휴먼-로봇 상호작용을 위한 비전 기반 3차원 손 제스처 인식

노명철°, 장혜민, 강승연, 이성환
고려대학교 정보통신대학 컴퓨터·통신공학부
{mcroh, hmjang, sykang, swlee}@image.korea.ac.kr

Vision-based 3D Hand Gesture Recognition for Human-Robot Interaction

Myung-Cheol Roh°, Hye-Min Chang, Seung-Yeon Kang, Seong-Whan Lee
Division of Computer and Communication Engineering, Korea University

요 약

최근 들어서 휴머노이드 로봇을 비롯한 로봇에 대하여 관심이 증대되고 있다. 이에 따라, 외모를 닮은 로봇 뿐 만 아니라, 사람과 상호 작용을 할 수 있는 로봇 기술의 중요성이 부각되고 있다. 이러한 상호 작용을 위한 효율적이고, 가장 자연스러운 방법 중의 하나가 비전을 기반으로 한 제스처 인식이다. 제스처를 인식하는데 있어서 가장 중요한 것은 손의 모양과 움직임을 인식하는 3차원 제스처 인식이다. 본 논문에서는 3차원 손 제스처를 인식하기 위하여 3차원 손 모델 추정 방법과 명령형 제스처 인식 시스템을 소개하고, 수화, 지화 등으로의 확장성을 위한 프레임워크를 제안한다.

1. 서 론

최근 들어서 휴머노이드 로봇을 비롯한 로봇에 대하여 관심이 증대되고 있다. 휴머노이드 로봇은 사람의 형태를 닮은 형태의 로봇으로써 사람에게 친숙한 형태의 모습을 보임으로써, 기존의 공업용 목적 뿐 아니라, 가정, 식당 등의 일상생활에서 사람에게 거부감 없이 서비스를 제공할 수 있는 목적을 가진다. 이러한 서비스를 위해서는 사람이 원하는 명령을 로봇이 인식하기 위한 기술이 필수적이다. 또한, 로봇 선진국에서는 고령화 사회를 대비하여, 노인 보조를 위한 로봇의 개발이 각광을 받고 있으며, 노인의 명령을 인식하여 필요한 행동을 취하는 수동적인 서비스 뿐 만 아니라, 위급 상황 등을 파악하는 능동적인 서비스의 개발이 지속적으로 연구되어오고 있다.

지능형 휴머노이드 로봇을 위하여서는 사람과 자연스러운 상호 작용 기술의 개발이 필수적이다. 손 동작과 표정 등의 비수지적(non-manual) 표현을 이용하여 대화의 내포적인 의미 및 감성 인식도 능동적인 서비스 제공을 위하여 필요하므로 수화, 지화의 연구와 더불어 많이 연구되고 있다[8,9]. 이러한 기술들 중에서 기타 부착장치 없이, 가장 자연스럽게 상호 작용할 수 있는 방법은 비전을 기반으로 둔 손 제스처 인식 기술이다. 사람의 제스처 중에서 가장 중요하고, 많은 의미를 가지고 있는 것은 손과 팔을 이용하는 손 제스처라고 할 수 있다. 대표적인 예로 수화, 지화, 명령형 제스처를 들 수 있다. 사람의 움직임 변화에 강인한 인식을 위하여서는 가려짐과 방향을 고려한 3차원 손 제스처 인식이 필수적이다.

본 논문에서는 3차원 손 제스처 인식을 위하여서 3차원 손 포즈 추정 및 궤적을 이용한 방법론을 소개하고, 수화, 지화로의 확장을 위한 3차원 손 제스처 인식 시스템 프레임워크를 제안한다. 3차원 손 제스처 인식을 이용한 활용 분야로는 휴먼-로봇 상호작용, 게임 제어, 수화/지화 통역 등을 들 수 있다.

2. 3차원 손 제스처 인식 시스템

3차원 손 제스처를 인식하기 위해서 기본적으로 3차원 손의 모양을 인식하는 단계와 손, 팔의 움직임을 분석하는 단계의 두 단계가 필요하고, 손을 추출하고, 팔의 3차원 상 움직임을 추적하고 분석하기 위해서는 3차원 휴먼 모델의 재구성이 필요하다. 다음 그림 1은 이러한 3차원 손 제스처 인식을 위한 기술의 구성도를 보여준다.

3차원 휴먼 모델 재구성을 통해 얻어진 몸의 구성 요소들을 추적함으로써 손과 팔의 움직임을 추적할 수 있고, 손의 영역을 추출할 수 있다. 3차원 손 포즈 추정을 이용하여 각도 및 자기 겹침(Self Occlusion)에 강인한 손 모양을 인식하고, 손의 움직임을 분석함으로써 명령

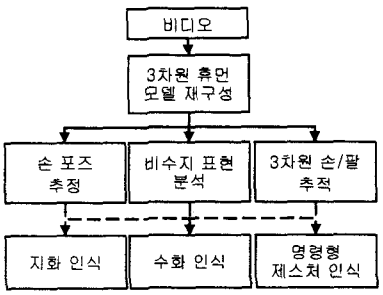


그림 1. 손 제스처 인식을 위한 기술 구성도

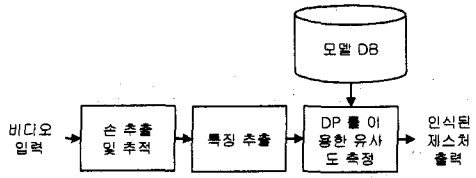


그림 2. 명령형 제스처 인식 시스템 구조

형 제스처를 인식한다. 이러한, 손 모양 인식과 명령형 제스처 인식의 기술 그리고, 비수지 표현을 분석함으로써 수화 인식 시스템을 개발할 수 있다.

본 논문에서는 지화 및 수화 인식으로 확장하기 위하여, 3차원 손 모양 추정과 3차원 손/팔의 움직임을 이용한 명령형 제스처 인식 기술을 다룬다. 손 포즈 추정을 위하여서 3차원으로 렌더링 된 영상을 사용하고, 명령형 제스처 인식을 위하여서는 Korea University Gesture Database(KUGDB)[7]와 카메라에서 입력받은 영상을 사용한다.

3. 명령형 제스처 인식 기술

명령형 제스처 인식을 위해서한 손 제스처만 고려하고 손의 위치, 각도, 속도의 특징들을 이용하여 제스처를 표현한다. 손의 특징을 추출하기 위하여 입력 영상에서 손 영역에 해당하는 손의 좌표를 추출하고 추적한다. 그림 2는 명령형 제스처 인식 시스템의 구조를 보여준다.

3.1 특징 추출 및 정규화

명령형 제스처 인식을 위한 특징으로 각 입력 영상으로부터 추적되는 손의 위치, 속도, 각도를 사용한다. 다음 수식 (1), (2), (3) 은 각각 위치, 각도, 속도 특징을 나타낸다. (C_x, C_y)은 궤적에서의 무게중심, l_t는 시간 t에서 중심과의 거리, θ_1는 무게중심과 현재 점과의 각도, θ_2는 연속된 두 점사이의 각도, v_t는 연속된 두 점 사이의 속도를 나타낸다. 식(1)에서 L_{max}는 중심점과 어떠한 점 사이로부터 가장 긴 거리를 말한다. 따라서 l_t 정규화한 값을 가지게 되고 이때 범위는 0에서부터 1사이의 값으로 되어있다. 식(3)에서 V_{max}는 두 점사이의 최대 속도 값을 말하고 범위 또한 0에서 1사이의 값을 갖는다. 그림 3은 수식 (1), (2), (3) 을 이용하여 사용된 특징 벡터를 보여준다.

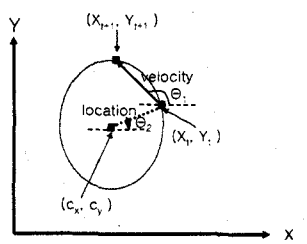


그림 3. 명령형 제스처 인식을 위한 특징

$$l_t = \frac{L_t}{L_{max}}, \quad where\, L_{max} = \max_{t=1}^{n}(L_t) \tag{1}$$

$$\theta_1 = atan2(d_{x1}, d_{x1}),\ \theta_2 = atan2(d_{x2}, d_{x2})$$
$$where\ d_{x1} = X_t - C_x,\ d_{y1} = Y_t - C_y,\ d_{x2} = X_t - X_{t+1},\ d_{y2} = Y_t - C_{t+1} \tag{2}$$

$$v_t = \frac{V_t}{V_{max}}, \quad where\, V_{max} = \max_{t=1}^{n}(V_t) \tag{3}$$

3.2 제스처 인식

본 논문에서 명령형 제스처를 인식하기 위하여 동적 프로그래밍을 이용하여, 모델 제스처의 특징 템플릿과 입력 제스처의 특징 템플릿 사이의 최소 거리를 계산함으로써 제스처를 인식한다[5].

모델 제스처의 특징은 $M=(M_1,\cdots,M_m)$으로 하자. 이때 M_i는 i번째 모델 프레임에서 추출된 특징 벡터이다. 마찬가지로 입력 제스처는 $I=(I_1,\cdots,I_j)$으로 나타낸다. 특징 벡터 M_i와 I_j가 주어졌을 때 거리는 d(i,j), 누적 거리는 D(i,j)로 나타낸다. 누적 거리는 식(4)와 같이 계산한다.

$$D(i,j) = \min D(i-1,j), D(i-1,j-1)D(i,j-1) + d(i,j) \tag{4}$$

이 때 경로의 (i-1,j-1),(i-1,j), (i,j-1) 중 가장 최단거리를 선택하고 현재 (i,j)를 더하여 누적 거리를 계산함으로써 모델과 입력 제스처의 유사도를 측정한다. 또한, 후진 알고리즘을 이용하여 최적의 경로를 찾음으로써 제스처의 시작 시점을 찾을 수 있다. 제스처 인식은 입력 제스처의 특징 템플릿과 가장 작은 거리를 가지는 데이터베이스에 있는 모델 제스처의 특징 템플릿을 찾음으로써 수행된다.

4. 3차원 손 포즈 추정

3차원 손 포즈 추정은 다음과 같이 세 가지 과정으로 구분된다. 첫 번째는 데이터베이스를 생성하는 과정, 그리고 두 번째와 세 번째는 각각 오프라인 임베딩 프로세스와 온라인 임베딩 프로세스이다. 그림 4는 포즈 추정을 위한 시스템을 보여준다.

4.1 데이터베이스 생성

이 시스템에서는 16개의 링크로 구성된 손 모델을 사용한다. 1개의 손바닥과 5개의 손가락으로서 각각의 손가락은 3개의 링크로 구성되어 있으며 20 DOF(Degree of freedom)을 갖는다. 이것을 형태 파라미터(Configuration parameter)라고 하며 여기에 시점 파라미터(Viewpoint parameter)를 더해 총 23가지의 DOF를 갖는다. 손의 형태를 나타내는 형태 파라미터 벡터 $C_h = (c_1, c_2,\cdots,c_{20})$와 시점을 나타내는 시점 파라미터 벡터 $V_h = (v_1, v_2, v_3)$가 주어질 때 한 장의 손 영상은 다음 식(5)와 같이 23개의 포즈 파라미터 벡터로서 표현될 수 있다.

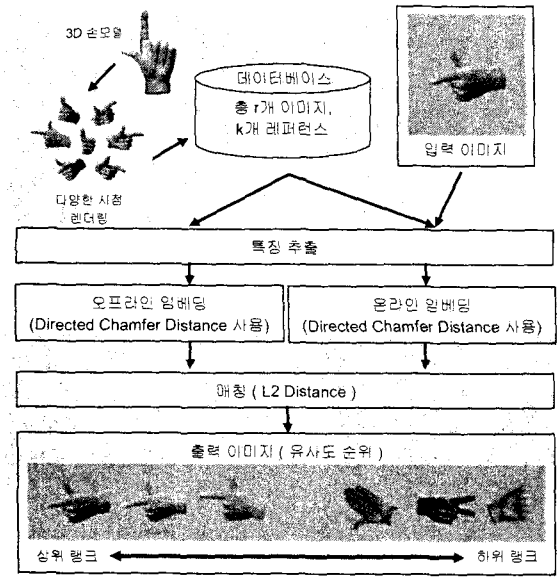


그림 4. 3차원 손 포즈 인식을 위한 시스템 구성

$$P_h = (c_1, c_2, c_3, \cdots, c_{20}, v_1, v_2, v_3) \quad (5)$$

3차원 손 이미지는 컴퓨터그래픽 툴로 렌더링하여 얻어지며 이때 파라미터 정보들도 렌더링된 이미지와 함께 데이터베이스에 저장된다.

4.2 임베딩 프로세스

오프라인 임베딩 프로세스는 데이터베이스의 r개의 이미지에 대한 각각의 k개의 레퍼런스 이미지 사이의 거리를 임베딩 하여 그 결과를 저장한다. 온라인 임베딩 프로세스에서는 입력이미지에 대한 k개의 레퍼런스 이미지 사이의 거리를 임베딩 하고 이 값을 오프라인으로 미리 계산되어 저장된 값과 매칭하여 가장 유사한 이미지가 선택되어진다.

4.3 포즈 파라미터 매칭

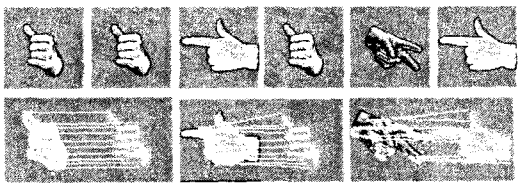
모델의 포즈를 추정하기 위하여 학습된 포즈와 입력된 포즈를 에지 정보를 이용하여 매칭을 수행한다. 에지 매칭을 수행하기 위하여 본 논문에서는 Chamfer Distance를 Lipschitz 임베딩 방법[3]을 이용해 근사화한 Directed Chamfer Distance를 이용한다[1].

Chamfer distance는 에지 간의 거리를 측정하기에 효과적이며 노이즈에 강인한 널리 알려진 방법이다[2]. 에지 이미지는 각각의 픽셀 위치에 대응하는 점들의 집합으로서 나타내어지며 이미지 A에서 B로의 "Directed chamfer distance" c(A, B)는 아래의 수식(6)와 같이 정의된다. 여기에서 는 두 픽셀 a와 b의위치를 나타내는 a(x,y)와 b(x,y)사이의 유클리디안 거리이다. "Undirected chamfer distance"는 아래의 수식(7)과 같이 정의되며 이는 A에서 B로의 "Directed chamfer distance"와 B에서 A로의 "Directed chamfer distance"의 합으로 나타내어진다.

$$c(A,B) = \frac{1}{|A|} \sum_{a \in A} \min_{b \in B} |a - b| \quad (6)$$

$$C(A,B) = c(A,B) + c(B,A) \quad (7)$$

본 논문에서는 "Directed chamfer distance"의 약자로서 DCD를, "Undirected chamfer distance"는 CD로 표기한다. 그림 5는 데이터베이스 이미지에 대해 매칭된 입력이미지의 예를 보여준다. 모델 이미지와 입력이미지 간의 DCD는 모델의 에지 이미지가 n개의 점들로 구성되어있으며 데이터베이스에 총 d개의 이미지가 있을 때, 시간 복잡도는 $O(n\log n)$이다. 따라서 데이터베이스의 이미지개수인 d와 에지를 구성하는 픽셀수 n이 커짐에 따라 시간이 많이 걸리며 많은 메모리를 필요로 하게 된다.



(a) (b) (c)

그림 5. Chamfer Distance로 매칭된 두 영상의 예

다차원공간상의 거리를 저차원 공간상으로 임베딩하는 기술은 최근 많은 관심을 받고 있다. 임베딩에 있어서, 임의의 공간 G에서 k차원 공간 R_k상으로 임베딩하였을 때, 공간 G상의 두 점간의 거리가 공간 R_k상에서 왜곡을 최소화하면서 효과적으로 보존이 되는가가 중요하다. 이러한 임베딩은 공간 G상에서의 거리측정방법의 계산량이 많은 경우 저차원 R_k상으로 매핑한 뒤 Lp norm의 연산으로서 복잡한 연산을 대체할 수 있으므로 유용하다.

본 논문에서는 Lipschitz embeddings를 사용하여 데이터베이스에 인덱싱하는 방식으로 계산복잡도 문제를 해결한다[1,4]. Lipschitz embeddings의 기본적인 아이디어는 두 개의 가까운 점은 제 삼의 점에 대하여 비슷한 거리를 갖는다는 것이다. 입력 에지이미지 g로부터

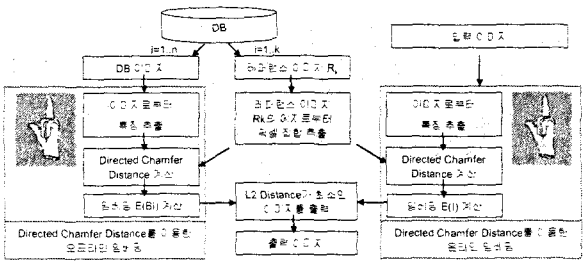


그림 6. 근사화된 Directed Chamfer Distance를 이용한 손 포즈 추정 방법

R_k상으로의 Lipschitz embedding E(g)는 아래의 식(8)와 같이 정의된다. 이 때 r_1, r_2,…,r_k는 데이터베이스로부터 임의로 선택되어진 k개의 레퍼런스 이미지를 의미하며 c는 식(6)에서 정의되어진 DCD를 뜻한다. 이러한 방식을 근사화된 Directed Chamfer Distance라 한다. 그림 6은 근사화된 Directed Chamfer Distance를 이용한 손 포즈 추정 방법을 보여준다.

$$E(g) = (c(g, r_1), c(g, r_2), \ldots, c(g, r_k)) \qquad (8)$$

5. 실험 및 결과 분석

5.1 명령형 제스처 인식 실험

명령형 제스처 인식을 위한 제스처는 숫자 0~9까지를 쓰는 제스처 사용하여 실험을 하였다. 본 논문에서는 실험을 위해서 사용된 데이터는 두 가지 종류로, KUGDB의 숫자 쓰기 데이터 총 100개와 웹 카메라를 이용하여 촬영한 데이터 50개로 총 150개이다. KUGDB를 이용한 실험은 KUGDB에서 신체 구성 요소 정보 파일을 이용하여 손의 좌표를 추출하였다. 웹 카메라를 이용한 실험에서 제스처의 시작과 끝을 손을 잠시 멈추어 있는 동작을 이용하여 추출하였다.

KUGDB에서 제공하는 신체 구성 요소 데이터는 그림 7과 같이 계층적 구조로 이루어져있고 각 관절은 이동과 회전 정보를 가지고 있다. 이 데이터를 이용하여 손의 좌표를 추출하고 이 좌표로 그림 8과 같은 10가지의 숫자 모델을 생성한다. 그림 9는 웹 카메라를 이용한 입력 영상과 추적되는 결과를 보여준다.

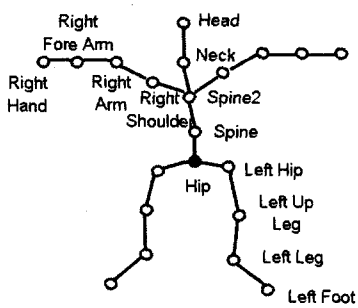


그림 7. KUGDB의 신체 구성 요소의 계층적 구조

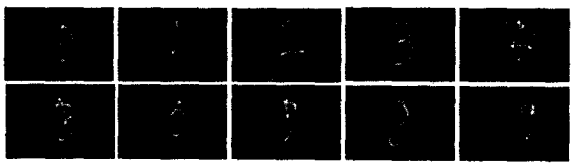


그림 8. KUGDB에서 추출한 손의 궤적 정보

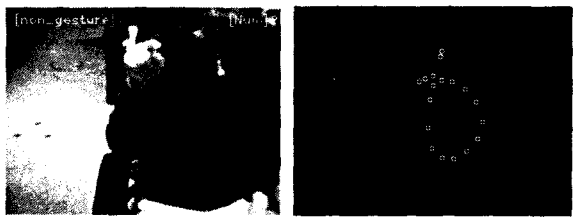


그림 9. 웹 카메라를 이용한 제스처 인식 인터페이스

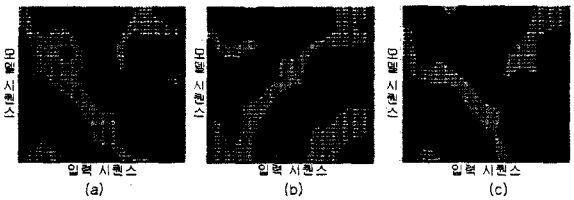


그림 10. 입력 제스처(5)와 모델(5, 6, 7)에 대한 DP 테이블

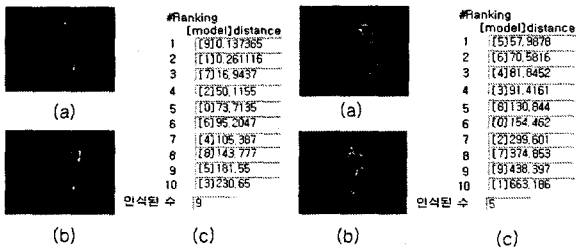


그림 11. 잘못 인식된 예((a) 입력된 숫자 (b) 인식된 숫자 (c) 입력 제스처와 모델 제스처 사이의 거리)

그림 10은 입력된 제스처와 데이터베이스에 있는 모델 제스처와의 거리를 측정한 DP 테이블을 보여준다. 특정 임계값을 주어서 그 임계값보다 크면 1값을 주고 작으면 0을 주어 이진영상으로 나타내었다. 모델과 매칭이 잘 되면, 그림 10(b)와 같이 대각선 형태가 나타나게된다.

표 1은 인식결과를 나타내고 있다. KUGDB데이터는 KUGDB를 사용한 결과를 웹 카메라는 웹 카메라로 촬영한 영상을 사용한 결과를 각각 보여준다. 여러 가지 잡영과 추적의 오류로 인해서 웹 카메라를 사용하여 실험한 결과가 KUGDB 데이터를 사용한 결과에 비해 14%정도 저하되는 결과를 보여준다. 비슷한 궤적을 가지고 있는 숫자들이 잘못 인식되는 경우가 발생한다. 그림 11은 서로 다른 숫자가, 비슷한 궤적을 가지고 있어서 잘못 인식되는 예를 보여준다. 그림 11은 입력된 제스처의 궤적(a), 인식된 숫자의 궤적(b), 그리고 입력된 제스처에 대한 전체 숫자 '0~9' 모델에 대한 거리를 보여준다. 주로 (1,7,9), (5,8), (0,6)의 쌍이 유사한 궤적을 가지고 있으므로, 비슷하게 인식된다.

표 1. 명령형 제스처 인식 실험 결과

사용된 데이터 / 결과	KUGDB 데이터	웹 카메라
Detection rate(%)	88%	74%
False matches	12/100	13/50

5.2 3차원 손 포즈 추정 실험

테스트를 위해서 그림 12와 같이 4가지 프로토타입을 정의하고 데이터 셋을 구성하였다. 각각의 4가지 프로토타입은 전방에서 바라보았을 때 수직으로 ±22.5도 이내, 수평으로는±90도 사이의 각도 이내의 범위에서 30가지 시점으로부터 본 손 모양을 렌더링하였다. 총 120장의 이미지 중 100장의 이미지는 오프라인 학습을 위해 사용하였고 나머지 20장의 영상은 테스트에서의 입력 영상으로 사용하였다. 데이터 셋의 100장의 이미지 중에서 20%는 레퍼런스 이미지로 (k=20) 사용되었다. 테스트에서 입력 이미지는 임의의 시점을 갖는 손 이미지이며 위치와 크기는 고정되었다고 가정하였다. 출력 이미지는 입력 이미지와 가장 유사한 이미지로서 매칭 결과가 최소인 이미지부터 차례로 10개의 이미지를 순위를 매겨 출력하였다. 아래의 그림 13은 테스트 결과의 상위 랭크 10까지 결과를 보여준다.

표 1은 실험 결과를 보여준다. 'F'는 잡영 없는 깨끗한 영상에서 테스트한 것이고 'C'는 임의의 잡영을 생성하여 추가한 영상에서 테스트한 것을 의미한다. ADCD는 이 논문에서 사용한 Approximated DCD 방법으로 테스트 한 것이고 DCD는 Directed Chamfer Distance를 사용하여 테스트 한 결과이다. Median은 Highest rank들의 중간값을 나타낸다. 실험결과 깨끗한 에지 이미지들을 사용해 실험한 결과 이 논문의 ADCD의 방법을 사용한 결과가 DCD만을 사용한 결과보다 인식율이 높게 나타났다. 잡영이 심하게 있는 이미지를 이용하여 테스트 결과 인식율은 깨끗한 에지이미지를 사용한 것 보다 떨어졌지만 DCD에 비해 ADCD를 사용한 쪽의 인식율이 상승한 것을 알 수 있다.

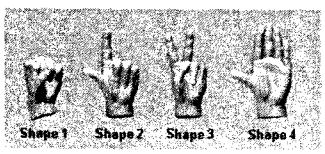


그림 12. 4가지 프로토타입

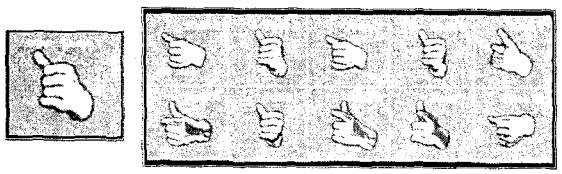


그림 13. 테스트에 사용된 입력 이미지와 상위랭크 10까지의 출력 결과 이미지

표 2. 손 포즈 추정 실험 결과

사용한 방법	1	1 - 3	1 - 6	1 - 12	Median
F, ADCD	85 %	90 %	100 %	100 %	1
F, DCD	75 %	90 %	95 %	100 %	1
C, ADCD	25 %	75 %	75 %	100 %	3
C, DCD	30 %	40 %	55 %	65 %	5

6. 결론 및 향후 연구

본 논문에서는 수화, 지화, 명령형 제스처 인식을 위한 명령형 제스처 인식과 3차원 손 포즈 인식 방법을 소개하였다. 각 방법은 3차원 제스처 데이터베이스(KUGDB), 카메라, 그리고, 3차원으로 모델링 된 3차원 손 모델에 대하여 실험을 하였다.

추후 연구로 양손을 이용한 복잡한 명령형 제스처의 인식과 다양한 손 모양에 대한 인식이 가능하도록 확장과 인식률을 높이는 방법에 대한 연구가 필요하다.

참고 문헌

[1] V. Athitsos and S. Sclaroff, "Estimating 3D Hand Pose from a Cluttered Image," Proc. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 2, June 2003, pp. 432-439.

[2] H. Barrow, J. Tenenbaum, R. Bolles, and H. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching," In International Joint Conference on Artificial Intelligence, 1977, pp. 659-663.

[3] J. Bourgain, "On Lipschitz embeddings of finite metric spaces in hilbert space," Israel Journal of Mathematics, Vol. 52, 1985, pp. 46-52.

[4] G. Hristescu and M. Farach-Colton, "Cluster-preserving embedding of proteins," Technical Report 99-50, Computer Science Department, Rutgers University, 1999.

[5] J. Alon, V. Athitsos, and S. Sclaroff, "Accurate and Efficient Gesture Spotting via Pruning and Subgesture Reasoning," Computer Vision in Human-Computer Interaction, Lecture Notes in Computer Science, Vol. 3766, October 2005, pp.189-198.

[6] H. S. Yoon, J. Soh, J. Bae, and H. S. Yang, "Hand Gesture Recognition Using Combined Features of Location, Angle and Velocity," Pattern Recognition, Vol. 34, No. 7, July 2001, pp. 1491-1501.

[7] B.-W. Hwang, S. Kim and S.-W. Lee, A Full-Body Gesture Database for Analyzing Daily Human Gestures," Intelligent Computing, Lecture Notes in Computer Science, Vol. 3644, August 2005, pp. 611-620. (http://Gesturedb.korea.ac.kr)

[8] H. Fillbrandt, S. Akyol, and K.-F. Kraiss, "Extraction of 3D Hand Shape and Posture from Images Sequences from Sign Language Recognition," Proc. International Workshop on Analysis and Modeling of Faces and Gestures, 2003, pp. 181-186.

[9] W. Gao, J. Ma, S. Shan, X. Chen, W. Zheng, H. Zhang, J. Yan, and J. Wu, "HandTalker: A Multimodal Dialog System Using Sign Language and 3-D Virtual Human," Proc. International Conference on Multimodal Interfaces, 2000, pp. 564-571.