동적 제스쳐 인식을 위한 지식 표현 기법

Knowledge Representation Method for Dynamic Gesture Recognition

고일주, 최형일
Il-Ju Ko, Hyung-Il Choi

숭실대학교 컴퓨터 학부
School of Computing, Soongsil University

요 약

본 논문은 컴퓨터 시각을 이용하여 동적 제스쳐를 인식하기 위한 효율적인 지식 표현 기법의 개발을 목표로 한다. 제스쳐란 시각적인 언어로서 소리를 대신하여 몸짓이나 손짓을 통하여 자신의 생각이나 의도를 전달하는 보조적인 의사 전달 수단이다. 제안된 기법은 여러 다양한 지식을 통합하여 총체적으로 표현하기에 적합한 프레임 구조를 기반으로 한다. 프레임 지식을 물체의 특성을 표현하는 객체 지식, 물체의 움직임을 표현하는 행동 지식, 그리고 객체 지식과 행동 지식의 순서화 된 집합으로써 동적인 제스쳐를 표현하는 스키마로 분류한다.

I. 서론

새로운 인터페이스의 개발은 컴퓨터의 사용이 대중화되고 많은 사람들이 컴퓨터를 사용하여야만 하는 입장에 처하게 됨으로써 키보드나 마우스 같은 입력장치에 익숙하지 않은 사람들을 대상으로 하여 매우 중요한 관심거리가 되고 있다. 특히 컴퓨터 시각을 기반으로 하는 제스쳐 인식 시스템에 대한 연구가 많이 진행되고 있다[1] [2]. 제스쳐는 인간의 의사 소통을 위해 글이나 음성과 함께 중요한 수단으로 사용되고 있다. 제스쳐란 시각적인 언어로서 소리를 대신하여 몸짓이나 손짓을 통하여 자신의 생각이나 의도를 전달하는 수단이다. 물론 주된 의사 교환의 수단을 사용하기에는 어렵지만 말로는 표현하기 힘든 느낌이나 상황들을 쉽게 표현할 수 있다는 장점을 갖고 있어서 보조적인 의사 전달 수단으로 많이 사용되고 있다.

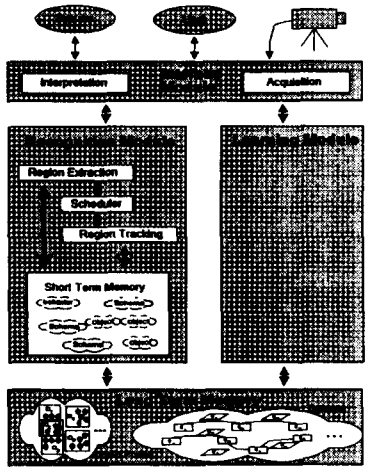
인간의 시각 시스템은 대개의 경우 목표 지향적이다. 즉 인간이 어떤 장면에서 특정한 물체를 인식하고자 할 때 미리 특정 물체가 존재할 것이라는 예상을 하고 그 예상을 확인하는 과정을 통해 물체를 인식한다. 이와 같이 인간이 문제 영역에 관한 지식을 물체를 인식하는데 사용하듯이 컴퓨터 시각 시스템도 문제 영역에 관한 지식을 이용하여 영상 분석 작업을 수행하는 것이 적절한 방법일 것이다. 주어진 문제 영역에 관한 지식을 어떠한 방식으로 표현할 것인가 하는 문제는 지식 표현의 효율성과 표현된 지식을 사용하여 수행되

는 영상 분석 작업의 효율성을 고려하여 결정되어야 한다. 이를 위하여 다양한 지식 표현 방법이 문헌에 소개되고 있으며, 영상 분석을 위하여는 프레임 구조의 지식 표현 방법을 많은 사람들이 선호하고 있다[3] [4]. 특히 연속적으로 입력되는 동적인 영상을 분석하여 물체의 동작에 대한 정보를 추론하기 위해서는 여러 유형의 지식을 필요로 한다. 즉, 영상 처리를 위한 여러 유형의 절차적 지식, 동작을 유발하는 물체의 형태 및 구조에 대한 객체 지식, 물체의 움직임에 대한 행동 지식, 예상되는 동작의 진행 순서에 대한 스크립트형 지식 등을 필요로 한다. 프레임 지식은 이러한 다양한 지식을 통합하여 총체적으로 표현하기에 적합하다.

본 논문은 컴퓨터 시각을 이용하여 동적 제스쳐를 인식하기 위한 효율적인 지식 표현 기법의 개발을 목표로 한다. 제안된 프레임 지식의 표현과 제어 기법의 유용성을 보이기 위해 제스쳐 인식 시스템을 구현하였다. 2장에서는 제스쳐 인식 시스템에 대한 전체적인 구성과 각 모듈에 대한 개략적인 설명을 한다. 3장은 본 논문에서 제안하는 프레임 지식의 표현 기법에 대하여 설명하고 4장에서는 제어 기법에 대하여 설명한다. 마지막으로 5장에서 제스쳐 인식 시스템의 실험 결과와 결론에 대하여 기술한다.

II. 제스쳐 인식 시스템

시스템은 인터페이스를 담당하는 인터페이스 모듈, 동적 제스쳐를 인식하는 인식 모듈, 작업 환경에 대한 환경 요소들을 학습하는 학습 모듈, 그리고 지식을 저장하고 Long Term Memory (LTM)의 4부분으로 구성되어 있으며, (그림 1)은 시스템의 전체적인 구성도를 보여준다.



(그림 1) 시스템 구성도

인터페이스 모듈(Interface Module)은 사용자, 응용 시스템, 그리고 카메라 같은 환경 요소와 제스쳐 인식 시스템간의 상호작용을 담당하는 역할을 수행한다. 사용자와 응용 시스템으로부터 명령을 입력받아 제스쳐 인식 시스템에게 전달하고, 제스쳐 인식 시스템의 결과를 해석하여 사용자나 응용 시스템에게 알려 준다. 그리고 카메라로부터 입력 영상을

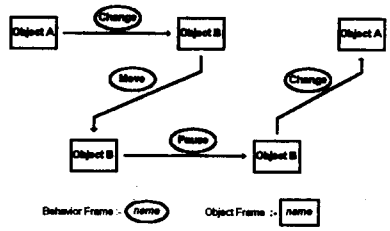
획하여 인식 모듈이나 학습 모듈에 전달하는 역할도 수행한다. 인식 모듈(Recognition Module)은 손 영역을 분리하는 기능을 수행하는 영역 추출(Region Extraction)과 영역 추적(Region Tracking), 이들을 관리하고 제어하는 계획기(scheduler), 그리고 프레임 지식들을 LTM에서 읽어 와서 저장하고 활성화하는 역할을 수행하는 Short Term Memory(STM)의 네 가지 요소로 구성되어 있다. STM은 제스쳐 영역이 획득되면 STM에 저장되어 있는 객체 프레임들을 활성화시켜 획득된 제스쳐 영역에 부합되는 객체 프레임을 할당하는 역할을 수행한다. STM에는 LTM으로부터 읽어 온 객체 프레임과 행동 프레임, 그리고 스키마들뿐만 아니라 인식 모듈에 의해서 분석되어진 결과를 스키마의 형태로 저장하여 다음 시점에 사용하도록 한다. 학습 모듈(Learning Module)은 조명이나 사용자 등이 계속적으로 변화하는 환경에 대한 지식을 사용자와의 상호작용을 통하여 학습하고 LTM에 저장한다. 학습된 지식을 인식 모듈에서 활성화되는 프레임들에 반영함으로써 변화하는 환경에서도 제스쳐 인식 시스템의 성능 저하를 방지할 수 있다. 학습 모듈에서 획득된 작업 환경에 대한 지식은 LTM에 저장된다. LTM에는 이과 함께 미리 구축된 객체 프레임들과 스키마들이 저장되어 있다. LTM은 인식 모듈이나 학습 모듈에서 필요로 하는 지식들을 제공해 주는 인간의 두뇌와 같은 역할을 수행한다.

III. 프레임 지식의 표현

제스쳐 인식 시스템에서 사용하는 프레임 지식은 학습 모듈에서 학습된 지식과 미리 구축된 프레임 지식으로 구분된다. 학습된 지식은 현재의 환경에 맞는 손의 색상에 대한 값 또는 조명도에 대한 값 같은 시스템의 환경에 대한 정보들로 구성되어 있다[5]. 미리 구축된 프레임 지식은 제스쳐를 인식하고 인식 대상이 되는 물체에 대한 정보를 갖고 있다. 프레임 지식은 스키마, 객체 프레임, 행동 프레임으로 나눈다.

3.1 스키마(Schema)

스키마는 하나의 동적인 또는 정적인 제스쳐에 대한 지식을 표현하며, 객체 프레임과 행동 프레임이 순차적으로 연결된 구조를 갖는다. 스키마는 시스템에 미리 구축된 정적 스키마(static schema)와 시스템의 인식 과정에서 작성되는 동적 스키마(dynamic schema)의 두 종류가 있다. 정적 스키마는 시스템이 저장하고 있는 지식으로써 LTM에 저장되어 있다가 필요한 경우에 STM에 읽혀져 사용된다. 동적 스키마는 STM에 저장되며 현재 시스템이 인식하고 있는 제스쳐에 대한 정보를 유지한다. 최종적인 인식 결과는 동적 스키마를 STM에 활성화되어 있는 정적 스키마와 비교하여 결정한다. (그림 2)에서는 스키마의 기본 구조를 보여 준다.

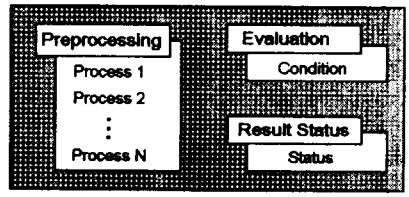


(그림 2) 스키마의 예

정적 스키마는 인식 결과를 얻기 위해 사용될 뿐만 아니라 입력 영상으로부터 획득된 손 영역과 STM에 있는 객체 프레임을 매칭할 경우에도 사용된다. 이상적으로는 손 영역과 LTM에 존재하는 모든 객체 프레임은 매칭을 시도하여 그 중에서 가장 유사한 객체 프레임을 선택하여야 하지만, LTM에는 많은 객체 프레임 존재하기 때문에 그러한 방법은 비효율적이다. 이런 단점을 극복하기 위해 객체 프레임에 매칭에 대한 우선순위를 정하여 가장 우선순위가 높은 객체 프레임부터 매칭을 시도하여 매칭 결과가 미리 정의된 임계값 이상이면 해당 객체 프레임을 선택한다.

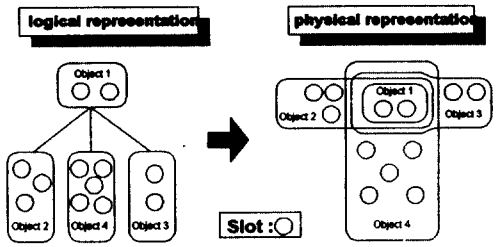
3.2 객체 프레임 (Object Frame)

인식하고자 하는 물체의 특징을 표현하는 지식으로써 객체 프레임이 사용된다. 객체 프레임은 슬롯(Slot)들의 집합으로 구성되어 계층성을 갖는다. 그리고 하나의 슬롯은 여러 개의 객체 프레임에 포함될 수 있다. 즉, 슬롯이 객체 프레임과 일대일의 관계를 갖는 것이 아니라 슬롯 자체가 하나의 지식원으로써 독립적으로 존재한다는 것을 의미한다. 슬롯은 (그림 3)과 같이 전처리, 평가, 결과 상태로 구성된다. 슬롯이 활성화되면 전처리(Preprocessing)에 정의되어 있는 프로세스들을 수행한다. 수행된 결과는 평가(Evaluation)에 정의되어 있는 조건(Condition)과 비교하여 해당 슬롯에 대한 만족도를 구한다. 비교 결과는 결과 상태(Result Status)에 저장되어 같은 슬롯이 반복적으로 활성화될 경우에 다시 전처리와 평가를 수행하지 않고 상태값을 이용하여 만족도를 얻는다.



(그림 3) 슬롯의 구조

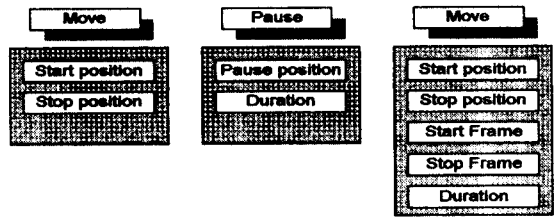
객체 프레임의 계층구조는 슬롯이 각 객체 프레임에 독립적이라는 특성에 의하여 비명시적으로 구축된다. 슬롯은 객체 프레임에 포함 관계를 갖고 있지만 독립적이고, 상위 계층에서 객체 프레임을 활성화한 경우, 또는 하위 계층에서 활성화한 경우에도 슬롯은 단 한번만 평가를 수행하고 평가된 결과를 저장하여 계층구조를 명시적으로 표현할 필요가 없다. (그림 4)는 이러한 특성을 갖는 객체 프레임의 계층구조에 대한 예를 보인다.



(그림 4) 객체 프레임의 계층구조

3.3 행동 프레임 (Behavior Frame)

동적인 제스쳐를 인식하기 위해서는 객체 프레임이 전 시점에서 현 시점까지 어떻게 행동하였는 가를 인식해야 한다. 이러한 객체 프레임의 행동을 인식하기 위해 행동 프레임을 사용한다. 행동 프레임은 전 시점에서부터 현 시점까지의 객체 프레임들간의 관계를 표현하는 지식으로써 이동, 정지, 변환의 3가지 기본 틀을 갖고 있다. 계획기에서는 전 시점에서 인식된 객체 프레임과 현 시점에서 추출된 손 영역이 얼마나 유사한가를 계산하여 현 시점의 행동 프레임에 3가지 기본 틀 중 하나를 할당하게 된다. 유사성이 높고 손 영역의 위치가 변한 경우에는 이동(Move)을 할당하고 유사성이 높고 손 영역의 위치가 변하지 않은 경우에는 정지(Pause)를 할당하게 된다. 그러나 유사성이 낮은 경우에는 변환(Change)을 할당한다. 행동 프레임은 시간에 매우 종속적이므로 일정한 간격의 시간을 하나의 시간 단위로 정의하여 사용한다. 하나의 시간 단위 동안에 위치가 변한 경우는 객체 프레임이 움직인 것으로 인식하고 하나의 시간 단위 동안 위치가 변하지 않은 경우는 객체 프레임이 움직이지 않은 것을 인식한다. 적당한 시간 단위를 정의함으로써 행동 프레임의 이동과 정지를 구별한다. 행동 프레임의 구조는 (그림 5)와 같다.



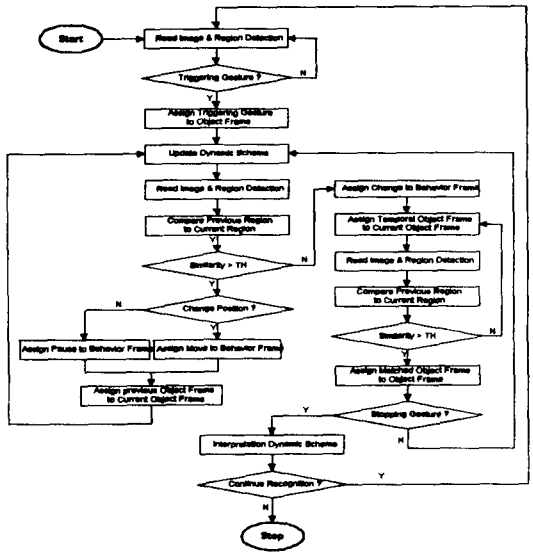
(그림 5) 행동 프레임의 구조

IV. 프레임 지식의 제어 구조

시스템은 각 시점에서 발생하는 객체 프레임과 행동 프레임을 인식하여 동적 스키마를 작성하고 LTM에 저장된 정적 스키마의 매칭을 통하여 동적인 제스쳐를 인식한다. 제스쳐의 효율적인 인식을 위하여 시스템은 사용자와의 상호작용을 통하여 시작 동작을 인식함으로써 제스쳐의 인식을 시작한다. 시간에 따라서 동적으로 변하는 제스쳐를 시작 제스쳐를 사용하지 않고 임의의 시점에서 인식하려고 하면 시스템은 항상 모든 객체 프레임에 대하여 매칭을 수행하여야 한다. 이러한 방법은 매우 비효율적이어서 시스템의 성능을 저하시키므

로 이를 해결하기 위해서는 제스쳐의 시작을 사용자가 시스템에게 알려 주어야 한다. 시작 제스쳐(triggering gesture)는 주먹을 쥔 상태로 정의한다. 모든 제스쳐는 주먹을 쥔 상태에서 시작하여 주먹을 쥔 상태로 끝나게 된다. 동적 제스쳐의 인식은 (그림 6)과 같은 방법으로 수행된다.

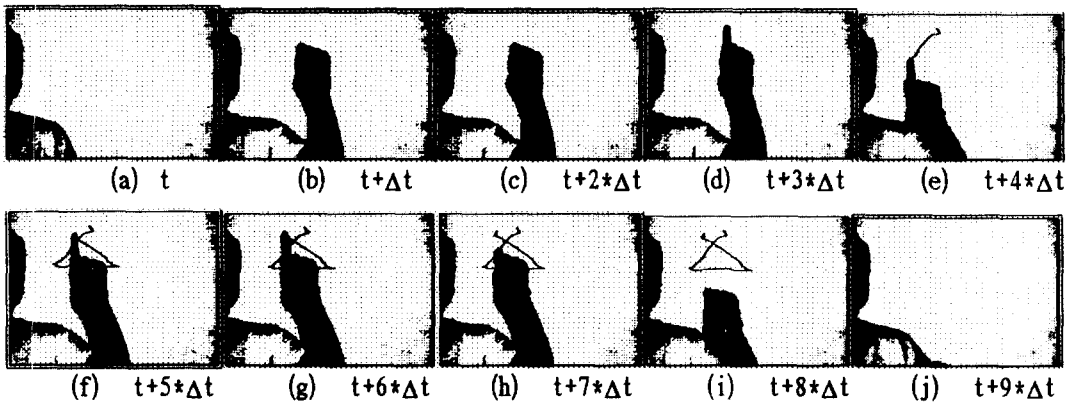
시스템이 초기화되면 카메라로부터 영상을 획득하여 손 영역을 분리한다. 그리고 분리된 영역이 시작 제스쳐인가를 판별하여 시작 제스쳐가 아니면 다음 영상을 획득하여 같은 작업을 반복한다. 시작 제스쳐이면 객체 프레임에 시작 제스쳐를 할당하고 동적 스키마를 생성한다. 다음 단계는 영상을 입력받아 손 영역을 분리하고 전 시점의 영역과 비교하여 적절한 행동 프레임을 할당한다. 유사성이 임계값이상인 경우에 손 영역의 위치를 판별하여 위치가 변하지 않았으면 정지를 행동 프레임에 할당하고 그렇지 않으면 이동을 행동 프레임에 할당한다. 행동 프레임이 정지나 이동인 경우에는 전 시점의 객체 프레임을 현 시점의 객체 프레임으로 할당하고 동적 스키마를 갱신한다. 그리고 다음 영상을 입력받아 같은 작업을 반복한다. 그러나 유사성이 임계값이하인 경우에는 객체 프레임이 변하였으므로 행동 프레임에 변환을 할당하고 현재의 객체 프레임은 임시 객체 프레임(temporary object frame)으로 할당하여 다음 단계를 수행한다. 변환인 경우에는 다음 영상을 입력받아 분리된 손 영역에 해당하는 객체 프레임을 찾을 때까지 같은 작업을 반복한다. 손 영역과 매칭되는 객체 프레임을 찾으면 해당 객체 프레임을 현재 객체 프레임으로 할당하고 마침 제스쳐(stopping gesture)인가를 판별한다. 마침 제스쳐가 아닌 경우에는 동적 스키마를 갱신하고 다음 단계를 수행한다. 그러나 현재의 객체 프레임이 마침 제스쳐인 경우에는 LTM에서 STM으로 읽혀진 정적 제스쳐와 작성된 동적 제스쳐를 분석하여 동적 제스쳐를 인식한 결과를 사용자에게 알려 준다. 그리고 계속 제스쳐 인식을 수행하기 위해 STM을 초기화하여 다른 동적 제스쳐를 인식하기 위한 준비를 한다.



(그림 6) 프레임 지식의 제어 흐름도

V. 실험 결과 및 결론

제안한 프레임 지식의 유용성을 보이기 위해 손가락을 추적하는 시스템을 구현하였다. 즉, 마우스를 대신하여 손가락을 이용하여 커서의 위치를 변경시키거나 마우스의 클릭과 같은 의미의 선택 제스쳐를 인식하여 실제 마우스를 사용하지 않고 손가락을 움직임으로써 마우스를 사용하는 것과 같은 효과를 얻을 수 있다. 실험은 DOS환경에서 수행하였다. 컴퓨터는 PC 486 DX/2 50를, 언어는 C언어를, 컴파일러는 WATCOM C/C++를 사용하였다. 실험 영상의 획득 방법은 비디오 카메라를 사용하여 사용자가 행하는 제스쳐를 촬영하여 초당 30 프레임씩 실험 영상을 획득하였다. 하나의 제스쳐를 동작하는데는 약 10초에서 20초 정도 소요되었다. (그림 7)은 시스템이 손가락을 추적하는 장면을 보여 준다. 실험 결과로는 손가락의 움직임에 따라서 손가락을 추적한 궤적을 보여 준다. 현재 이 시스템은 몇 가지의 제스쳐만을 인식하고 있지만 제안된 지식 구조를 사용하면서 객체를 구별할 수 있는 좋은 특징들을 사용한다면 더욱 다양한 제스쳐도 인식이 가능할 것으로 기대된다.



(그림 7) 실험 결과 영상

참고문헌

[1] Yoshinori Kuno, Kang Hyun Jo, "Human-Ceneterd Human-Computer Interface using Multiple View Invariants," Inter. Workshop on Automatic Face- and Gesture-Recognition, pp. 266-271, Zurich, 1995.

[2] A. Azarbayejani, T. Starner, B. Horowitz, and A. Pentland, "Visually Controlled Graphics," PAMI, Vol. 15, No. 6, pp. 602-605, June, 1993.

[3] A. Rao and R. Jain, "Knowledge Representation and Control in Computer Vision Systems," IEEE Expert, pp. 64-79, Spring, 1988.

[4] Hwang V. S., L. S. Davis and T. Matshyyama, "Hypothesis Intergration in Image Understanding System," Computer Vision Graphics and Image Processing, Vol. 36, pp. 321-371, 1986.

[5] 고일주, 이양원, 이근수, 최형일, "컴퓨터 시작에 기반한 제스쳐 인식," 한국정보처리학회 95' 추계 학술 발표논문집, pp. 604-609, 1995.