인식할 동작이 매 시간마다 입력되면, 그림 6(b)와 같이 표본 동작의 기준점들과 대응되는 점들의 위치가 입력 동작에 나타나기를 검사한다. 동작 사이클이 끝난 후, 동작의 전체 데이터를 가지고 검색 및 비교하는 것이 가장 쉽지만, 동작의 인식은 실시간으로 이루어져야 하므로 입력 동작의 현재 위치값이 표본 동작의 끝점 "e"과 유사한지 검사한다. 유사할 경우, 저장된 이전 프레임의 데이터를 역으로 검사해 표본 동작의 시작점 "s"와 유사한 위치에 놓은 점을 찾는다. 셋째, 동일한 기준점 배열이 발견되면, 그 배열의 시간 거리를 보간법을 이용해 표본 동작의 시간 거리와 동일하게 만든다(그림 6(c)). 표본 동작과 입력 동작의 시간 거리가 동일하면, 3차원 공간의 각 축에 대해 두 동작의 차를 구한다. 이 과정은 각 특징에 대해 개별적으로 적용되고, 각 특징의 일치율은 식 (13)와 같이 백분율로 표현한다. 실험적으로 입력 동작이 표본 동작에 대해 75%이상의 일치율을 보일 경우, 동작이 동일하다고 가정한다. 동작이 일치하면 상태 친이도에 따라 현재의 상태에서 적절한 다음 상태로 친이된다.

$$C = 100 \cdot \left( 1 - \frac{\sum_{n=0}^{N-1} |S(n) - I(n)|}{2 \cdot \sum_{n=0}^{N-1} |S(n)|} \right) \quad (13)$$

여기서,  $S(n)$ 은 표본 동작의  $n$ 번째 데이터, 그리고  $I(n)$ 은 입력 동작의  $n$ 번째 데이터,  $C$ 는 일치율을 나타낸다.

## 2. 얼굴 방향을 이용한 의미 표현

얼굴의 방향 전환을 이용해 아바타가 바라보는 방향, 아바타의 이동, 퍼피의 동작에 대한 긍정 또는 부정의 표현에 대한 의미를 얻을 수 있다. 머리의 동작은 손의 자유로운 동작과 달리 좌우상하로의 회전, 그리고 의미 없이 움직이는 동작으로 한정되어 있다. 그러므로, 머리의 동작은 하나의 동작이 복수개의 의미 - 예를 들면, 좌우로 움직이는 동작으로부터 "부정"의 동작과 "좌우로 둘러보는" 동작과 같은 - 를 구별해야 한다. 이것을 위해, 머리의 움직임은 표 3에 기술된 것과 같이 각 명령에 대한 유효성을 인정할 수 있는 시간 내에 이루어질 경우에만 명령으로 인정되도록 한다. 그렇지 않을 경우, 움직임은 의미없는 동작으로 간주되고 가상 공간

내의 아바타는 이러한 움직임에 대해 바라보는 시선의 방향을 이동시키는 동작으로 반응한다.

아바타 모드에서, 아바타의 이동은 동작의 유효성을 인정할 수 있는 시간 내에 머리를 상하로 한번 끄덕임으로써 이루어진다. 아바타가 이동중일 때 다시 한번 머리를 상하로 움직임으로써 아바타의 이동을 멈춘다. 퍼피 모드에서는 일정 시간 내에 머리를 상하로 한번 움직이는 동작은 퍼피의 행동에 대해 긍정의 답을 전달하는 명령으로 인식된다. 그리고 좌우로 움직이는 동작은 부정의 답을 전달하는 명령으로 인식된다. 각 모드에 대해 위에서 기술된 머리의 움직임을 제외하고는 모두 의미없는 동작으로 간주한다.

3. 손 동작을 이용한 의미 표현

손은 머리와 달리 다양한 동작과 공간의 이동이 가능하다. 손의 동작에서는 아바타가 오브젝트를 잡거나 이동시키는 동작, 창문 또는 문을 여닫는 동작, 퍼피에게 명령을 내리고 퍼피의 행동에 대해 응답하는 동작에 대한 의미를 얻을 수 있다.

손의 동작 인식은 손 영역의 크기, 위치, 형상 정보와 동작 시간에 의해 결정된다. 동작자의 손의 위치 정보는 가상 공간내의 아바타의 손의 위치와 직접적으로 대응된다. 크기 정보는 움직임에 대한 상대적 크기 변화로 비교된다. 손의 위치나 크기의 변화가 어떠한 동작을 이루는 흐름에 부합되더라도 형상 정보가 그 동작에서 요구되는 형태와 일치하지 않으면 동작은 이루어지지 않는다. 오브젝트를 잡는 동작을 예로 들면, 동작자의 손은 가상 공간내의 오브젝트가 존재하는 상대적인 위치에 있어야 하며, 형상 정보는 손바닥에서 주먹 형태로 변화가 있어야 한다. 또한, 크기도 주먹보다 손바닥일 경우에 더 커야 동작의 인식이 이루어진다.

손의 동작은 주기성 동작과 비주기성 동작으로 나뉘어진다. 퍼피를 부르는 동작은 팔을 위아래로 반복적으로 움직이는 주기성 동작이고 오브젝트를 잡는 동작은 비주기성 동작이다. 주기성 동작은 인정 가능한 유효 시간 내에 동작이 완료되지 않으면 의미없는 동작으로 간주된다. 명령이 인정되기 위해서는 손의 크기, 위치, 형상 정보가 그 명령에 요구되는 정보와 일치하여야 하고 동작이 이루어지는 시간도 만족되어야만 한다. 표 3에 동작이 인정되기 위해 요구되는 영상의 수를 주기성 동작과 비주기성 동작으로 나누어 기술한다.

표 3. 동작 인정 요구 영상 수( 단위: frame, X: don't care)

Table 3. Number of required frames to approve valid gesture

구분	동작		최소 요구 영상 수	최대 요구 영상 수
	현재 상태	다음 상태		
주기성 동작	Neutralize	Call	13	20
	Neutralize	Pat	8	13
	Neutralize	Affirm	4	8
	Neutralize	Negate	5	8
	Rest	Call	13	20
	Rest	Affirm	4	8
	Rest	Negate	5	8
	Rest	Move	4	8
	Move	Stop	4	8
비주기성 동작	Neutralize	Point	3	X
	Neutralize	Release	3	X
	Rest	Grip Object	5	X
	Rest	Grip Window	5	X
	Rest	Grip Door	5	X
	Grip Object	Rest	3	X
	Grip Window	Rest	3	X
	Grip Door	Rest	3	X
	Grip Window	Open Window	1	X
	Grip Window	Close Window	1	X
	Grip Door	Open Door	1	X
	Grip Door	Close Door	1	X
	Stop	Rest	1(자동 전이)	X
	Release	Rest	1(자동 전이)	X
	X	Neutralize	1(자동 전이)	X
자기 전이			1	X

4. 명령 인식에 대한 행동 표현

명령은 상태 천이도에 의해 전체 24개로 조합된다. 그 중, 대표적인 명령의 인식 조건은 다음과 같다. "Neutralize" 또는 "Rest"에서 "Call"로 천이될 조건은 첫째, 최소, 최대 요구 영상 수 범위 내에 동작이 완료되어야 한다. 둘째, 동작이 이루어지는 손의 x축의 위치 변화가 없어야 하고, y축과 z축은 2주기동안 주기적으로 위치 변화가 생기며, 그 주기동안 위치 변화의 최대점과 최소점들은 각각 그 위치가 서로 일정해야 한다. 셋째, 손의 크기는 2주기동안 y축의 위치 변화에 비례해야 하며, 동작 완료까지 손바닥 형태를 유지해야 한다.

"Neutralize" 또는 "Rest"에서 "Negate"로 천이될 조건은 첫째, 최소, 최대 요구 영상 수 범위 내에 동작이 완료되어야 한다. 둘째, 머리의 y축을 기준으로 하는 수평축(horizontal axis)은 1주기동안 '-'방향 또는 '+'방향으로 증가하다가 다시 반대 방향으로 증가, 그리고

표 4. 명령의 인식 조건  
Table 4. The Condition of recognition of orders

동작		조건
현재상태	다음상태	
Neutralize	Call	최소 최대 요구 영상 수 내에 동작이 완료 x축의 위치 변화가 없어야 함. y축, z축은 2주기 동안 주기적으로 위치 변화가 생겨야 하고, 주기 동안 peak점들은 그 크기가 서로 일정해야 함.
Rest		손의 크기는 2주기 동안 y축의 위치 변화에 비해, 동작 완료까지 손바닥 형태를 유지해야 함.
Neutralize	Pat	최소 최대 요구 영상 수 내에 동작이 완료 y축, z축의 위치 변화가 없어야 함. x축은 2주기 동안 주기적으로 위치 변화가 생겨 야 하고, 주기동안 peak점들은 크기가 서로 일정 해야 함. 손의 크기는 일정. 동작 완료까지 손바닥 형태를 유지해야 함.
Neutralize	Affirm	최소 최대 요구 영상 수 내에 동작이 완료
Rest	Affirm	머리의 x축을 기준으로 하는 (는 1주기 동안 '-' 방향으로 증가하다가 다시 '+'방향으로 증가, 그 리고 0°로 돌아와야 함.
	Move	$\beta$ , $\gamma$ 각은 0°부근에서 변화가 없어야 함.
Move	Stop	
Neutralize	Negate	최소 최대 요구 영상 수 내에 동작이 완료
Rest Affirm		머리의 y축을 기준으로 하는 $\beta$ 는 1주기 동안 '-'방향 또는 '+'방향으로 증가하다가 다시 반대 방향으로 증가, 그리고 0°로 돌아와야 함. $\alpha$ , $\gamma$ 각은 0°부근에서 변화가 없어야 함.
Neutralize	Point	마지막 3프레임이상은 지시 형태를 유지해야 함. 마지막 2프레임이상은 크기 변화가 없어야 하고, 위치의 변화도 없어야 함.
Rest	Release	마지막 3프레임이상은 손바닥 형태를 유지해야 하고, 크기 변화와 위치 변화가 없어야 한다. 손 의 y축 위치는 얼굴 중심의 y축 위치의 부근, x 축 위치는 팔꿈치의 x축 위치 부근, z축은 시선 의 방향으로 50cm정도에 위치해야 함. 손의 크 기는 얼굴 영역 크기의 1/3이상이어야 함.
Rest	Grip Object	마지막 3프레임이상은 주먹이어야 함. 마지막 2프레임이상은 크기 변화가 없어야 함. 주먹의 형태가 갖추어지면 위치 변화가 없어야 함.
	Grip Window	주먹형태가 되기 전 최소 3프레임이상 연속으로 손바닥형태가 되어야 함.
	Grip Door	손의 위치와 가상 공간에 존재하는 object, 창문 손잡이, 문 손잡이의 위치가 일치해야 함.
Grip Object	Rest	마지막 2프레임이상은 손바닥이어야 함.
Grip Window		손바닥 형태가 이루어지기 전의 마지막 2프레임 이상은 주먹이어야 함.
Grip Door		
Grip Window	Open Window	동작의 시작부터 완료까지 주먹 형태이어야 한다. x축은 가상 공간내의 창문 손잡이가 움직일 수 있는 방향으로 움직여야 함.
	Close Window	y축, z축의 이동은 없어야 함.
Grip Door	Open Door	동작의 시작부터 완료까지 주먹 형태이어야 한다. x축, z축은 가상 공간내의 문 손잡이가 움직일 수 있는 방향으로 움직여야 함.
	Close Door	y축의 이동은 없어야 함.
Stop	Rest	
Release		
X 저기 루프 전이	Neutralize	다음 프레임에서 자동 천이 됨.

0°로 돌아와야 한다. 수직축(vertical axis), 광축(optical axis)의 각은 0°부근에서 변화가 없어야 한다.

"Rest"에서 "Grip Object/Window/Door"로 천이될 조  
건은 첫째, 주먹형태가 되기 전 최소 3프레임이상 연속  
으로 손바닥형태가 되어야 한다. 둘째, 동작의 마지막 3  
프레임이상은 주먹형태를 유지하며 크기와 위치의 변  
화가 없어야 한다. 셋째, 손의 위치와 가상 공간에 존재  
하는 오브젝트, 창문 손잡이, 문 손잡이의 위치가 일치  
해야 한다.

"Grip Window"에서 "Open/Close Window"로 천이  
될 조건은 첫째, 동작의 시작부터 완료까지 주먹형태가  
유지되어야 한다. 둘째, 손의 x축은 가상 공간내의 창  
문 손잡이가 움직일 수 있는 방향으로 움직여야 하며,  
y축, z축의 이동은 없어야 한다.

인식 명령에 대한 행동 표현은 표 4에 상세히 기술  
되어 있다.

#### IV. 휴먼 인터페이스를 이용한 상호 작용 실험

##### 1. 상호 작용 시스템 구성

동작자와 상호 작용을 위한 가상 공간은 동작자의  
동작을 수행하는 아바타와 스스로 독립적인 행동을 하  
는 퍼피, 비자율적인 객체들인 창문, 문, 여러 개의 탁  
자, 공과 같은 형태의 오브젝트 등으로 구성되어 있다.

그림 7(a)와 7(b)는 구현된 동작 인식 시스템내의 특  
징 추출 부분에 대한 예와 상호 작용 시스템내의 가상  
공간에 대한 예를 보여준다. 그림 7(a)에서, 위쪽 영상  
은 전방상단 카메라로부터 입력된 이미지이고, 아래쪽  
영상은 전방 카메라로부터 입력된 이미지이다. 우측 상  
단의 박스에서 얼굴의 방향 정보를, 우측 하단 박스에  
서 두 손의 위치, 크기, 형상 정보에 대한 특징 추출의  
결과를 볼 수 있다. 이 추출 과정은 [21]에 기술되어 있  
다. 그림 8(b)와 그림 10(b)는 구현된 가상 공간의 일부  
를 보여주고 있다. 그림 8(b)에서는 두 개의 손잡이를  
가진 여닫이 창문과 오브젝트가 놓여질 수 있는 탁자  
를 볼 수 있다. 또한 그림 10(b)에서는 스스로 움직이  
는 자율적 객체인 퍼피의 뒷모습을 볼 수 있다. 그림의  
가운데에 출력되어 있는 십자 모양의 커서는 아바타가  
바라보고 있는 시선의 중앙을 나타내고 있다.

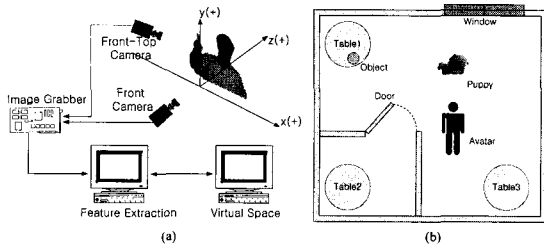


그림 7. (a) 동작 인식 시스템 구성도, (b) 상호 작용 시스템 구성도

Fig. 7. (a) HI system configuration, (b) virtual space configuration.

2. 휴먼 인터페이스를 이용한 상호 작용 실험

동작이 이루어지는데 요구되는 영상 수를 주기성 동작과 비주기성 동작으로 나누어 측정하였다(표 3). 동작 인식 시스템에서 얻어진 각 영상에 대한 특징은 내부 버퍼에 저장되고, 인식하고자 하는 동작은 그 동작이 요구하는 최소 소요 영상 수 보다 클 경우에만 버퍼에 저장되고 특징 정보를 읽어서 검사를 하였다. 만약, 주기성 동작일 경우 최대 소요 영상 수 내에 동작이 완료되지 않으면 인식되지 않는다.

실험은 아바타가 직접 창문을 여닫는 동작과, 아바타가 피피에게 오브젝트를 가져오라는 명령을 내려 간접적으로 가상 공간 내의 객체를 이동시키는 동작으로 나누어 이루어졌다. 첫번째 실험은 동작자가 가상 공간 내에서 자신의 의지를 수행하는 아바타와 직접적으로 상호 작용하는 것에 대한 실험이다. 두 번째 실험은 하나의 가상 공간 내에 아바타와 또 하나의 독립된 의지를 가진 객체인 피피를 설정한 후, 두 개의 독립된 객체가 상호 작용하는 것에 대한 실험이다. 실험은 동작 천이시 비교될 표본 동작의 동작 속도에 흡사한 속도로 동작을 하는 경우와 약간 느린 속도, 조금 빠른 속도에 대해 이루어졌다. 천이가 이루어지는 각 동작은 표본 동작을 가지고 있고, 동작자의 동작이 입력되는 때 시간마다 입력 데이터는 표본 동작의 것과 비교된다.

1) 아바타의 직접 행동에 대한 실험

이 실험은 동작자와 아바타와의 직접적인 상호 작용에 의해, 아바타가 직접 이동하여("Move") 오른손으로 창문의 손잡이를 잡은 후("Grip Window"), 창문을 여닫는 동작("Open Window", "Close Window")에 대한 것이다. 각 동작에 대한 머리와 두 손의 명령은 표 5와 같다.

실험은 앞서 설명한 것과 같이 동작자는 카메라가

표 5. 동작자와 avatar의 상호 작용에 대한 머리와 두 손의 명령

Table 5. Order of head and hands for interaction between an operator and an avatar

영역	명령
머리	Rest → Move → Stop → Rest
오른손	Rest → Grip Window → Open Window → Close Window → Rest
왼손	Rest

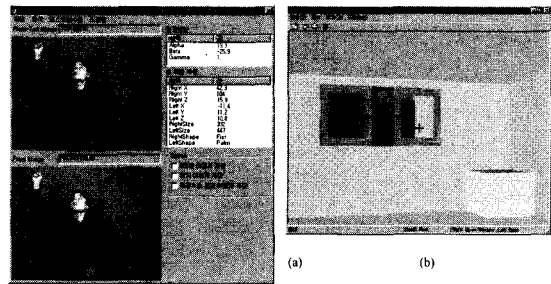


그림 8. "Open Window": (a) 특징 추출 결과, (b) 명령 후의 가상 공간

Fig. 8. "Open Window": (a) result of feature extraction, (b) a part of the virtual space after the order.

바라보는 공간 내에서 의자에 앉은 상태로 진행되었다. 초기 자세가 인식되어 가상 공간에 전달이 되면 아바타는 가상 공간의 한 방향을 바라보는 상태가 된다. 가상 공간 내의 창문을 찾기 위해 동작자가 머리를 이리저리 두리번거리면, 이 동작은 인식기에서 아바타의 시선 방향 이동에 대한 명령으로 인식이 된다. 아바타의 이동도 머리의 동작에 의해 인식되었다. 머리를 한번 끄덕이는 동작은 x축을 기준으로 하는 각 수직축을 변화시킨다. 아바타의 이동과 멈춤은 이 동작이 "Rest"에서 "Move"와 "Move"에서 "Rest"의 명령 인식 조건을 만족시킬 경우에 인식이 되었다. 또한, 동작자의 손의 위치와 가상 공간 내의 아바타의 손의 위치가 상대적으로 일치하기 때문에, 창문의 손잡이를 잡기위해 동작자가 오른손을 들어올릴 때에도 이 동작은 가상 공간에 그대로 전달이 됨을 확인할 수 있었다. 창문의 손잡이 위치와 아바타의 손의 위치가 동일하게 될 경우에만, 손잡이를 쥐는 동작("Grip Window")이 인식됨을 확인하였다. 마찬가지로, 창문을 여닫는 "Open Window", "Close Window", 그리고, 창문의 손잡이를

놓는 "Rest"도 손의 동작이 각각의 명령 인식 조건을 만족시킬 경우에 인식이 됨을 확인할 수 있었다. 그림 8(a)는 아바타가 창문을 열고있는 동작의 일부를 나타 내었다. 그리고, 그림 8(b)는 가상 공간 내의 아바타가 직접 이동해서 창문을 연 후, 창문이 열린 상태를 멀리서 바라본 장면이다.

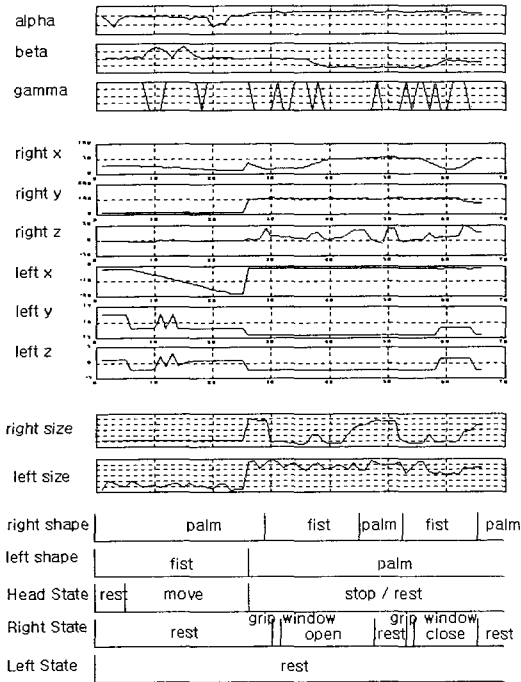


그림 9. 동작자와 avatar의 상호 작용에 대한 동작 인식의 과정

Fig. 9. A sequence of gesture recognition for interaction between an operator and an avatar

그림 9는 표본 동작과 흡사한 속도로 동작을 했을 경우에 대한 명령 인식의 과정을 나타내었다. 동작자의 움직임은 머리와 두 손의 각 특징의 3차원 위치변화에 의한다. 그림 9는 동작에 대한 위치 변화 과정을 보여 준다.  $\alpha$ 각이 감소했다가 증가한 후에 머리의 상태는 "Rest"에서 "Move"로, 다시 한번  $\alpha$ 각의 변화 후에 "Move"에서 "Stop"으로 천이했음을 확인할 수 있다. "Stop"에서 "Rest"로는 자동적으로 천이 된다. 또한, 오른손의 크기가 감소하고 형태가 주먹으로 바뀔 때 따라 오른손의 상태가 "Rest"에서 "Grip Window"로 천이했음을 알 수 있다. "Grip Window"명령의 인식 후, 오른손의 x축의 위치가 증가함에 따라 오른손의 상태는

"Open Window"로 천이하였고, 오른손의 이동이 멈춘 후에 오른손의 크기가 증가하고 형태가 손바닥으로 바뀔 때 따라 다시 "Grip Window"에서 "Rest"상태로 천이 하였다. 그리고, 오른손의 x축의 위치가 감소하면서 같은 작업을 반복하여 "Grip Window", "Close Window"와 같은 순서로 동작이 천이된 후, 다시 "Grip Window"와 "Rest"상태로 천이함을 확인할 수 있었다.

실험에 대한 인식률은, 아바타의 이동("Move"와 "Stop")에 대한 인식률은 평균 95.3%로 나타났고, 창문을 여닫는 동작("Open Window"와 "Close Window")에 대한 인식률은 평균 93.8%로 나타났다. 창문의 손잡이를 잡는 동작("Grip Window")은 79.3%로 비교적 낮게 나타났는데, 이것은 단지 쥐는 동작에 대한 인식률은 95%가 넘는 높은 인식률을 보였음에도 불구하고, 동작자가 창문을 바라보는 과정에서 조금만 고개를 움직여도 가상 공간의 바라보는 시점이 심하게 움직여지는 현상 때문에, 창문 손잡이 위치에 아바타의 손을 제대로 옮겨가는 것이 쉽지 않았고, 이 이유 때문에 손잡이를 잡는 동작이 어려웠던 것으로 해석되었다.

표 6. 동작자와 자율적 객체(퍼피)의 상호 작용에 대한 머리와 두 손의 명령

Table 6. Order of head and hands for interaction between an operator and a puppy.

영역	명령
머리	Rest → Affirm → Rest
오른손	Rest → Call → Point → Pat → Rest
왼손	Rest → Grip Object → Rest

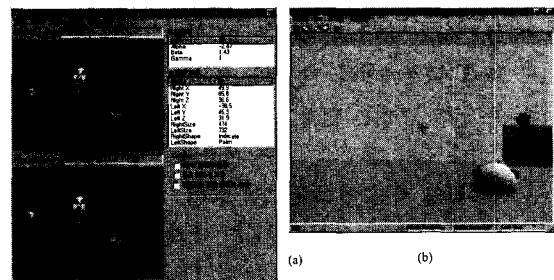


그림 10. Puppy에게 명령("Point")을 내리는 과정의 일부: (a) 특징 추출의 결과, (b) Puppy의 동작

Fig. 10. Part of the sequence that an operator activates a puppy and orders him by pointing: (a) result of feature extraction, (b) puppy's motion.

2) 아바타의 간접 행동에 대한 실험

이 실험은 아바타와 가상 공간 내에 존재하는 또 하나의 독립된 의지를 가지는 자율적인 객체인 퍼피간의 상호 작용에 대한 실험이다. 이 실험에서 아바타는 자유롭게 움직이고 있는 퍼피를 부른 다음("Call") 퍼피에게 오브젝트를 가져오라고("Point") 명령을 내린다. 이 명령을 받은 퍼피는 오브젝트가 있는 위치로 이동한 후 오브젝트를 잡고, 자신이 실행한 동작이 아바타가 요구한 명령과 일치하는지에 대한 응답을 기다린다. 이때 아바타가 긍정의 답을 하면 오브젝트를 가지고 아바타에게 돌아오고, 부정의 답을 하면 오브젝트를 원래 위치에 놓아두고 명령을 받기 전과 같이 가상 공간을 자유롭게 돌아다닌다. 실험에서는, 긍정("Affirm")의 답을 내린 후 왼손으로 퍼피가 가져온 오브젝트를 잡고("Grip Object"), 오른손으로 퍼피를 쓰다듬는다("Pat") 동작을 취했다.

아바타와 자율적 객체의 상호 작용에 대한 머리와 두 손의 명령은 표 6과 같다. 그림 10(a)는 퍼피를 부른 후 오브젝트를 가져오라는 명령("Point")을 내리는 동작자의 동작의 일부를, 그림 10(b)는 명령이 수행되는 가상 공간의 일부를 보여준다.

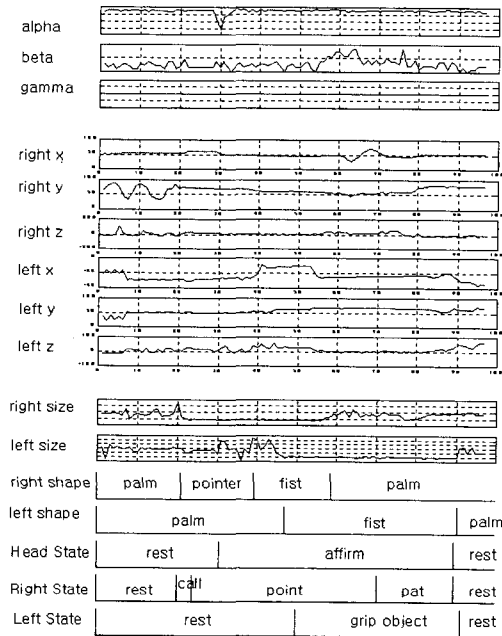


그림 11. 동작자와 자율적 객체의 상호 작용에 대한 동작 인식의 과정  
 Fig. 11. A sequence of gesture recognition for interaction between an operator and a puppy.

그림 11은 표본 동작과 흡사한 속도로 동작을 했을 경우에 대한 명령 인식의 과정을 나타내었다. 오른손이 수직으로 2주기 동안 왕복되면서 표 4에서 기술한 "Call"조건이 만족될 경우, 동작은 "Call"로 인식되었다. "Call"은 아바타가 퍼피를 부르는 동작이며 상태 천이도의 모드를 아바타 모드에서 퍼피 모드로 천이시킨다. "Call"명령 후, 오른손의 형태를 지시 형태로 유지하면서 위치를 고정시켰을 경우 퍼피에게 오브젝트를 가져오라는 "Point"명령이 인식되었다. 퍼피는 "Point"명령 실행 후에 아바타에게 자신의 행동에 대한 응답을 요구한다. 실험에서, 아바타가 긍정의 답을 하기 위해 머리를 한 번 끄덕이면 동작은 "Affirm"으로 인식되었다. 그림 12는 "Affirm"이 인식된 뒤의 퍼피의 동작으로써 긍정의 답변을 받은 퍼피가 오브젝트를 들고 동작자에게 다가오고 있는 상태를 보여준다.

아바타 앞에 퍼피가 도착하면, 왼손의 형태를 손바닥에서 주먹으로 바꾸어 퍼피가 가져온 오브젝트를 잡는다("Grip object"). 그리고, 오른손을 손바닥 형태를 유지하며 수평으로 1주기 동안 왕복하는 동작을 취해 퍼피를 쓰다듬어준다("Pat").

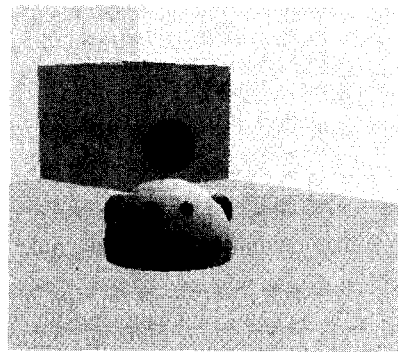


그림 12. "Affirm"명령 후의 puppy의 동작  
 Fig. 12. Puppy's motion after "Affirm".

퍼피를 부르는 동작("Call")에 대한 인식률은 평균 92.7%, 퍼피에게 명령을 내리는 동작("Point")은 97.5%, 퍼피를 쓰다듬어 주는 동작("Pat")은 69.8%로 나타났다. 퍼피를 부르는 동작은 손을 위아래로 반복해서 움직이는 동작이기 때문에, 손의 형태가 손바닥으로 비교적 잘 인식이 되었는데 비해, 퍼피를 쓰다듬어 주는 동작은 손의 형태가 잘 보이지 않는 동작이다. 카메라에 입력되는 손의 모습은 손을 카메라 방향으로 뺀 모습이기 때문에 손바닥의 형태로 인식되지 않는 경우가

많이 발생함으로 인해 낮은 인식율을 나타내었다. 또한, 긍정의 답을 하는 동작("Affirm")은 96.8%로 나타났다.

## V. 결 론

논문에서는 컨텍스트 기반의 인식법을 이용해, 동작자의 얼굴과 손으로부터 추출된 동작 정보를 연속된 시간의 흐름에 대해 해석을 함으로써 동작자의 동작을 인식할 수 있었다. 또한, 인식된 동작을 이용하여 동작자와 상호 작용할 수 있는 가상 공간을 구성할 수 있었다. 가상 공간은 동작자의 동작을 직접적으로 수행하는 아바타와 독립된 의지를 가진 자율적인 객체인 페피, 그리고, 비자율적인 객체인 문과 창문, 공과 같은 형태의 오브젝트들이 존재한다. 동작자는 아바타를 이용해 가상 공간에서 직접 어떠한 작업을 수행하거나, 페피와 상호 작용을 함으로써 간접적으로 작업을 수행할 수 있었다.

제한한 상태 천이도는 머리와 두 손의 동작을 독립적으로 처리하기 때문에, 한 손은 페피 모드에서 페피에게 명령을 내리고, 다른 한 손은 아바타 모드에서 아바타가 직접 작업을 할 수 있는 유연성을 얻을 수 있었고, 동시에 발생할 수 없는 동작에 대해서도 동작의 천이가 이루어지기 전에 미리 검사를 함으로써 시스템의 안정성을 얻을 수 있었다.

인식되는 동작은 주기성 동작과 비주기성 동작으로 구분되었으며, 주기성 동작의 경우, 인정 가능한 유효 시간 내에 동작이 완료되지 않으면 의미없는 동작으로 간주되었다. 특히, 손 동작의 경우, 손의 크기, 위치, 형상 정보가 그 명령에 요구되는 정보와 일치하여야 하고 동작이 이루어지는 시간도 만족되어야만 명령으로 인식이 가능하였다.

논문에서는 기계와의 인터페이스를 인간의 행동을 바탕으로 한, 인간 중심의 방식으로 접근함으로써 더욱 친밀감을 느낄 수 있는 시스템을 구현할 수 있었다. 구현된 시스템에서는, 동작자는 의자에 앉은 상태의 고정된 위치에서 머리와 두 손의 동작만으로 명령을 내렸다. 현재, 의자 등에 구속되지 않은 보다 자유로운 3차원 공간에서 머리와 두 손, 두 발을 비롯한 몸 전체의 다양한 동작을 추출 및 추적, 이해할 수 있는 더욱 지능적인 휴먼 인터페이스 시스템을 개발 중이다. 또한, 복수개의 아바타가 하나의 가상 공간 내에 동시에 존재하면서 같이 상호 작용할 수 있는 시스템을 개발할

계획이다.

## 참 고 문 헌

- [1] K.H. Jo, Y. Kuno, and Y. Shirai, "Manipulative hand gesture recognition using task knowledge for human computer interaction", *Proc. 3rd IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 468~473, 1998.
- [2] Francis K.H. Quek, "Unencumbered Gestural Interaction", *IEEE MultiMedia*, Vol. 3, No. 4, pp. 36~47, Winter 1996.
- [3] Ming-Hsuan Yang and Narendra Ahuja, "Extraction and Classification of Visual Motion Patterns for Hand Gesture Recognition", *In Proceedings of the IEEE CVPR*, pp. 892~897, Santa Barbara, 1998.
- [4] Maylor K. Leung and Yee-Hong Yang, "First Sight: A Human Body Outline Labeling System", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 17, No. 4, April 1995.
- [5] David J. Beymer, "Face Recognition Under Varying Pose", *A.I. Memo No. 1461*, Artificial Intelligence Lab., MIT, 1993.
- [6] Greg Welch and Gray Bishop, "An Introduction to the Kalman Filter", *TR 95-041*, University of North Carolina at Chapel Hill, February 8, 2001.
- [7] C.H. Sul, S.K. Jung, and K.Y. Wohn, "Synthesis of Human Motion using Kalman Filter", *Proceedings of CapTech'98, First International Workshop on Modeling and Motion Capture Technologies for Virtual Environments*, 26-28 November 1998, Geneva, Switzerland.
- [8] Matheen Siddiqui, "Calibration and its Application to Stereo", *ENG SC467*, Senior Honors Thesis, Fall 1999.
- [9] Zhengyou Zhang, "A Flexible New Technique for Camera Calibration", *Technical Report, MSR-TR-98-71*, Microsoft Research, One

- Microsoft Way, Redmond, WA 98052-6399, USA.
- [10] Emanuele Trucco and Alessandro Verri, "Introductory Techniques for 3-D Computer Vision", *Prentice Hall*, New Jersey, 1998.
- [11] Ismail Haritaoglu, David Harwood, and Larry S. Davis, "W4: Real-Time Surveillance of People and Their Activities", *IEEE Transaction on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 8, August 2000.
- [12] J.K. Aggarwal and Q. Cai, "Human Motion Analysis: A Review", *Computer Vision and Image Understanding*, Vol. 73, No. 3, pp. 428~440, March 1999.
- [13] Christopher Wren, Ali Azarbayejani, Trevor Darrell, Alex Pentland, "Pfinder: Real-Time Tracking of the Human Body", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 19, No. 7, pp. 780~785, July 1997.
- [14] Christopher R. Wren and Alex P. Pentland, "Dynamic Models of Human Motion", *IEEE International Conference on Automatic Face and Gesture Recognition*, Japan, April 14-16, 1998.
- [15] Ali Azarbayejani and Alex Pentland, "Real-time self-calibrating stereo person tracking using 3-D shape estimation from blob features", *ICPR'96*, Vienna, Austria, August 1996.
- [16] Pattie Maes, Bruce Blumberg, Trevor Darrell, and Alex Pentland, "The Alive System: Wireless, Full-body Interaction with Autonomous Agents", *ACM Multimedia Systems*, 5:105-112, 1997.
- [17] F. Parke, "Parameterized models for facial animation", *IEEE Computer Graphics and Applications*, 2(9):61-68, November, 1982.
- [18] D. Terzopoulos and K. Waters, "Physically-based facial modeling, analysis, and animation", *Visualization and Computer Animation*, 1:73-80, 1990.
- [19] Y. Lee, D. Terzopoulos, and K. Waters, "Realistic Modeling for Facial Animation", *In Computer Graphics Proceedings, Annual Conference Series, 1995*, ACM SIGGRAPH, pp. 55~62.
- [20] Lawrence R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", *Proc. of the IEEE*, Vol. 77, No. 2, pp. 257~286, Feb. 1989.
- [21] 김성은, 조강현, 전희성, 최원호, 박경섭, "인간의 행동 인식을 위한 얼굴 방향과 손 동작 해석", *제어 자동화 시스템 공학 논문지*, Vol. 7, No. 4, pp. 309~318, April, 2001

저 자 소 개



金 聖 恩(正會員)  
1973년 2월 13일생. 1998년 울산대학교 제어계측공학과 졸업(공학사). 2000년 동대학원 졸업(공학석사). 1999년~2000년 한국엑시스 기술연구소 연구원. 2000년~현재 ETRI 가상현실연구센터 연구원. <주관심

분야 : 컴퓨터비전, 휴먼컴퓨터인터랙션(HCI), 가상현실>



曹 綱 鉉(正會員)  
1964년 3월 19일생. 1989년 부산대학교 정밀기계공학과 졸업(공학사). 1993년 일본 Osaka대학 대학원 전자제어기계공학과 졸업(공학석사). 1997년 Osaka대학 대학원 전자제어기계공학과 졸업(공학박사). 1997

년~1998년 ETRI 시스템공학연구소 포스트닥터연구원. 1998년~현재 울산대학교 제어계측공학과 조교수. <주관심분야 : 컴퓨터비전, 휴먼컴퓨터인터랙션(HCI), 지능시스템 제어>





李仁浩(正會員)

1968년 3월 8일생. 1989년 부산대학교 정밀기계공학과 졸업(공학사).

1991년 부산대학교 정밀기계공학과 졸업(공학석사). 1996년 오사카대학 정보공학과 졸업(공학박사). 1996

년~현재 ETRI 컴퓨터소프트웨어연

구소 가상현실연구센터 모션정보연구팀장. <주관심분야: 가상현실, HCI, 컴퓨터비전>