



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2016년09월22일  
(11) 등록번호 10-1658937  
(24) 등록일자 2016년09월13일

(51) 국제특허분류(Int. Cl.)  
G06F 3/01 (2006.01) G06F 3/03 (2006.01)  
(21) 출원번호 10-2011-7028423  
(22) 출원일자(국제) 2010년05월29일  
심사청구일자 2015년04월24일  
(85) 번역문제출일자 2011년11월28일  
(65) 공개번호 10-2012-0026072  
(43) 공개일자 2012년03월16일  
(86) 국제출원번호 PCT/US2010/036774  
(87) 국제공개번호 WO 2010/138952  
국제공개일자 2010년12월02일  
(30) 우선권주장  
12/474,781 2009년05월29일 미국(US)  
(56) 선행기술조사문헌  
US20070176898 A1  
US20030156756 A1  
US20050210419 A1

(73) 특허권자  
마이크로소프트 테크놀로지 라이선싱, 엘엘씨  
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원  
마이크로소프트 웨이  
(72) 발명자  
라타 스티븐  
미국 워싱턴주 98052-6399 레드몬드 원 마이크로  
소프트 웨이 엘씨에이 - 인터내셔널 페이턴즈 마  
이크로소프트 코포레이션  
가이스너 케빈  
미국 워싱턴주 98052-6399 레드몬드 원 마이크로  
소프트 웨이 엘씨에이 - 인터내셔널 페이턴즈 마  
이크로소프트 코포레이션  
(뒷면에 계속)  
(74) 대리인  
김태홍

전체 청구항 수 : 총 20 항

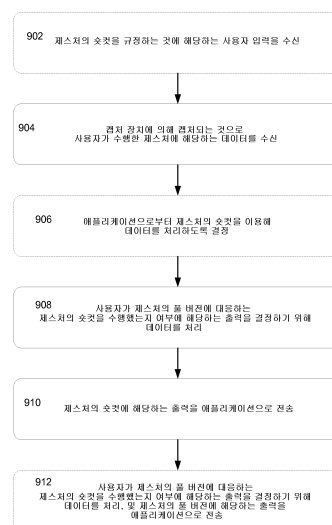
심사관 : 박인화

(54) 발명의 명칭 제스처 슛킷

(57) 요약

제스처 슛킷을 위한 시스템, 방법 및 컴퓨터 판독가능 매체가 개시된다. 사용자의 움직임이나 신체 자세가 시스템의 캡처 장치에 의해 캡처되어 시스템을 제공하기 위한 입력으로서 사용된다. 시스템 인식된 제스처에 대해, 그 제스처의 풀 버전 및 슛킷이 있을 수 있다. 시스템이 제스처의 풀 버전이나 제스처의 슛킷이 수행되었다고 인식하는 경우, 시스템은 시스템 인식 제스처가 관찰되었다는 표시를 애플리케이션으로 보낸다. 슛킷이 제스처의 풀 버전의 부분집합을 포함하고, 사용자가 제스처의 풀 버전을 수행할 때 제스처의 슛킷 및 풀 버전 모두 인식될 경우, 시스템은 하나의 제스처 수행만 발생했다고 인식하고 그러한 것을 애플리케이션으로 알린다.

대표도



(72) 발명자

**클라빈 존**

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 페이턴츠 마이크로소프트 코포레이션

**츠노다 쿠도**

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 페이턴츠 마이크로소프트 코포레이션

**페레즈 캐서린 스톤**

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 페이턴츠 마이크로소프트 코포레이션

**마코빅 펠자**

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 페이턴츠 마이크로소프트 코포레이션

**스눅 그레고리 엔**

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 페이턴츠 마이크로소프트 코포레이션

**킵맨 알렉스**

미국 워싱턴주 98052-6399 레드몬드 원 마이크로소프트 웨이 엘씨에이 - 인터내셔널 페이턴츠 마이크로소프트 코포레이션

## 명세서

### 청구범위

#### 청구항 1

사용자 제스처를 애플리케이션에 대한 입력으로서 채택하는 시스템에서 제스처 숏컷(gesture shortcut)을 이용하는 방법에 있어서,

캡처 장치에 의해 캡처된 데이터 - 상기 데이터는 사용자 모션이나 포즈에 해당함 - 를 수신하는 단계;

상기 데이터에 기초하여, 상기 사용자가 모션이나 포즈는 상기 사용자가 풀 제스처의 숏컷 - 상기 애플리케이션에 대한 입력을 식별하는 상기 풀 제스처의 숏컷의 수행은 또한 상기 풀 제스처의 수행에 의해 식별되고, 상기 풀 제스처의 숏컷은 상기 풀 제스처를 포함하는 사용자 모션이나 포즈의 부분집합을 포함함 - 을 수행하는 것에 해당하는 것을 식별하는 제1 출력을 결정하는 단계;

상기 제1 출력을 상기 애플리케이션으로 보내는 단계;

상기 데이터에 기초하여, 상기 사용자가 상기 풀 제스처를 수행하는 것에 해당하는 상기 사용자 모션이나 포즈를 식별하는 제2 출력을 결정하는 단계;

상기 제2 출력을 상기 애플리케이션으로 보내는 단계; 및

상기 제1 출력 및 제2 출력을 상기 애플리케이션에 대한 동일한 의도된 제스처 입력을 식별하는 것으로서 인식하는 단계를 포함하고,

적어도 상기 풀 제스처의 숏컷과 상기 풀 제스처를 미리 결정된 시간 안에 상기 애플리케이션에 대한 동일한 의도된 제스처 입력을 식별하는 것으로서 인식하는 것에 기초하여, 상기 애플리케이션은 상기 애플리케이션에 대한 의도된 제스처 입력으로서 상기 풀 제스처의 숏컷 또는 상기 풀 제스처 중 하나만을 이용하는 것인, 제스처 숏컷을 이용하는 방법.

#### 청구항 2

제1항에 있어서,

상기 풀 제스처의 숏컷을 수행하는 데 이용되는 사용자 모션 또는 포즈를 규정하는 것에 해당하는 사용자 입력을 수신하는 단계를 더 포함하는, 제스처 숏컷을 이용하는 방법.

#### 청구항 3

제1항에 있어서,

상기 풀 제스처의 숏컷은 상기 애플리케이션에 대한 제2 입력을 식별하는 제2 제스처에 해당하고, 상기 제1 출력은 상기 애플리케이션에 대한 상기 제2 입력이 제공되었음을 식별하는 것인, 제스처 숏컷을 이용하는 방법.

#### 청구항 4

제1항에 있어서,

상기 풀 제스처의 숏컷은 제스처 필터 및 파라미터에 해당하고, 상기 풀 제스처는 상기 제스처 필터 및 상기 파라미터에 해당하며, 상기 파라미터는 상기 풀 제스처의 숏컷에 대하여 상기 풀 제스처에 대한 것과 다른 값을 가지며,

상기 데이터에 기초하여, 상기 사용자가 모션이나 포즈는 상기 사용자가 풀 제스처의 숏컷을 수행하는 것에 해당하는지 여부를 식별하는 제1 출력을 결정하는 단계는, 상기 제스처 필터 및 파라미터를 이용하여 상기 데이터를 처리하는 단계를 더 포함하는 것인, 제스처 숏컷을 이용하는 방법.

#### 청구항 5

제1항에 있어서,

상기 데이터에 기초하여, 상기 사용자 모션이나 포즈는 상기 사용자가 풀 제스처의 슛컷을 수행하는 것에 해당하는지 여부를 식별하는 제1 출력을 결정하는 단계는,

상기 데이터를 처리하여 상기 제1 출력을 결정하기 전에, 상기 풀 제스처의 슛컷이 수행되었는지 여부를 결정하기 위해, 상기 애플리케이션으로부터 상기 데이터를 처리하게 결정하는 단계를 더 포함하는 것인, 제스처 슛컷을 이용하는 방법.

#### 청구항 6

제1항에 있어서,

상기 데이터에 기초하여, 상기 사용자 모션이나 포즈는 상기 사용자가 풀 제스처의 슛컷을 수행하는 것에 해당하는지 여부를 식별하는 제1 출력을 결정하는 단계는,

적어도 사용자로부터 제스처 슛컷들이 인식되어야한다는 표시를 수신한 것에 더 기초하여, 상기 데이터에 기초하여 상기 사용자 모션이나 포즈는 상기 사용자가 풀 제스처의 슛컷을 수행하는 것에 해당하는지 여부를 식별하는 제1 출력을 결정하는 단계를 더 포함하는 것인, 제스처 슛컷을 이용하는 방법.

#### 청구항 7

제1항에 있어서,

상기 제1 출력은 상기 애플리케이션에 대한 입력을 나타내고, 상기 애플리케이션은 상기 애플리케이션에 대한 입력을 수신하는 것을 스코어 또는 성취와 관련시키고,

상기 제1 출력을 상기 애플리케이션으로 보내는 단계는,

상기 애플리케이션은 상기 애플리케이션에 대한 입력이 상기 풀 제스처의 수행을 통해 제공되었을 때보다 더 작은 스코어 또는 성취를 상기 제1 출력과 관련시키고, 상기 애플리케이션으로 상기 제1 출력을 보내는 단계를 더 포함하는 것인, 제스처 슛컷을 이용하는 방법.

#### 청구항 8

사용자 제스처를 애플리케이션에 대한 입력으로서 채택하는 시스템에서 제스처 슛컷의 이용에 관한 컴퓨터 판독 가능한 명령어들을 저장하는 컴퓨터 판독가능한 저장 장치로서,

상기 컴퓨터 판독가능한 명령어들은, 적어도 프로세서 상에서 실행되는 것에 기초하여, 상기 프로세서로 하여금 동작들을 수행하게 하며, 상기 동작들은

캡처 장치에 의해 캡처된 데이터 - 상기 데이터는 사용자 모션이나 포즈에 해당함 - 를 수신하고;

상기 데이터에 기초하여, 상기 사용자 모션이나 포즈는 상기 사용자가 풀 제스처의 슛컷 - 상기 애플리케이션에 대한 입력을 식별하는 상기 풀 제스처의 슛컷의 수행은 또한 상기 풀 제스처의 수행에 의해 식별되고, 상기 풀 제스처의 슛컷은 상기 풀 제스처를 포함하는 사용자 모션이나 포즈의 부분집합을 포함함 - 을 수행하는 것에 해당하는 것을 식별하는 제1 출력을 결정하고;

상기 제1 출력을 상기 애플리케이션으로 보내고;

상기 데이터에 기초하여, 상기 사용자가 상기 풀 제스처를 수행하는 것에 해당하는 상기 사용자 모션이나 포즈를 식별하는 제2 출력을 결정하고;

상기 제2 출력을 상기 애플리케이션으로 보내고; 및

상기 제1 출력 및 제2 출력을 상기 애플리케이션에 대한 동일한 의도된 제스처 입력을 식별하는 것으로서 인식하는 것을 포함하고,

적어도 상기 풀 제스처의 슛컷과 상기 풀 제스처를 미리 결정된 시간 안에 상기 애플리케이션에 대한 동일한 의도된 제스처 입력을 식별하는 것으로서 인식하는 것에 기초하여, 상기 애플리케이션은 상기 애플리케이션에 대한 의도된 제스처 입력으로서 상기 풀 제스처의 슛컷 또는 상기 풀 제스처 중 하나만을 이용하는 것인, 컴퓨터 판독가능한 저장 장치.

## 청구항 9

사용자 제스처를 애플리케이션에 대한 입력으로서 채택하는 시스템에서 제스처 슛킷을 이용하는 시스템에 있어서,

프로세서; 및

상기 시스템이 동작 중일 때 상기 프로세서에 통신 가능하게 연결된 메모리를 포함하고,

상기 메모리는 적어도 상기 프로세서 상에서 실행되는 것에 기초하여 상기 시스템으로 하여금 적어도,

캡처 장치에 의해 캡처된 데이터 - 상기 데이터는 사용자 모션이나 포즈에 해당함 - 를 수신하고;

상기 데이터에 기초하여, 상기 사용자 모션이나 포즈는 상기 사용자가 풀 제스처의 슛킷 - 상기 애플리케이션에 대한 입력을 식별하는 상기 풀 제스처의 슛킷의 수행은 또한 상기 풀 제스처의 수행에 의해 식별되고, 상기 풀 제스처의 슛킷은 상기 풀 제스처를 포함하는 사용자 모션이나 포즈의 부분집합을 포함함 - 을 수행하는 것에 해당하는 것을 식별하는 제1 출력을 결정하고;

상기 제1 출력을 상기 애플리케이션으로 보내고;

상기 데이터에 기초하여, 상기 사용자가 상기 풀 제스처를 수행하는 것에 해당하는 상기 사용자 모션이나 포즈를 식별하는 제2 출력을 결정하고;

상기 제2 출력을 상기 애플리케이션으로 보내고;

상기 제1 출력 및 제2 출력을 상기 애플리케이션에 대한 동일한 의도된 제스처 입력을 식별하는 것으로서 인식하고; 및

상기 풀 제스처의 슛킷과 상기 풀 제스처를 미리 결정된 시간 안에 상기 애플리케이션에 대한 동일한 의도된 제스처 입력을 식별하는 것으로서 인식하는 것에 적어도 기초하여, 상기 애플리케이션에 대한 의도된 제스처 입력으로서 상기 풀 제스처의 슛킷 또는 상기 풀 제스처 중 하나만을 이용하게 하는, 프로세서 실행가능한 명령어들을 저장하는 것인, 제스처 슛킷을 이용하는 시스템.

## 청구항 10

제9항에 있어서,

상기 메모리는 적어도 상기 프로세서 상에서 실행되는 것에 기초하여 상기 시스템으로 하여금 적어도,

상기 풀 제스처의 슛킷을 수행하는 데 이용되는 사용자 모션 또는 포즈를 규정하는 것에 해당하는 사용자 입력을 수신하게 하는, 프로세서 실행가능한 명령어들을 더 저장하는 것인, 제스처 슛킷을 이용하는 시스템.

## 청구항 11

제9항에 있어서,

상기 풀 제스처의 슛킷은 상기 애플리케이션에 대한 제2 입력을 식별하는 제2 제스처에 해당하고, 상기 제1 출력은 상기 애플리케이션에 대한 상기 제2 입력이 제공되었음을 식별하는 것인, 제스처 슛킷을 이용하는 시스템.

## 청구항 12

제9항에 있어서,

상기 제1 출력은 상기 애플리케이션에 대한 입력을 나타내고, 상기 애플리케이션은 상기 애플리케이션에 대한 입력을 수신하는 것을 스코어 또는 성취와 관련시키고, 적어도 상기 프로세서 상에서 실행되는 것에 기초하여 상기 시스템으로 하여금 적어도 상기 제1 출력을 상기 애플리케이션으로 보내게 하는 상기 명령어들은 상기 시스템으로 하여금 적어도,

상기 애플리케이션은 상기 애플리케이션에 대한 입력이 상기 풀 제스처의 수행을 통해 제공되었을 때보다 더 작은 스코어 또는 성취를 상기 제1 출력과 관련시키고, 상기 애플리케이션으로 상기 제1 출력을 보내게 하는 것인, 제스처 슛킷을 이용하는 시스템.

### 청구항 13

제8항에 있어서,

적어도 상기 프로세서 상에서 실행되는 것에 기초하여, 상기 프로세서로 하여금 상기 풀 제스처의 슛컷을 수행하는 데 이용되는 사용자 모션 또는 포즈를 규정하는 것에 해당하는 사용자 입력을 수신하는 것을 포함하는 동작들을 수행하게 하는 컴퓨터 판독가능한 명령어들을 더 포함하는, 컴퓨터 판독가능한 저장 장치.

### 청구항 14

제8항에 있어서,

상기 풀 제스처의 슛컷은 상기 애플리케이션에 대한 제2 입력을 식별하는 제2 제스처에 해당하고, 제1 출력은 상기 애플리케이션에 대한 상기 제2 입력이 제공되었음을 식별하는 것인, 컴퓨터 판독가능한 저장 장치.

### 청구항 15

제1항에 있어서,

상기 제1 출력은 상기 사용자 모션 또는 포즈가 상기 풀 제스처의 슛컷을 수행하는 상기 사용자에게 해당할 가능성을 식별하는 신뢰 수준을 포함하고,

상기 방법은, 적어도 상기 제1 출력의 수신에 기초하여, 상기 애플리케이션이 상기 애플리케이션에 대한 입력이 제공되었음을 인식하고, 적어도 상기 제2 출력의 수신에 기초하여, 상기 애플리케이션이 상기 제2 출력에 기초하여 상기 애플리케이션에 대해 제공된 입력에 세부사항을 추가하는 것을 더 포함하는 것인, 제스처 슛컷을 이용하는 방법.

### 청구항 16

제15항에 있어서,

상기 제2 출력은 상기 사용자가 신체 부분을 이동한 거리 또는 상기 신체 부분의 위치를 표시하는 것인, 제스처 슛컷을 이용하는 방법.

### 청구항 17

제15항에 있어서,

적어도 상기 풀 제스처의 수행 없이 상기 사용자가 상기 풀 제스처의 슛컷을 수행한 것을 결정하는 것에 기초하여, 상기 사용자가 상기 풀 제스처의 슛컷을 수행한 것을 나타내는 제3 출력을 상기 애플리케이션으로 보내는 단계를 더 포함하고,

상기 애플리케이션은 상기 애플리케이션에 대한 입력이 제공되었음을 인식하고, 상기 사용자가 상기 풀 제스처를 수행하지 않은 것을 결정하는 것에 기초하여, 상기 애플리케이션은 상기 애플리케이션에 대한 입력을 제공하기 위해 디폴트 값을 이용하는 것인, 제스처 슛컷을 이용하는 방법.

### 청구항 18

제8항에 있어서,

제1 출력은 상기 애플리케이션에 대한 입력을 제공하는 것을 나타내고, 상기 애플리케이션은 상기 애플리케이션에 대한 입력을 제공하는 것을 스코어 또는 성취와 관련시키고, 상기 제1 출력을 상기 애플리케이션으로 보내는 것은,

상기 애플리케이션은 상기 애플리케이션에 대한 입력이 상기 풀 제스처의 수행을 통해 제공되었을 때보다 더 작은 스코어 또는 성취를 상기 제1 출력과 관련시키고, 상기 애플리케이션으로 상기 제1 출력을 보내는 것을 포함하는, 컴퓨터 판독가능한 저장 장치.

### 청구항 19

제8항에 있어서,

상기 제1 출력은 상기 사용자 모션 또는 포즈가 상기 풀 제스처의 슛컷을 수행하는 상기 사용자에게 해당할 가능성을 식별하는 신뢰 수준을 포함하고,

적어도 상기 프로세서 상에서 실행되는 것에 기초하여, 상기 프로세서로 하여금,

적어도 상기 제1 출력의 수신에 기초하여, 상기 애플리케이션은 상기 애플리케이션에 대한 입력이 수신되었음을 인식하고,

적어도 상기 제2 출력의 수신에 기초하여, 상기 애플리케이션은 상기 제2 출력에 기초하여 상기 애플리케이션에 대해 제공된 입력에 세부사항을 추가하는 단계들을 수행하게 하는 컴퓨터 판독가능한 명령어들을 더 포함하는, 컴퓨터 판독가능한 저장 장치.

## 청구항 20

제9항에 있어서,

상기 제1 출력은 상기 사용자 모션 또는 포즈가 상기 풀 제스처의 슛컷을 수행하는 상기 사용자에게 해당할 가능성을 식별하는 신뢰 수준을 포함하고,

적어도 상기 제1 출력의 수신에 기초하여, 상기 애플리케이션은 상기 애플리케이션에 대한 입력이 수신되었음을 인식하고, 적어도 상기 제2 출력의 수신에 기초하여, 상기 애플리케이션은 상기 제2 출력에 기초하여 상기 애플리케이션에 대해 제공된 입력에 세부사항을 추가하는 것인, 제스처 슛컷을 이용하는 시스템.

## 발명의 설명

### 배경 기술

[0001]

컴퓨터 게임, 멀티미디어 애플리케이션, 오피스 애플리케이션 등과 같은 많은 컴퓨팅 애플리케이션들은 사용자가 게임 캐릭터나 다른 애플리케이션 양태들을 조작할 있게 해주는 컨트롤들을 사용한다. 통상적으로, 그러한 컨트롤은 예를 들어 제어기, 리모컨, 키보드, 마우스 등을 이용해 입력된다. 공교롭게도 그러한 컨트롤은 습득하기가 어려울 수 있으며, 그로 인해 사용자와 그 게임 및 애플리케이션 사이에 장벽을 만들 수 있다. 또한, 그러한 컨트롤은 그들이 사용되는 실제 게임 액션들이나 기타 애플리케이션 액션들과 다를 수 있다. 예를 들어 게임 캐릭터가 야구 배트를 휘두르게 하는 게임 컨트롤은 야구 배트를 휘두르는 실제 모션과 일치하지 않을 수 있다.

### 발명의 내용

#### 과제의 해결 수단

[0002]

제스처 슛컷을 위한 시스템 및 방법이 여기에 개시된다.

[0003]

일 실시예에서, 컴퓨팅 장치는 카메라로부터 일련의 이미지를 수신한다. 그 카메라는 (적색-녹색-청색 또는 RGB 같은) 컬러 카메라, 깊이 인식 카메라(depth camera), 및 삼차원(3D) 카메라를 포함할 수 있다. 그 데이터는 각각의 깊이 및 컬러 이미지, 깊이와 컬러 정보를 합한 결합 이미지, 또는 골격이 매핑되어 있는 사람 같이 대상이 식별되는 과장된 이미지를 포함할 수 있다. 이 데이터는 적어도 한 사용자에게 의해 만들어진 모션이나 포즈를 캡처한다. 그 모션 또는 포즈는 컴퓨팅 장치에 의해 입력의 일종(제스처(gesture))으로 인식될 수 있다. 소정 제스처(가령, 똑바로 걷는 제스처)에 대해, 사용자가 만들 수 있는 제스처의 풀 버전 및 사용자가 만들 수 있는 제스처의 슛컷이 있을 수 있으며, 제스처의 슛컷은 일반적으로 보다 적은 시간과 움직임, 또는 보다 적은 난이도의 사용자 움직임을 요한다.

[0004]

컴퓨팅 장치가 사용자에게 의해 제스처의 슛컷이나 제스처의 풀 버전이 수행되었다는 것을 인식하는 경우, 컴퓨팅 장치는 그것에 대한 표시를, 제스처들을 입력으로 사용하는 애플리케이션으로 보낸다.

[0005]

스�컷이 제스처의 풀 버전의 부분집합을 포함하고, 사용자가 제스처의 풀 버전을 수행할 때 제스처의 슛컷 및 풀 버전 둘 모두가 인식되는 일 실시예에서, 컴퓨팅 장치는 그 제스처에 대해 하나의 행위만이 일어났다고 인식하고 애플리케이션에 그러한 것을 표시한다.

[0006] 상술한 내용은 요약부분이고 따라서 당연히 세부내용의 단순화, 일반화 및 생략을 포함한다. 상기 요약부분은 단지 예시적인 것으로 어떤 식으로든 한정하려는 것이 아님을 알 수 있을 것이다.

### 도면의 간단한 설명

[0007] 이 명세서에 따른 제스처 슛컷을 위한 시스템, 방법, 및 컴퓨터 판독가능 매체는 첨부된 도면을 참조하여 상세히 설명된다.

도 1a 및 1b는 게임을 하는 사용자가 있는 타깃 인식, 분석, 및 추적 시스템의 전형적인 실시예를 도시한다.

도 2는 타깃 인식, 분석, 및 추적 시스템에 사용될 수 있는 캡처 장치의 전형적 실시예를 도시한다.

도 3a는 타깃 인식, 분석, 및 추적 시스템에서 하나 이상의 제스처를 해석하는데 사용될 수 있는 컴퓨팅 환경의 전형적 실시예를 도시한다.

도 3b는 타깃 인식, 분석, 및 추적 시스템에서 하나 이상의 제스처를 해석하는데 사용될 수 있는 컴퓨팅 환경의 또 다른 전형적 실시예를 도시한다.

도 4a는 도 2의 타깃 인식, 분석 및 추적 시스템으로부터 생성된 사용자의 골격 매핑을 예시한다.

도 4b는 도 2에 도시된 제스처 인식기 구조의 세부사항들을 예시한다.

도 5a 및 5b는 보다 복잡한 제스처 필터들을 생성하기 위해 제스처 필터들이 어떻게 스택킹될 수 있는지를 예시한다.

도 6(a), 6(b), 6(c), 6(d) 및 6(e)는 미식축구 비디오 게임에서 사용자가 "페어 캐치(fair catch)"를 신호하기 위해 만들 수 있는 전형적 제스처를 예시한다.

도 7(a), 7(b), 7(c), 7(d), 및 7(e)는 사용자의 골격 맵을 생성하기 위해 이미지 데이터의 각각의 프레임이 과싱되었을 때 도 6(a)-6(e)의 "페어 캐치" 제스처의 예를 도시한다.

도 8(a)는 풀(full) 달리기 제스처를 만들고 있는 사용자를 예시한다.

도 8(b)는 도 8(a)의 풀 달리기 제스처 움직임의 부분집합을 포함하는 슛컷 달리기 제스처를 만드는 사용자를 예시한다.

도 8(c)는 도 8(a)의 풀 달리기 제스처와 별개의 두 번째 유형의 슛컷 달리기 제스처를 만드는 사용자를 예시한다.

도 9는 제스처 슛컷들에 대한 전형적 동작 절차를 묘사한다.

### 발명을 실시하기 위한 구체적인 내용

[0008] 앞으로 여기에 기술되겠지만, 사용자는 하나 이상의 제스처를 수행함으로써 게임 콘솔, 컴퓨터 등과 같은 컴퓨팅 환경에서 실행되는 애플리케이션을 제어할 수 있다. 일 실시예에 따르면, 제스처들은 예컨대 캡처 장치에 의해 수신될 수 있다. 예를 들어 캡처 장치는 장면(scene)의 깊이 이미지를 캡처할 수 있다. 일 실시예에서 캡처 장치는 장면 내 한 개 이상의 타깃이나 오브젝트가 사용자 같은 인간 타깃에 해당하는지 여부를 판단할 수 있다. 장면 내 타깃이나 오브젝트가 인간 타깃에 해당하는지 여부를 판단하기 위해, 타깃들 각각은 플러드 필(flood fill)되어 사람의 신체 모델 패턴과 비교될 수 있다. 그런 다음 사람의 신체 모델에 매치되는 각각의 타깃이나 오브젝트가 그와 관련된 골격 모델을 생성하기 위해 검색될 수 있다. 그리고 나서 컴퓨팅 환경이 골격 모델을 추적하고 골격 모델과 관련된 아바타를 렌더링하고 가령 그 골격 모델로부터 인식된 사용자의 제스처들에 기반하여 컴퓨터 환경에서 실행되는 애플리케이션에서 어떤 컨트롤을 수행할지를 결정할 수 있도록 골격 모델이 컴퓨팅 환경으로 제공될 수 있다. 사용자에게 의해 특정 제스처가 이뤄졌을 때를 판단하기 위해 이하에서 보다 상세히 설명되는 구조를 가진 제스처 인식기 엔진이 사용된다.

[0009] 도 1a 및 1b는 복싱 게임을 하는 사용자(18)가 있는 타깃 인식, 분석, 및 추적 시스템의(10) 전형적인 실시예를 도시한다. 전형적 실시예에서, 타깃 인식, 분석, 및 추적 시스템(10)은 사용자(18) 같은 인간 타깃을 인식, 분석 및/또는 추적하는데 사용될 수 있다.



- [0010] 도 1a에 도시된 바와 같이, 타겟 인식, 분석, 및 추적 시스템(10)은 컴퓨팅 환경(12)을 포함할 수 있다. 컴퓨팅 환경(12)은 컴퓨터, 게임 시스템 또는 콘솔 등일 수 있다. 전형적인 일 실시예에 따르면, 컴퓨팅 환경(12)은 컴퓨팅 환경(12)이 게임 애플리케이션, 비게임 애플리케이션 등과 같은 애플리케이션들을 실행하는데 사용될 수 있도록 하드웨어 및/또는 소프트웨어 컴포넌트를 포함할 수 있다.
- [0011] 도 1a에 도시된 바와 같이, 타겟 인식, 분석, 및 추적 시스템(10)은 캡처 장치(20)를 더 포함할 수 있다. 캡처 장치(20)는 예컨대 사용자(18) 같은 한 명 이상의 사용자들을 시각적으로 모니터링 하는데 사용될 수 있는 카메라일 수 있어, 이하에서 보다 상세히 기술되는 것과 같이 어떤 애플리케이션 안에서 하나 이상의 컨트롤이나 액션을 수행하도록 한 명 이상의 사용자들에 의해 행해진 제스처들이 캡처, 분석, 및 추적될 수 있다.
- [0012] 일 실시예에 따르면, 타겟 인식, 분석, 및 추적 시스템(10)은 텔레비전, 모니터, 고해상도 텔레비전(HDTV) 등과 같이 사용자(18) 같은 사용자에게 게임이나 애플리케이션 비주얼 및/또는 오디오를 제공할 수 있는 시청각 장치(16)에 연결될 수 있다. 예를 들어, 컴퓨팅 환경(12)은 그래픽 카드 같은 비디오 어댑터 및/또는 사운드 카드 같은 오디오 어댑터를 포함하여 게임 애플리케이션, 비게임 애플리케이션 등과 관련된 시청각 신호를 제공할 수 있다. 시청각 장치(16)는 컴퓨팅 환경(12)으로부터 시청각 신호를 수신하고 그런 다음 그 시청각 신호와 관련된 게임 또는 애플리케이션 비주얼 및/또는 오디오를 사용자(18)에게 출력할 수 있다. 일 실시예에 따르면, 시청각 장치(16)는 가령 S-비디오 케이블, 동축 케이블, HDMI 케이블, DVI 케이블, VGA 케이블 등을 통해 컴퓨팅 환경(12)에 연결될 수 있다.
- [0013] 도 1a 및 1b에 도시된 바와 같이, 타겟 인식, 분석, 및 추적 시스템(10)은 사용자(18) 같은 인간 타겟을 인식, 분석 및/또는 추적하는데 사용될 수 있다. 예를 들어, 사용자(18)의 움직임이 컴퓨터 환경(12)에 의해 실행되고 있는 애플리케이션에 영향을 미치는데 사용될 수 있는 컨트롤로서 해석될 수 있도록 캡처 장치(20)를 사용해 사용자(18)가 추적될 수 있다. 따라서, 일 실시예에 따르면 사용자(18)는 애플리케이션을 제어하기 위해 자신의 신체를 움직일 수 있다.
- [0014] 도 1a 및 1b에 도시된 바와 같이, 전형적 실시예에서 컴퓨팅 환경(12)을 실행하는 애플리케이션은 사용자(18)가 플레이하고 있을 수 있는 복싱 게임일 수 있다. 예를 들어, 컴퓨팅 환경(12)은 시청각 장치(16)를 사용해 사용자(18)의 복싱 상대(22)에 대한 시각적 표현을 제공할 수 있다. 컴퓨팅 환경(12)은 또한 시청각 장치(16)를 사용해 사용자(18)가 자신의 움직임을 통해 제어할 수 있는 플레이어 아바타(24)의 시각적 표현을 제공할 수 있다. 예를 들어 도 1b에 도시된 바와 같이, 사용자(18)는 플레이어 아바타(24)가 게임 공간에서 펀치를 날릴 수 있게 물리적 공간에서 펀치를 휘두를 것이다. 따라서, 전형적인 일 실시예에 따라 타겟 인식, 분석, 및 추적 시스템(10)의 컴퓨터 환경(12) 및 캡처 장치(20)는 물리적 공간에서의 사용자(18)의 펀치가 게임 공간에서 플레이어 아바타(24)의 게임 컨트롤로서 해석될 수 있도록 물리적 공간의 펀치를 인식 및 해석하는데 사용될 수 있다.
- [0015] 사용자(18)에 의한 다른 움직임들 역시, 빠르게 움직이거나, 비틀거리거나, 질질끌거나, 막거나, 찰을 날리거나, 다른 다양한 파워 펀치를 날리기 위한 컨트롤들 같은 다른 컨트롤들이나 액션들로서 해석될 수 있다. 또한 일부 움직임들은 플레이어 아바타(24)를 컨트롤하는 것 이외의 액션들에 해당할 수 있는 컨트롤들로서 해석될 수 있다. 예를 들어 플레이어는 게임을 종료하거나 일시 정지하거나 저장하고, 레벨을 선택하고, 높은 스코어를 확인하고, 친구와 통신하는 등의 움직임들을 이용할 수 있다.
- [0016] 전형적 실시예들에서, 사용자(18) 같은 인간 타겟은 오브젝트를 가질 수 있다. 그러한 실시예들에서, 전자 게임 사용자는 오브젝트와 플레이어의 모션이 게임의 파라미터들을 조정 및/또는 제어하는데 사용될 수 있게 오브젝트를 들고 있을 수 있다. 예를 들어, 라켓을 든 플레이어의 모션이 추적되어 전자 스포츠 게임의 스크린 상의 라켓을 제어하는데 활용될 수 있다. 또 다른 전형적 실시예에서는, 오브젝트를 들고 있는 플레이어의 모션이 추적되어 전자 전투 게임에 있어 스크린 상의 무기를 제어하는데 활용될 수 있다.
- [0017] 다른 전형적 실시예들에 따르면, 타겟 인식, 분석, 및 추적 시스템(10)은 타겟 움직임들을 게임 영역 밖에 있는 운영체제 및/또는 애플리케이션 컨트롤들로서 해석하도록 더 사용될 수 있다. 예를 들어, 운영체제 및/또는 애플리케이션의 어떤 가상적으로 제어가능한 양태가 사용자(18) 같은 타겟의 움직임에 의해 제어될 수 있다.
- [0018] 도 2는 타겟 인식, 분석, 및 추적 시스템(10)에 사용될 수 있는 캡처 장치(20)의 전형적 실시예를 도시한다. 전형적 실시예에 따르면, 캡처 장치(20)는 예컨대 TOF(time-of-flight), 구조화된 이미지(structured image), 스테레오 이미지 등을 포함하는 어떤 적절한 기법을 통해, 깊이 값들을 포함할 수 있는 깊이 이미지를 구비한 깊이 정보를 가진 비디오를 캡처하도록 구성될 수 있다. 일 실시예에 따르면, 캡처 장치(20)는 산출된 깊이 정

보를, "Z 계층들", 또는 깊이 인식 카메라로부터 그 시선을 따라 뻗어나온 Z 축에 수직일 수 있는 계층들 안에 조직할 수 있다.

- [0019] 도 2에 도시된 바와 같이, 캡처 장치(20)는 이미지 카메라 구성요소(22)를 포함할 수 있다. 전형적 일 실시예에 따르면 이미지 카메라 구성요소(22)는 어떤 장면의 깊이 이미지를 캡처할 수 있는 깊이 인식 카메라일 수 있다. 깊이 이미지는 캡처된 장면의 이차원(2D) 픽셀 영역을 포함할 수 있고, 여기서 2D 픽셀 영역 내 각 픽셀은 카메라로부터 캡처된 장면 내 어떤 오브젝트에 대한 가령 센티미터, 밀리미터 단위 등의 길이를 나타낼 수 있다.
- [0020] 도 2에 도시된 바와 같이, 전형적인 일 실시예에 따르면, 이미지 카메라 구성요소(22)는 장면의 깊이 이미지를 캡처하는데 사용될 수 있는 IR 광 구성요소(24), 삼차원(3D) 카메라(26), 및 RGB 카메라(28)를 포함할 수 있다. 예를 들어, TOF 분석시, 캡처 장치의 IR 광 구성요소(24)는 장면 위에 적외선 광을 발산하고 그런 다음 센서들(미도시)을 사용하여 가령 3D 카메라(26) 및/또는 RGB 카메라(28)를 이용해 장면 내 한 개 이상의 타겟 및 오브젝트 표면으로부터 후방산란된 광을 검출할 수 있다. 일부 실시예에서, 나가는 광 펄스와 그에 대응하여 들어오는 광 펄스 사이의 시간이 측정되고 캡처 장치(20)로부터 장면 내 타겟이나 오브젝트들 상의 특정 위치까지의 물리적 거리를 판단하는데 사용될 수 있도록 펄스되는(pulsed) 적외선 광이 이용될 수 있다. 또한 다른 전형적 실시예에서, 위상 쉬프트(shift)를 판단하기 위해 나가는 광 파동의 위상이 들어오는 광 파동의 위상과 비교될 수 있다. 그런 다음 위상 쉬프트는 캡처 장치로부터 타겟이나 오브젝트들 상의 특정 위치까지의 물리적 거리를 판단하는데 사용될 수 있다.
- [0021] 또 다른 전형적 실시예에 따르면, 가령 서터링된 광 펄스 이미지를 포함하는 다양한 기법들을 통해 시간에 따른 광의 반사 빔 강도를 분석함으로써, 캡처 장치(20)로부터 타겟이나 오브젝트들 상의 특정 위치까지의 물리적 거리를 간접적으로 판단하는데 TOF 분석이 사용될 수 있다.
- [0022] 또 다른 전형적 실시예에서, 캡처 장치(20)는 깊이 정보를 캡처하기 위해 구조화된 광을 사용할 수 있다. 그러한 분석시, 패턴화된 광(즉, 그리드 패턴이나 스트라이프 패턴 같이 알려진 패턴으로서 디스플레이되는 광)이 가령 IR 광 구성요소(24)를 통해 장면 위로 투사될 수 있다. 장면 내 한 개 이상의 타겟이나 오브젝트의 표면에 닿을 때, 그에 응하여 패턴이 변형될 수 있다. 그러한 패턴의 변형은 가령 3D 카메라(26) 및/또는 RGB 카메라(28)에 의해 캡처될 수 있고 그런 다음 캡처 장치로부터 타겟이나 오브젝트들 상의 특정 위치까지의 물리적 거리를 판단하기 위해 분석될 수 있다.
- [0023] 또 다른 실시예에 따르면, 캡처 장치(20)는 서로 다른 각도로부터 장면을 볼 수 있는 둘 이상의 물리적으로 분리된 카메라들을 포함함으로써, 깊이 정보를 산출하기 위해 판단될 수 있는 비주얼 스테레오 데이터를 획득할 수 있다.
- [0024] 캡처 장치(20)는 마이크로폰(30)을 더 포함할 수 있다. 마이크로폰(30)은 소리를 수신하여 전기 신호로 변환할 수 있는 트랜듀서나 센서를 포함할 수 있다. 일 실시예에 따르면, 마이크로폰(30)은 타겟 인식, 분석, 및 추적 시스템(10)의 캡처 장치(20) 및 컴퓨팅 환경(12) 사이의 피드백을 줄이는 데 사용될 수 있다. 또한, 컴퓨팅 환경(12)에 의해 실행될 수 있는 게임 애플리케이션, 비게임 애플리케이션 등과 같은 애플리케이션들을 제어하기 위해 사용자에게 의해 제공될 수 있는 오디오 신호를 수신하도록 마이크로폰(30)이 사용될 수 있다.
- [0025] 전형적인 일 실시예에서, 캡처 장치(20)는 이미지 카메라 컴포넌트(22)와 유효한 통신상태에 있을 수 있는 프로세서(32)를 더 포함할 수 있다. 프로세서(32)는 깊이 이미지를 수신하고, 깊이 이미지에 적절한 타겟이 포함될 수 있는지 여부를 판단하고, 적절한 타겟을 타겟의 골격 표현이나 모델로 변화하는 명령어들이나 어떤 다른 적절한 명령어를 포함할 수 있는 명령어들을 실행할 수 있는 규격화된 프로세서, 특수형 프로셋, 마이크로프로세서 등을 포함할 수 있다.
- [0026] 캡처 장치(20)는 프로세서(32)에 의해 실행될 수 있는 명령어들, 3D 카메라나 RGB 카메라에 의해 캡처되는 이미지들이나 이미지들의 프레임들, 또는 어떤 다른 적절한 정보, 이미지 등을 저장할 수 있는 메모리 구성요소(34)를 더 포함할 수 있다. 전형적인 일 실시예에 따르면, 메모리 구성요소(34)는 RAM(random access memory), ROM(read only memory), 캐시, 플래시 메모리, 하드 디스크, 또는 어떤 다른 적절한 저장 구성요소를 포함할 수 있다. 도 2에 도시된 바와 같이, 일 실시예에서 메모리 구성요소(34)는 이미지 캡처 구성요소(22) 및 프로세서(32)와 통신하는 별도의 구성요소일 수 있다. 또 다른 실시예에서, 메모리 구성요소(34)는 프로세서(32) 및/또는 이미지 캡처 구성요소(22) 안에 병합될 수 있다.
- [0027] 도 2에 도시된 바와 같이, 캡처 장치(20)는 통신 링크(36)를 통해 컴퓨팅 환경(12)과 통신할 수 있다. 통신 링

크(36)는 가령 USB 접속, FireWire 접속, 이더넷 케이블 접속 등을 포함하는 유선 접속 및/또는 무선 802.11b, g, a, 또는 n 접속 같은 무선 접속일 수 있다. 일 실시예에 따르면, 컴퓨팅 환경(12)은 가령 통신 링크(36)를 통한 장면을 언제 캡처할지를 결정하는데 사용될 수 있는 클록을 캡처 장치(20)로 제공할 수 있다.

[0028] 또한 캡처 장치(20)는 가령 3D 카메라(26) 및/또는 RGB 카메라(28)에 의해 캡처된 깊이 정보 및 이미지들, 및 캡처 장치(20)에 의해 생성될 수 있는 골격 모델을 통신 링크(36)를 통해 컴퓨팅 환경(12)으로 제공할 수 있다. 그러면 컴퓨팅 환경(12)은 골격 모델, 깊이 정보, 및 캡처된 이미지들을 사용하여, 가령 사용자 제스처들을 인식하고 그에 응하여 게임이나 워드 프로세스 같은 애플리케이션을 제어할 수 있다. 예를 들어 도 2에 도시된 바와 같이, 컴퓨팅 환경(12)은 제스처 인식기 엔진(190)을 포함할 수 있다. 제스처 인식기 엔진(190)은 제스처 필터들의 모음을 포함할 수 있으며, 각각의 필터는 (사용자가 움직이면서) 골격 모델에 의해 수행될 수 있는 제스처에 관한 정보를 포함한다. 카메라들(26, 28) 및 장치(20)에 의해 캡처된 골격 모델 형태의 데이터와 그와 관련된 움직임들은 사용자(골격 모델에 의해 표현됨)가 언제 한 가지 이상의 제스처를 수행했는지를 식별하도록 제스처 인식기 엔진(190) 내 제스처 필터들과 비교될 수 있다. 그러한 제스처들은 애플리케이션의 다양한 컨트롤과 관련될 수 있다. 따라서, 컴퓨팅 환경(12)은 제스처 인식기 엔진(190)을 사용해 골격 모델의 움직임을 해석하고 그 움직임에 기초해 애플리케이션을 제어할 수 있다.

[0029] 도 3a는 도 1a-2의 컴퓨팅 환경(12)을 구현하기 위해 사용될 수 있는 컴퓨팅 환경의 전형적 일 실시예를 예시한다. 컴퓨팅 환경은 게임 콘솔 같은 멀티미디어 콘솔(100)일 수 있다. 도 3a에 도시된 바와 같이, 멀티미디어 콘솔(100)은 레벨 1 캐시(102), 레벨 2 캐시(104), 및 플래시 ROM(Read Only Memory)(106)을 가진 중앙 프로세싱 유닛(CPU)(101)을 가진다. 레벨 1 캐시(102) 및 레벨 2 캐시(104)는 임시로 데이터를 저장하고 그에 따라 메모리 액세스 사이클 횟수를 줄임으로써 프로세싱 속도 및 처리율을 향상시킬 수 있다. CPU(101)는 하나를 넘는 코어를 가지는 것으로 주어질 수 있고, 그에 따라 추가적 레벨 1 및 레벨 2 캐시들(102 및 104)을 가질 수 있다. 플래시 ROM(106)은 멀티미디어 콘솔(100)이 파워 온(ON) 될 때 부팅 프로세스의 초기 단계 중에 로드되는 실행가능 코드를 저장할 수 있다.

[0030] 그래픽 프로세싱 유닛(GPU)(108) 및 비디오 인코더/비디오 코덱(코더/디코더)(114)가 고속 고해상도의 그래픽 프로세싱을 위한 비디오 프로세싱 파이프라인을 형성한다. 데이터는 버스를 통해 그래픽 프로세싱 유닛(108)으로부터 비디오 인코더/비디오 코덱(114)으로 전달된다. 비디오 프로세싱 파이프라인은 텔레비전이나 다른 디스플레이로의 전송을 위해 데이터를 A/V(오디오/비디오) 포트(140)로 출력한다. 메모리 제어기(110)가 GPU(108)에 연결되어 비한정적인 것으로서 RAM(Random Access Memory) 같은 다양한 타입의 메모리(112)에 대한 프로세서 액세스를 돕는다.

[0031] 멀티미디어 콘솔(100)은 모듈(118) 상에 바람직하게 구현되는 I/O 제어기(120), 시스템 관리 제어기(122), 오디오 프로세싱 유닛(123), 네트워크 인터페이스 제어기(124), 제1USB 호스트 제어기(126), 제2USB 제어기(128) 및 전면 패널 I/O 하위 어셈블리(130)를 포함한다. USB 제어기들(126 및 128)은 주변 제어기들(142(1)-142(2)), 무선 어댑터(148) 및 외부 메모리 장치(146)(가령, 플래시 메모리, 외부 CD/DVD-ROM 드라이브, 착탈형 매체 등)에 대한 호스트로서의 역할을 한다. 네트워크 인터페이스(124) 및/또는 무선 어댑터(148)는 네트워크(가령, 인터넷, 홈 네트워크 등)로의 액세스를 제공하며, 이더넷 카드, 모뎀, 블루투스 모듈, 케이블 모뎀 등을 포함하는 광범위한 유무선 어댑터 구성요소들 중 어느 하나일 수 있다.

[0032] 시스템 메모리(143)는 부팅 프로세스 중에 로드되는 애플리케이션 데이터를 저장하기 위해 제공된다. 매체 드라이브(144)가 제공되며, DVD/CD 드라이브, 하드 드라이브, 또는 다른 착탈형 매체 드라이브 등을 포함할 수 있다. 매체 드라이브(144)는 멀티미디어 콘솔(100)의 내부나 외부에 있을 수 있다. 애플리케이션 데이터가 멀티미디어 콘솔(100)에 의한 실행, 재생 등을 위해 매체 드라이브(144)를 통해 액세스될 수 있다. 매체 드라이브(144)는 직렬 ATA 버스나 기타 고속 접속(가령, IEEE 1394) 같은 버스를 통해 I/O 제어기(120)에 연결된다.

[0033] 시스템 관리 제어기(122)는 멀티미디어 콘솔(100)의 가용성을 보장하는 것과 관련된 다양한 서비스 기능들을 제공한다. 오디오 프로세싱 유닛(123) 및 오디오 코덱(132)은 높은 충실도를 가지고 스테레오 처리를 하는 해당 오디오 프로세싱 파이프라인을 형성한다. 오디오 데이터는 통신 링크를 통해 오디오 프로세싱 유닛(123) 및 오디오 코덱(132) 사이에서 전송된다. 오디오 프로세싱 파이프라인은 외부 오디오 플레이어나 오디오 사양을 가진 장치에 의한 재생을 위해 데이터를 A/V 포트(140)로 출력한다.

[0034] 전면 패널 I/O 하위 어셈블리(130)는 파워 버튼(150)과 이젝트(eject) 버튼(152) 뿐 아니라 멀티미디어 콘솔(100)의 외부 면 상에 노출되는 어떤 LED(light emitting diodes)나 기타 표시자들의 기능을 지원한다. 시스템 전력 공급 모듈(136)은 멀티미디어 콘솔(100)의 구성요소들로 전력을 제공한다. 팬(138)은 멀티미디어 콘솔

(100) 안의 회로를 냉각시킨다.

- [0035] CPU(101), GPU(108), 메모리 제어기(110), 및 멀티미디어 콘솔(100) 안의 다른 다양한 구성요소들은 다양한 버스 구조들 중 어느 하나를 이용하는 직병렬 버스, 메모리 버스, 주변기기 버스, 및 프로세서나 로컬 버스를 포함하는 한 개 이상의 버스들을 통해 서로 연결된다. 예로서 그러한 구조들로는 PCI(Peripheral Component Interconnects) 버스, PCI-Express 버스 등이 포함될 수 있다.
- [0036] 멀티미디어 콘솔(100)이 파워 온 될 때, 애플리케이션 데이터가 시스템 메모리(143)로부터 메모리(112) 및/또는 캐시들(102, 104)로 로드되고 CPU(101) 상에서 실행된다. 애플리케이션은 멀티미디어 콘솔(100)에서 사용가능한 각종 미디어 유형들을 탐색할 때 일관된 사용자 경험을 제공하는 그래픽 사용자 인터페이스를 제공할 수 있다. 제스처시, 애플리케이션들 및/또는 매체 드라이브(144) 안에 포함되는 기타 매체가 매체 드라이브(144)로부터 시동되거나 재생되어 멀티미디어 콘솔(100)로 추가 기능들을 제공하도록 할 수 있다.
- [0037] 멀티미디어 콘솔(100)은 시스템을 단순히 텔레비전이나 다른 디스플레이로 연결함으로써 단독형 시스템으로서 작동될 수 있다. 이러한 단독 모드에서, 멀티미디어 콘솔(100)은 한 명 이상의 사용자들이 시스템과 상호작용하거나 영화를 보거나 음악을 듣는 것을 가능하게 할 수 있다. 그러나, 네트워크 인터페이스(124)나 무선 어댑터(148)를 통해 이용 가능한 광대역 접속의 통합을 통해, 멀티미디어 콘솔(100)은 더 큰 네트워크 커뮤니티 안의 참여자로서 추가 작동될 수 있다.
- [0038] 멀티미디어 콘솔(100)이 파워 온될 때, 정해진 분량의 하드웨어 자원들이 멀티미디어 콘솔 운영체계에 의한 시스템 사용을 위해 예비된다. 그 자원들은 메모리(가령 16MB), CPU 및 GPU 사이클(가령 5%)의 비축, 네트워크 대역폭(가령 8kbs) 등을 포함할 수 있다. 이러한 자원들은 시스템 부팅 시점에 준비되기 때문에, 예비 자원들은 애플리케이션의 보기로부터 존재하지 않는다.
- [0039] 특히, 메모리 예비량은 시동 커널(launch kernel), 동시적 시스템 애플리케이션들 및 드라이버들을 포함하기 충분할 만큼 많은 것이 바람직하다. 예비된 CPU 사용이 시스템 애플리케이션들에 의해 이용되지 않는 경우 아이들(idle) 스레드가 어떤 미사용 사이클을 소비하도록 CPU 예비는 일정한 것이 바람직하다.
- [0040] GPU 예비와 관련해, 시스템 애플리케이션(가령, 팝업)에 의해 생성된 가벼운 메시지가 팝업을 중첩(overlay)된 것으로 렌더링하는 코드를 스케줄링할 GPU 인터럽트를 사용함으로써 디스플레이된다. 중첩에 필요한 메모리 양은 중첩 영역 사이즈에 좌우되며, 중첩은 스크린 해상도에 따라 축적됨이 바람직하다. 동시발생적 시스템 애플리케이션에 의해 풀(full) 사용자 인터페이스가 사용되는 경우, 애플리케이션 해상도와 무관한 해상도를 사용함이 바람직하다. 주파수를 변경하고 TV 재동기를 유도해야 할 필요성이 없도록 그러한 해상도를 설정하는데 스케일러가 사용될 수 있다.
- [0041] 멀티미디어 콘솔(100)이 부팅되고 시스템 자원들이 예비된 뒤, 시스템 기능을 제공하기 위해 동시발생적 시스템 애플리케이션들이 실행된다. 시스템 기능들은 상술한 예비 시스템 자원들 안에서 실행되는 시스템 애플리케이션들의 집합 내에 캡슐화된다. 운영체제 커널이 시스템 애플리케이션 스레드 대(vs) 게임 애플리케이션 스레드인 스레드들을 식별한다. 시스템 애플리케이션들은 애플리케이션에 일관된 시스템 자원 보기를 제공하기 위해 소정 시점 및 인터벌로 CPU(101) 상에서 실행되도록 스케줄링됨이 바람직하다. 스케줄링은 콘솔 상에서 실행되는 게임 애플리케이션에 대한 캐시 혼란을 최소화해야 한다.
- [0042] 동시적 시스템 애플리케이션이 오디오를 필요로 할 때, 시간에 대한 민감도로 인해 오디오 프로세싱은 게임 애플리케이션과는 비동기적으로 스케줄링된다. 시스템 애플리케이션들이 액티브될 때, 멀티미디어 콘솔 애플리케이션 관리자(이하에 기술됨)가 게임 애플리케이션 오디오 레벨(가령, 묵음(mute), 감쇠(attenuate))을 제어한다.
- [0043] 입력 장치들(가령, 제어기들(141(1) 및 142(2)))이 게임 애플리케이션 및 시스템 애플리케이션들에 의해 공유된다. 입력 장치들은 예비 자원들이 아니지만, 각각이 장치의 포커스를 가지도록 시스템 애플리케이션 및 게임 애플리케이션들 사이에서 스위칭되어야 한다. 애플리케이션 관리자는 게임 애플리케이션의 지식 없이 입력 스트림의 스위칭을 제어함이 바람직하고 드라이버는 포커스 스위치들에 관한 상태 정보를 관리한다. 카메라(26, 28) 및 캡처 장치(20)가 콘솔(100)에 대한 추가 입력 장치들을 규정할 수 있다.
- [0044] 도 3b는 타겟 인식, 분석, 및 추적 시스템에서 한 개 이상의 제스처를 해석하는데 사용되는 도 1a-2에 도시된 컴퓨팅 환경(12)일 수 있는 컴퓨팅 환경(220)의 또 다른 전형적 실시예를 도시한다. 컴퓨팅 시스템 환경(220)은 다만 적절한 컴퓨팅 환경의 예일 뿐이며 현재 여기에 개시된 주요 대상의 사용이나 기능의 범위에 대한 어떠한 한계를 제시하고자 하는 것이 아니다. 컴퓨팅 환경(220)이 전형적 컴퓨팅 환경(220)에 예시된 구성요소들



중 어느 하나나 그 조합에 관해 어떠한 종속성이나 필요조건을 가지는 것으로도 해석되어서는 안될 것이다. 일부 실시예에서 여러 묘사된 컴퓨팅 구성요소들은 본 개시의 특정 양태를 예를 들어 설명하기 위해 구성된 회로를 포함할 수 있다. 이를테면, 이 개시물에서 사용되는 회로라는 용어는 펌웨어나 스위치들에 의해 기능(들)을 수행하도록 구성된 특수화된 하드웨어 구성요소들을 포함할 수 있다. 다른 전형적 실시예에서 회로라는 용어는 기능(들)을 수행하도록 작동되는 로직을 구현하는 소프트웨어 명령어들에 의해 설정되는 범용 프로세싱 유닛, 메모리 등을 포함할 수 있다. 회로가 하드웨어 및 소프트웨어의 조합을 포함하는 전형적 실시예에서, 구현자가 로직을 실체화하는 소스 코드를 작성할 수 있고, 소스 코드는 범용 프로세싱 유닛에 의해 프로세싱될 수 있는 기계 판독가능 코드 안에 컴파일될 수 있다. 최신 기술이 하드웨어, 소프트웨어, 또는 하드웨어/소프트웨어 사이에 차가 거의 없는 수준까지 진화하였다는 것을 알 수 있으므로, 특정 기능을 실시하기 위한 하드웨어 대 소프트웨어의 선택은 구현자에게 맡겨진 디자인 선택사항이다. 더 특정하여 말하면, 당업자는 소프트웨어 프로세스가 균등한 하드웨어 구조로 변환될 수 있고, 하드웨어 구조 자체가 균등한 소프트웨어 프로세스로 변환될 수 있다는 것을 예상할 수 있다. 따라서, 하드웨어 구현 대 소프트웨어 구현의 선택은 디자인 선택사항으로서 구현자에게 맡겨진다.

[0045] 도 3b에서 컴퓨팅 환경(220)은 통상적으로 다양한 컴퓨터 판독가능 매체를 포함하는 컴퓨터(241)를 포함한다. 컴퓨터 판독가능 매체는 컴퓨터(241)에 의해 액세스될 수 있는 어떤 가용 매체일 수 있으며, 휘발성 및 비휘발성 매체 양자 모두와 착탈형 및 비착탈형 매체를 포함한다. 시스템 메모리(222)는 ROM(read only memory)(223) 및 RAM(random access memory)(260) 같은 휘발성 및/또는 비휘발성 메모리의 형식으로 된 컴퓨터 저장 매체를 포함한다. 가령 시동 중에, 컴퓨터(241) 내 구성요소들 사이에서 정보를 전달하는 것을 돕는 기본 루틴들을 포함하는 기본 입출력 시스템(224)(BIOS)은 통상적으로 ROM(223)에 저장된다. RAM(260)은 통상적으로 즉시 액세스가능하고/하거나 프로세싱 유닛(259)에 의해 현재 운영되고 있는 데이터 및/또는 프로그램 모듈들을 포함한다. 한정하는 것이 아닌 예로서, 도 3b는 운영체제(225), 애플리케이션 프로그램(226), 기타 프로그램 모듈(227), 및 프로그램 데이터(228)를 예시한다.

[0046] 컴퓨터(241)는 또한 다른 착탈형/비착탈형, 휘발성/비휘발성 컴퓨터 저장 매체를 포함할 수 있다. 다만 예로서, 도 3b는 비착탈형, 비휘발성 마그네틱 매체로부터/로 읽고/쓰기를 하는 하드 디스크 드라이브(238), 착탈형 비휘발성 마그네틱 디스크(254)로부터/로 읽고/쓰기를 하는 마그네틱 디스크 드라이브(239), 및 CD ROM 또는 다른 광 매체 같은 착탈형, 비휘발성 광 디스크(253)로부터/로 읽기/쓰기를 하는 광 디스크 드라이브(240)를 예시한다. 전형적인 동작 환경에 사용될 수 있는 다른 착탈형/비착탈형, 휘발성/비휘발성 컴퓨터 저장 매체로는 마그네틱 테이프 카세트, 플래시 메모리 카드, DVD(digital versatile disks), 디지털 비디오 테이프, 고체 상태 RAM, 고체 상태 ROM 등이 포함되나 그러한 것에 국한되지 않는다. 하드 디스크 드라이브(238)는 통상적으로 인터페이스(234) 같은 비착탈형 메모리 인터페이스를 통해 시스템 버스(221)에 연결되고, 마그네틱 디스크 드라이브(239) 및 광 디스크 드라이브(240)는 통상적으로 인터페이스(235) 같은 착탈형 메모리 인터페이스를 통해 시스템 버스(221)에 연결된다.

[0047] 위에서 논의되고 도 3b에서 예시된 그러한 드라이브들 및 그 관련 컴퓨터 저장 매체는 컴퓨터(241)에 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 및 기타 데이터의 저장을 지원한다. 도 3b에서, 예를 들어 하드 디스크 드라이브(238)는 운영체제(258), 애플리케이션 프로그램(257), 기타 프로그램 모듈(256), 및 프로그램 데이터(255)를 저장하는 것으로 예시되어 있다. 이 구성요소들은 운영체제(225), 애플리케이션 프로그램(226), 기타 프로그램 모듈(227), 및 프로그램 데이터(228)와 같거나 다를 수 있다는 것을 알아야 한다. 운영체제(258), 애플리케이션 프로그램(257), 기타 프로그램 모듈(256), 및 프로그램 데이터(255)에는 여기서 그들이 다른 사본임을 나타내기 위해 다른 넘버들이 주어진다. 사용자는 키보드(251) 및 전형적으로는 마우스, 트랙볼 또는 터치 패드라 일컫는 포인팅 장치(252) 같은 입력 장치들을 통해 컴퓨터(241) 안에 명령과 정보를 입력할 수 있다. 다른 입력 장치들(미도시)에는 마이크로폰, 조이스틱, 게임 패드, 위성 접시, 스캐너 등이 포함될 수 있다. 이들 및 기타 입력 장치들은 흔히 시스템 버스에 연결된 사용자 입력 인터페이스(236)를 통해 프로세싱 유닛(259)에 연결되나, 병렬 포트나 게임 포트 또는 유니버설 시리얼 버스(USB) 같은 다른 인터페이스 및 버스 구조에 의해 연결될 수도 있다. 카메라(26, 28) 및 캡처 장치(20)가 콘솔(100)에 대한 추가 입력 장치들을 규정할 수 있다. 모니터(242) 또는 다른 유형의 디스플레이 장치 역시 비디오 인터페이스(232) 같은 인터페이스를 통해 시스템 버스(221)에 연결된다. 모니터 외에, 컴퓨터들은 출력 주변기기 인터페이스(233)를 통해 연결될 수 있는 스피커(244) 및 프린터(243) 같은 다른 주변기기 출력 장치들을 포함할 수도 있다.

[0048] 컴퓨터(241)는 원격 컴퓨터(246) 같은 한 개 이상의 원격 컴퓨터로의 로직 연결을 이용하여, 네트워킹 환경 안에서 동작할 수 있다. 원격 컴퓨터(246)는 퍼스널 컴퓨터, 서버, 라우터, 네트워크 PC, 피어 장치 또는 다른

전형적 네트워크 노드일 수 있으며, 도 3b에는 메모리 저장 장치(247) 만이 예시되었지만 통상적으로 컴퓨터(241)와 관련해 위에서 기술된 구성요소들 중 다수나 전부를 포함한다. 도 3b에 묘사된 로직 연결은 LAN(local area network)(245) 및 WAN(wide area network)(249)을 포함하지만, 다른 네트워크들 역시 포함할 수 있다. 그러한 네트워킹 환경은 사무소, 기업 전체의 컴퓨터 네트워크, 인트라넷 및 인터넷에서 일반적이다.

[0049] LAN 네트워킹 환경에서 사용될 때, 컴퓨터(241)는 네트워크 인터페이스나 어댑터(237)를 통해 LAN(245)에 연결된다. WAN 네트워킹 환경에서 사용될 때, 컴퓨터(241)는 통상적으로, 인터넷 같은 WAN(249)을 통한 통신을 설정하기 위한 모뎀(250) 또는 다른 수단을 포함한다. 내장형 또는 외장형일 수 있는 모뎀(250)이 사용자 입력 인터페이스(236)나 다른 알맞은 메커니즘을 통해 시스템 버스(221)에 연결될 수 있다. 네트워킹 환경에서, 컴퓨터(241)와 관련해 묘사된 프로그램 모듈들이나 그 일부는 원격 메모리 저장 장치에 저장될 수 있다. 한정하는 것이 아닌 예로서, 도 3b는 원격 애플리케이션 프로그램(248)을 메모리 장치(247) 상에 상주하는 것으로서 도시한다. 도시된 네트워크 접속은 본보기적인 것이며 컴퓨터들 사이에 통신 링크를 설정하는 다른 수단 역시 사용될 수 있다는 것을 알 수 있을 것이다.

[0050] 도 4a는 캡처 장치(20)로부터 생성될 수 있는 사용자의 전형적 골격 매핑을 묘사한다. 이 실시예에서, 다양한 관절 및 뼈대들이 식별된다: 각각의 손(302), 각각의 팔뚝(304), 각각의 팔꿈치(306), 각각의 상완(308), 각각의 어깨(310), 각각의 힙(312), 각각의 대퇴부(314), 각각의 무릎(316), 각각의 정강이(318), 각각의 발(320), 머리(322), 몸통(324), 척추의 최상부(326) 및 최하부(328), 및 손목(330). 더 많은 지점들이 추적되는 경우, 손가락이나 발가락의 뼈대와 관절 같은 추가적 특징들이나 코와 눈 같은 얼굴의 개별 특징들이 식별될 수 있다.

[0051] 자신의 몸을 움직임으로써 사용자가 제스처를 생성할 수 있다. 제스처는 이미지 데이터로서 캡처되어 의미 파악될 수 있는 사용자의 모션이나 포즈를 포함한다. 제스처는 공 던지는 흉내 같은 모션을 포함하는 동적인 것일 수 있다. 제스처는 자신의 몸통(324) 앞에서 팔뚝들(304)로 팔짱을 끼는 것 같은 정적인 포즈일 수 있다. 제스처는 또, 가령 가짜 칼을 휘두르는 것 같이, 소품을 포함할 수도 있다. 제스처는 손들(302)을 서로 마주치는 것 같이 한 신체 부분을 넘는 부분을 포함하거나, 입술을 오무리는 것 같은 좀 더 미묘한 제스처를 포함할 수 있다.

[0052] 제스처들은 일반적인 컴퓨팅 맥락에서 입력으로 사용될 수 있다. 예를 들어 양 손(302)이나 다른 신체 부분들의 다양한 모션들은 계층적 리스트의 상하를 탐색하거나, 파일을 열거나, 파일을 닫거나, 파일을 저장하는 것 같은 전형적인 모든 시스템 작업들에 해당할 수 있다. 제스처들은 또한 비디오 게임 고유의 맥락에서 그 게임에 좌우되어 사용될 수도 있다. 예를 들어, 운전 게임을 할 때 손(302)과 발(320)의 다양한 모션은 어떤 방향으로 차를 조종하고, 기어를 변속하고, 가속시키고, 브레이크를 밟는 것에 해당할 수 있다.

[0053] 사용자는 제자리에서 스스로 걷거나 달림으로써 걷거나 달리는 것에 해당하는 제스처를 생성할 수 있다. 사용자가 이동하지 않고 각각의 다리(312-320)를 교대로 들거나 내려 놓으면서 걷기를 흉내낼 수도 있다. 시스템은 각각의 힙(312) 및 각각의 대퇴부(314)를 분석하여 이러한 제스처를 파악할 수 있다. 한 힙-대퇴부 각도(수직선을 기준으로 측정됨, 이때 서 있는 다리는 0°의 힙-대퇴부 각도를 가지며, 앞으로 수평하게 뻗은 다리는 90°의 힙-대퇴부 각도를 가짐)가 다른 대퇴부 기준으로 소정 문턱치를 초과할 때 한 걸음이 인식될 수 있다. 두 다리를 교대함에 따른 일정 수의 연속적인 걸음 뒤에 걷거나 달리기가 인식될 수 있다. 가장 최근의 두 걸음 사이의 시간을 주기로 고려할 수 있다. 해당 문턱치 각도가 만족되지 않는 일정 수의 주기들 뒤에, 시스템은 그 걷거나 달리기 제스처가 중단되었다고 판단할 수 있다.

[0054] "걷거나 달리기" 제스처가 주어질 때, 애플리케이션은 이 제스처와 관련된 파라미터들의 값들을 설정할 수 있다. 이 파라미터들에는 상기 문턱치 각도, 걷거나 달리기 제스처를 일으키는데 필요한 걸음의 수, 제스처를 마치기 위해 아무 걸음 걷지도 발생하지 않은 경우의 주기들의 개수, 및 제스처가 걷기인지 달리기인지 여부를 판단하는 문턱 주기가 포함될 수 있다. 사용자가 자신의 다리를 빠르게 이동할 것이기 때문에 빠른 주기는 달리기에 해당할 것이며, 느린 주기는 걷기에 해당할 것이다.

[0055] 제스처는 애플리케이션이 자체 파라미터들을 가지고 무효화시킬 수 있는 디폴트 파라미터들의 집합과 우선 관련될 수 있다. 이러한 상황에서, 애플리케이션은 파라미터들을 제공하도록 강제되지 않으며, 대신 애플리케이션이 정의한 파라미터들이 부재할 때 제스처가 인식될 수 있게 하는 디폴트 파라미터들의 집합을 사용할 수 있다.

[0056] 제스처와 관련될 수 있는 다양한 출력이 존재한다. 제스처가 일어나고 있는지 여부에 관한 기준선인 "예 또는

아니오"가 있을 수 있다. 추적된 사용자 움직임이 제스처에 상응할 가능성에 해당하는 신뢰 수준 또한 있을 수 있다. 이것은 0과 1을 포함하는 그 사이의 부동소수점 숫자들로 된 범위에 있는 선행 스케일일 수 있다. 이 제스처 정보를 수신하는 애플리케이션이 긍정 오류(false-positive)를 입력으로서 수용할 수 없을 때, 애플리케이션은 적어도 .95 같은 높은 신뢰 수준을 가지는 인지된 제스처들만을 사용할 수 있다. 애플리케이션이 부정 오류를 맺가로 해서라도 모든 제스처 인스턴스를 인식해야 하는 경우, 애플리케이션은 단지 .2보다 큰 것들 같이 적어도 훨씬 더 낮은 신뢰 수준을 가지는 제스처들을 사용할 것이다. 제스처는 가장 최근의 두 걸음 사이의 시간에 대한 출력을 가질 수 있고, 첫 번째 걸음만이 등록되었을 경우 그것은 -1 같은 예비 값으로 세팅될 수 있다(이는 어떤 두 걸음 사이의 시간이 양이어야 하기 때문이다). 제스처는 또한 가장 최근의 걸음 중에 도달했던 가장 높은 대퇴부 각도에 대한 출력을 가질 수도 있다.

[0057] 또 다른 제스처의 예가 "뒤꿈치 들고 점프하기"이다. 여기서 사용자는 뒤꿈치를 땅으로부터 들면서 발가락은 땅에 닿게 함으로써 이 제스처를 만들 수 있다. 다른 대안으로서, 사용자는 대기중으로 점프할 수 있고 이때 자신의 두 발(320)은 지면을 완전히 떠나 있다. 시스템은 어깨(310), 힙(312) 및 무릎(316)의 각도 관계를 분석하여 이들이 똑바로 서 있는 것에 상응하는 배열의 위치에 있는지를 파악하도록 함으로써 이 제스처에 대한 골격을 파악할 수 있다. 그런 다음 어떤 위를 향한 가속에 대해 이 지점들과 상위(326) 및 하위(328) 척추 지점들이 모니터링된다. 충분한 가속의 조합이 점프 제스처를 일으킬 수 있다.

[0058] "뛰뛰뛰 들고 점프" 제스처가 주어질 때, 애플리케이션은 이 제스처와 관련된 파라미터들의 값들을 설정할 수 있다. 파라미터들은 사용자의 어깨(310), 힙(312) 및 무릎(316)의 어떤 조합이 그 제스처를 일으키기 위해 얼마나 빠르게 위로 이동해야 하는지를 결정하는 상기 가속 문턱치뿐만 아니라, 점프가 여전히 유발될 수 있는 어깨(310), 힙(312) 및 무릎(316) 사이의 최대 정렬 각도를 포함할 수 있다.

[0059] 출력은 신뢰 수준뿐만 아니라 점프 시점의 사용자 신체의 각도를 포함할 수 있다.

[0060] 제스처들을 정밀하게 식별하는데 있어, 제스처를 수신할 애플리케이션의 특정사항들에 기초하여 제스처의 파라미터들을 설정하는 것이 중요하다. 적절하게 제스처들 및 사용자 의도를 식별하는 것이 긍정적인 사용자 경험을 만드는 데 있어 큰 도움이 된다. 제스처 인식기 시스템이 너무 민감하고 손(302)의 약간의 포워드 모션조차 던지기로 해석되는 경우, 사용자가 제스처를 만들 의도가 없을 때 제스처들이 인식되고 그에 따라 그가 시스템에 대한 통제권을 충분히 가지지 못하기 때문에 사용자가 낙담하게 될 수 있다. 제스처 인식기 시스템이 충분한 감도를 갖지 못하는 경우, 시스템은 던지기 제스처를 만들고자 하는 사용자의 의식적 시도를 인식하지 못할 수 있고 그에 따라 같은 식으로 사용자를 낙담시킬 수 있다. 감도 스펙트럼의 어느 한 쪽 끝에 있을 때 시스템으로 입력을 알맞게 제공할 수 없기 때문에 사용자는 낙담하게 된다.

[0061] 제스처에 대한 다른 파라미터는 이동한 거리일 수 있다. 사용자의 제스처들이 가상 환경 내 아바타의 액션을 제어하는 경우, 그 아바타는 공으로부터 팔 길이만큼 떨어질 수 있다. 사용자가 공과 상호작용하면서 그것을 잡고자 하는 경우는 사용자가 잡기 제스처를 취하면서 자신의 팔(302-310)을 완전히 뻗는 것을 요할 것이다. 이 상황에서 사용자가 부분적으로만 자신의 팔(302-310)을 뻗는 비슷한 잡기 제스처는 공과 상호작용하는 결과를 달성할 수 없을 것이다.

[0062] 제스처 또는 그 일부는 그것이 발생할 수 있는 공간의 체적을 파라미터로서 가질 수 있다. 이러한 공간의 체적은 통상적으로 제스처가 신체의 움직임을 포함하는 경우 신체와 관련하여 표현될 수 있다. 예를 들어, 오른손잡이 사용자의 미식축구 던지기 제스처는 던지기 하는 팔(302a-310a)과 같은 머리(322)의 방향에서 오른쪽 어깨(310a)보다 낮지 않은 공간의 체적 안에서만 인식될 수 있다. 신체로부터 먼 바깥쪽 바운드는 규정되지 않은 채로 있는 이러한 던지기 제스처와 같은 것에서 체적의 모든 바운드들을 반드시 규정할 필요는 없을 것이며, 체적은 무한정으로 확장되거나 모니터링되는 장면의 가장자리까지 확장된다.

[0063] 도 4b는 도 2의 제스처 인식기 엔진(190)의 전형적인 일 실시예의 보다 상세사항을 제공한다. 도시된 바와 같이, 제스처 인식기 엔진(190)은 제스처가나 제스처들을 판단하기 위해 적어도 한 개의 필터(418)를 포함할 수 있다. 필터(418)는 제스처(426)(이제부터 "제스처(gesture)"이라 칭함)를 규정하는 정보와 함께 그 제스처에 대한 파라미터(428)나 메타데이터를 포함한다. 예를 들어, 신체의 뒤쪽 너머로부터 신체의 앞쪽을 지날 때까지 양손 중 하나의 모션을 포함하는 던지기는 깊이 인식 카메라에 의해 움직임이 캡처될 때, 신체의 뒤쪽 너머로부터 신체의 앞쪽을 지날 때까지의 사용자의 양손 중 하나의 움직임을 표현하는 정보를 포함하는 제스처(426)로서 구현될 수 있다. 그런 다음 그 제스처(426)에 대한 파라미터들(428)이 설정될 수 있다. 제스처(426)이 던지기인 경우, 파라미터(428)는 손이 도달해야 하는 문턱 속도, 손이 이동해야 하는 거리(절대적이거나 대체적으로 사용자의 사이즈에 대해 상대적임) 및 제스처가 일어났다는 인식기 엔진에 의한 신뢰 평가가 될 수 있다.



제스처(426)에 대한 이 파라미터들(428)은 애플리케이션들 사이에서, 혹은 단일 애플리케이션의 상황들 사이에서, 혹은 시간에 따른 한 애플리케이션의 한 상황 안에서 가변될 수 있다.

- [0064] 필터들은 모듈방식이거나 교환가능한 방식일 수 있다. 일 실시예에서 필터는 각각이 하나의 유형을 가지는 여러 입력들과 각각이 하나의 유형을 가지는 여러 출력들을 가진다. 이 상황에서 제1필터는 같은 개수 및 유형의 입력들을 가지고 인식기 엔진 구조의 어떤 다른 양태를 변경하지 않고 제1필터로서 출력하는 제2필터로 대체될 수 있다. 예를 들어, 운전에 대한 제1필터가 있을 수 있고, 제1필터는 입력으로서 골격 데이터를 취하고 그 필터와 관련된 제스처가 일어난다는 신뢰도 및 조종 각도를 출력한다. 이러한 제1필터를 제2운전 필터로 치환하고자 하는 경우(어쩌면 제2운전 필터가 보다 효율적이고 보다 적은 프로세싱 자원을 요한다는 이유로), 제2필터가 같은 입력과 출력들(골격 데이터 유형이라는 한 개의 입력과 신뢰도 유형 및 각도 유형이라는 두 개의 출력)을 가지는 한 간단히 제1필터를 제2필터로 교체함으로써 그렇게 할 수 있다.
- [0065] 필터가 파라미터를 가질 필요는 없다. 예를 들어 사용자의 키를 리턴하는 "사용자 키" 필터는 튜닝될 수 있는 어떤 파라미터들도 고려하지 않을 수 있다. 다른 "사용자 키" 필터는 사용자의 신발류, 헤어 스타일, 머리에 착용하는 것 및 사용자 키를 판단할 때의 자세를 고려할지 여부에 대한 것 같은 튜닝가능한 파라미터들을 가질 수 있다.
- [0066] 필터로의 입력들로는 관절에서 만나는 뼈들에 의해 만들어지는 각도 같이 사용자의 관절 위치에 대한 관절 데이터, 장면으로부터의 RGB 컬러 데이터, 및 사용자의 관점 변화율 같은 것들이 포함될 수 있다. 필터로부터의 출력들은 소정 제스처가 취해지고 있다는 신뢰도, 제스처 모션이 취해지는 속도, 및 제스처 모션이 취해지는 시점 같은 것들을 포함할 수 있다.
- [0067] 맥락은 문화적 맥락일 수 있고, 환경적 맥락일 수 있다. 문화적 맥락은 시스템을 사용하는 사용자의 문화를 의미한다. 상이한 문화들은 현저히 다른 의미를 전달하는 비슷한 제스처들을 사용할 수 있다. 예를 들어 다른 사용자에게 "보아라" 또는 "눈을 사용하라"는 말을 하고자 하는 미국인 사용자는 자신의 검지를 자신의 눈의 말단에 가까운 머리에 댈 수 있다. 그러나, 이탈리아인 사용자에게는 이 제스처가 마피아에 대한 언급으로서 해석될 수 있다.
- [0068] 마찬가지로 하나의 애플리케이션에 대해 서로 다른 환경들 간에 상이한 맥락들이 있을 수 있다. 자동차 작동을 수반하는 FPS(first-person shooter) 게임을 인용할 수 있다. 사용자가 서 있을 때, 땅을 향해 손가락들로 주먹을 만들고 그 주먹을 몸에서 멀리 앞으로 뻗는 것은 펀치 제스처를 나타낼 수 있다. 사용자가 운전하고 있는 상황에 있을 때, 그와 동일한 모션은 "기어 변속" 제스처를 나타낼 수 있다. 사용자가 자신의 게임을 저장하고 자신의 캐릭터들 사이에서 장비를 선택하거나 직접 게임 플레이를 포함하지 않는 유사 액션을 수행할 수 있는 한 개 이상의 메뉴 환경들 또한 있을 수 있다. 그러한 환경에서, 이런 동일한 제스처는 어떤 것을 선택하거나 다른 스크린으로 이동하는 것 같은 제3의 의미를 가질 수 있다.
- [0069] 제스처 인식기 엔진(190)은 제스처 필터(418)에 기능성을 부여하는 기본 인식기 엔진(416)을 가질 수 있다. 일 실시예에서, 인식기 엔진(416)이 구현하는 기능은 인식된 제스처들 및 기타 입력을 추적하는 시간에 따른 입력(input-over-time) 아카이브, 은닉 마코프 모델(Hidden Markov Model) 구현예(여기서 모델링되는 시스템은 미지의 파라미터를 가진 마코프 프로세스-현재의 상태가 미래의 상태를 결정하는데 필요한 어떤 과거 상태 정보를 캡슐화하고, 그래서 어떤 다른 과거 상태 정보도 이러한 목적으로 관리되지 않음- 뿐 아니라 제스처 인식의 특정 인스턴스들을 해결하는데 필요로 되는 다른 기능 역시 포함한다.
- [0070] 필터들(418)은 기본 인식기 엔진(416)의 맨 위에 장착되어 구현되며, 엔진(416)에 의해 필터들(418)로 제공되는 서비스들을 활용할 수 있다. 일 실시예에서 기본 인식기 엔진(416)은 수신된 데이터를 처리하여 그것이 어떤 필터(418)의 요건을 만족하는지 여부를 판단하도록 한다. 입력을 파싱하는 것 같이 그렇게 제공된 서비스들은 각각의 필터(418)보다는 기본 인식기 엔진(416)에 의해 한 번 제공되므로, 그러한 서비스는 한 주기 동안 필터(418) 마다 한 번씩이 아닌 그 주기 동안 단 한번 처리되면 되므로, 제스처를 결정하는데 필요한 처리가 줄게 된다.
- [0071] 애플리케이션은 인식기 엔진(190)에 의해 제공된 필터들(418)을 사용하거나, 기본 인식기 엔진(416)에 플러그인 하는 자체 필터(418)를 제공할 수 있다. 일 실시예에서 모든 필터들(418)은 그러한 플러그인 특성을 수행하기 위한 공통 인터페이스를 가진다. 또한 모든 필터들(418)이 파라미터들(428)을 활용할 수 있으므로, 아래에 기술되는 것과 같은 하나의 제스처 툴이 전체 필터 시스템(418)을 디버깅 및 튜닝하는데 사용될 수 있다.
- [0072] 이러한 파라미터들(428)은 제스처 툴(420)에 의해 애플리케이션이나 애플리케이션의 맥락에 대해 튜닝될 수 있



다. 일 실시예에서, 제스처 툴(420)은 복수의 슬라이더(422)를 포함하며, 각각의 슬라이더(422)는 파라미터(428) 뿐 아니라 신체(424)의 도해적 표현에 해당한다. 대응하는 슬라이더(422)를 통해 파라미터(428)가 조정되면서, 신체(425)는 그 파라미터들(428)을 가지고 제스처로서 인식될 수 있는 액션들 및 그렇게 식별된 파라미터들(428)을 가지고 제스처로서 인식되지 못할 액션들 모두를 시연한다. 제스처 파라미터들(428)의 이러한 심상(visualization)은 제스처를 디버깅하고 또한 정밀 튜닝하기 위한 효과적인 수단을 제공한다.

[0073] 도 5는 스택킹된 제스처들이나 필터들(418)로부터 생성된 보다 복잡한 제스처들이나 필터들(418)을 묘사한다. 제스처들은 서로의 위에서 스택킹될 수 있다. 즉, 한 개를 넘는 제스처가 사용자에게 의해 한 시점에서 표현될 수 있다. 예를 들어, 던지기 제스처가 만들어질 때 던지기가 아닌 어떤 입력도 불허하기 보다는, 사용자가 그 제스처의 성분들을 제외하고는 움직이지 않고 가만히 있을 것(가령, 오직 한 팔이 개입되는 던지기 제스처를 만드는 동안 가만히 서있기)을 요구한다. 제스처가 스택킹될 경우 사용자는 점프 제스처와 던지기 제스처를 동시에 만들 수 있고, 이 제스처들 모두가 제스처 엔진에 의해 인식될 것이다.

[0074] 도 5a는 스택킹 패러다임에 따른 간단한 제스처 필터(418)를 묘사한다. IFilter 필터(502)는 모든 제스처 필터에서 사용될 수 있는 기본 필터(418)이다. IFilter(502)는 사용자 위치 데이터(504)를 취하고 제스처가 발생했다는 신뢰 수준(506)을 출력한다. 그것은 또한 그 위치 데이터(504)를, 그 위치 데이터를 입력으로 취하여 사용자가 조종하는 각도(가령, 사용자의 현재 방향에서 오른쪽으로 40도)를 출력하는 Steering Wheel 필터(508)로 공급한다.

[0075] 도 5b는 도 5a의 제스처 필터 위에 필터들(418)을 스택킹하는 보다 복잡한 제스처를 묘사한다. IFilter(502) 및 Steering Wheel(508) 외에, IFilter(502)로부터 위치 데이터(504)를 수신하고 사용자가 제스처(514)를 통해 만든 진전의 정도를 출력하는 ITracking 필터(512)가 있다. ITracking(512) 역시 위치 데이터(504)를, 비상 브레이크를 사용하는 것 같이 차량을 조종할 때 취할 수 있는 다른 제스처들에 관한 필터들(418)인 GreaseLightning(516) 및 EBrake(518)로 제공한다.

[0076] 도 6은 미식축구 비디오 게임에서 "페어 캐치(fair catch)"를 신호하기 위해 사용자가 만들 수 있는 전형적 제스처를 묘사한다. 이 도면들은 시점들마다 사용자를 묘사한 것으로, 도 6(a)는 최초의 시점이고, 도 6(e)는 마지막 시점이다. 이 도면들 각각은 깊이 인식 카메라(402)에 의해 캡처된 것 같은 스냅샷이나 이미지 데이터의 프레임에 해당할 수 있으며, 반드시 이미지 데이터의 연속 프레임들에 해당하는 것은 아닌데, 이는 깊이 카메라(402)가 사용자가 거리를 이동하는 것보다 더 빠르게 프레임들을 캡처할 수 있기 때문이다. 예를 들어, 그 제스처는 3초의 시간 동안 일어날 수 있으며, 깊이 인식 카메라가 초당 40 개의 프레임들로 데이터를 캡처할 경우 그 카메라는 사용자(602)가 이 페어 캐치 제스처를 만들었을 동안 이미지 데이터의 60 개 프레임을 캡처했을 것이다.

[0077] 도 6(a)에서 사용자(602)는 자신의 양 팔(604)을 양 옆에 늘어뜨린 채 시작한다. 그런 다음 그는 도 6(b)에 묘사된 것과 같이 팔들을 자신의 양 어깨 위로 들어올리고, 그 다음에 더 올려 도 6(c)에 묘사된 것과 같이 거의 자신의 머리 높이까지 들어올린다. 거기서부터 사용자는 도 6(d)에 묘사된 바와 같이 양 팔(604)을 어깨 높이까지 내리고, 그런 다음 다시 도 6(e)에 묘사된 것과 같이 거의 자기 머리 높이까지 양 팔을 들어 올린다. 시스템이 제스처가 취소되거나 다른 제스처가 만들어지고 있다는 것을 신호할 수 있는 아무 간섭 자세도 없이 사용자(602)에 의한 그 자세들을 캡처할 경우, 시스템은 페어 캐치 제스처 필터가 사용자(602)가 페어 캐치 제스처를 만들었다는 높은 신뢰 수준을 출력하게 할 수 있다.

[0078] 도 7은 이미지 데이터의 각각의 프레임이 사용자의 골격 맵을 생성하기 위해 파싱되었을 때 도 5의 "페어 캐치" 제스처를 예시한다. 사용자의 깊이 이미지로부터 골격 맵을 만들었던 시스템은 이제 그 사용자의 신체가 시간에 따라 어떻게 움직이는지를 판단하고, 그로부터 제스처를 파싱할 수 있다.

[0079] 도 7(a)에서, 사용자의 양 어깨(310)는 자신의 팔꿈치들(306) 위에 있고, 팔꿈치들은 자신의 양손(302) 위에 있다. 양 어깨(310), 양 팔꿈치(306) 및 양 손(302)은 도 7(b)에서 이제 같은 높이 상에 있다. 이제 시스템은 도 7(c)에서 양 손(302)이 양 팔꿈치 위에 있고, 양 팔꿈치는 양 어깨(310) 위에 있는 것을 묘사한다. 도 7(d)에서, 사용자는 양 어깨(310), 양 팔꿈치(306) 및 양 손(302)이 같은 높이에 있는 도 7(b)의 자세로 돌아갔다. 도 7(e)에 보인 제스처의 마지막 자세에서, 사용자는 양 손(302)이 양 팔꿈치 위에 있고, 양 팔꿈치는 양 어깨(310) 위에 있는 도 7(c)의 자세로 돌아간다.

[0080] 캡처 장치(20)가 일련의 정지 이미지들을 캡처함으로써 어느 한 이미지 안에 사용자가 정지된 상태로 나타나지만, 사용자는 (위에 논의된 바와 같이 정적인 제스처와 다른) 그 제스처를 수행하는 동안에 움직이고 있다. 시

시스템은 각각의 정지 이미지 내 그러한 일련의 포즈들을 가져올 수 있고, 그로부터 사용자가 만들고 있는 움직이는 제스처의 신뢰 수준을 판단할 수 있다.

[0081] 제스처를 수행할 때, 사용자가 자신의 오른쪽 어깨(310a), 오른쪽 팔꿈치(306a) 및 오른쪽 손(302a)에 의해 형성되는 각도, 예컨대, 140° 및 145° 사이의 각도를 생성할 수 있는 가능성은 낮다. 그래서, 페어 캐치 제스처(426)에 대한 필터(418)를 이용하는 애플리케이션은 애플리케이션의 세부사항을 가장 잘 지원하도록 관련 파라미터들(428)을 튜닝할 수 있다. 예를 들어 도 7(c) 및 7(e)의 자세들은 팔꿈치(306) 위치와 관계없이 사용자가 자신의 양 손(302)을 자신의 양 어깨(310) 위에 놓을 때마다 인식될 수 있다. 보다 엄격한 파라미터 집합은 양 손(302)이 머리(310) 위에 있을 것과 양 팔꿈치(306)가 양 어깨(310) 위에 있으면서 머리(310)와 양 손(302) 사이에 있을 것을 요할 수 있다. 또한, 페어 캐치 제스처(426)를 위한 파라미터들(428)은 사용자가 1.5초 같이 특정 시간 안에 도 7(a)의 자세부터 도 7(e)까지의 자세까지 취할 것을 요할 수 있고, 만약 사용자가 그러한 자세들을 취하는데 1.5초가 넘는 시간이 걸리면 그것을 페어 캐치(418)로 인식하지 않을 것이며, 매우 낮은 신뢰 수준이 출력될 것이다.

[0082] 도 8(a)-8(c)는 여러 캡처되는 움직임들 및 포즈들을 통해 동일 시스템으로 인식하는 달리기 제스처를 만들고 있는 사용자를 예시한다.

[0083] 도 8(a)는 풀 달리기 제스처를 만들고 있는 사용자를 예시한다. 사용자(18)는 캡처 장치(20)에 의해 캡처된다. 사용자(18)는 제자리에서 달림으로써(자신의 양 무릎 각각을 거의 허리 높이까지 교대로 들어올린 다음 다리를 땅으로 떨어뜨림으로써) 풀 달리기 제스처를 생성한다. 이 버전의 풀 달리기 제스처는 사용자(18)가 이 제스처를 지속시키고 싶어하는 지속기간 동안 그 제스처를 포함하는 모션들을 반복한다는 점에서 주기적인 제스처가다.

[0084] 도 8(b)는 도 8(a)의 풀 달리기 제스처 움직임의 부분집합을 포함하는 슛트 달리기 제스처를 만드는 사용자를 예시한다. 이 버전의 슛트 제스처를 만들기 위해 사용자(18)는 자신의 무릎이 거의 힙 수준에 있도록 양다리 중 하나를 들어올리고 그 포즈를 유지한다. 이 제스처 슛트는 도 8(a)의 풀 제스처 움직임의 부분집합을 포함한다. 도 8(a)의 사용자는 자신의 양 무릎을 반복적으로 들었다 내리며, 이때 사용자는 무릎을 올리고 그 포즈를 유지한다. 도 8(a)의 풀 제스처가 이 실시예에서는 주기적 움직임을 수반하지만, 슛트 제스처는 일련의 반복되지 않은 움직임들이나 전체적으로 그 일련의 것들이 반복되지 않는 일련의 움직임들을 수반한다. 일 실시예에서 사용자(18)는 그 제스처를 마치고 싶을 때 서있는 포즈가 되도록 무릎을 아래로 내린다. 일 실시예에서 이렇게 무릎을 내리는 행위 역시 풀 제스처의 부분집합을 포함할 수 있다. 일 실시예에서 컴퓨팅 환경(12)은 사용자(18)가 특정 시간을 넘는 동안 자신의 무릎을 거의 힙 수준에 유지시킬 때 이 제스처가 풀 제스처를 생성하기 보다는 제스처 슛트를 끝내기 위한 것이라고 판단한다.

[0085] 도 8(c)는 도 8(a)의 풀 달리기 제스처와 별개의 두 번째 유형의 슛트 달리기 제스처를 만드는 사용자를 예시한다. 여기서 사용자(18)는 한걸음 앞으로 내딛고, 한 발이 다른 발보다 앞에 있고 양 발은 바닥에 있는 그 포즈를 달리기 제스처를 생성하고자 하는 지속기간 동안 유지한다. 이 자세는 도 8a의 풀 달리기 제스처에서는 보이지 않는다. 사용자(18)는 다시 서 있는 포즈로 걸음을 옮김으로써 제스처를 마칠 수 있다. 이 제스처는 도 8(b)의 제스처와 유사한데, 그것은 양자가 그 제스처를 시작하고 그 제스처를 유지하기 위해 포즈를 유지하는 움직임 및 그 제스처를 끝내기 위한 움직임을 수반한다는 점에서 그러하다.

[0086] 도 9는 제스처 슛트를 위한 전형적 동작 절차를 예시한다. 위에서 논의된 바와 같이, 컴퓨팅 장치로의 하나의 제스처 입력은 사용자에 의해 수행되는 복수의 양태들의 결과로서 컴퓨팅 장치에 의해 인식될 수 있다. 일 실시예에서, 제스처가 사용자에 의해 수행될 수 있는 이러한 복수의 양태들에는 제스처의 풀 버전과 제스처의 슛트 버전이 포함된다.

[0087] 제스처 슛트는 다양한 적용 맥락에서 사용될 수 있다. 예를 들어, 달리기 제스처 슛트는 트랙과 필드 게임 같이 달리기를 수반하는 애플리케이션들에서 사용될 수 있다. 텍스트 입력 슛트는 한 애플리케이션의 텍스트 입력 맥락에서 사용될 수 있다. 예를 들어 사용자는 텍스트를 입력하기 위해 수화를 사용할 수 있다. 풀 버전의 워드 제스처는 H-E-A-R-T 같이 워드의 각 문자를 수화로 행하는 것을 포함할 수 있다. "heart"라는 단어의 슛트는 손을 하트 표시로 만드는 사용자 같이, 하트에 대한 하나의 제스처를 포함할 수도 있다. 그러한 수화는 미국 수화(ASL)를 포함할 수 있다.

[0088] 제스처 슛트는 제스처의 풀 버전에 해당하는 것과는 다른 신체 부분들을 개입시킬 수 있다. 예를 들어, 사용자가 양다리를 사용하기에는 모자라고, 풀 버전의 달리기 제스처는 제자리에서의 달리기를 수반하는 경우, 그 제

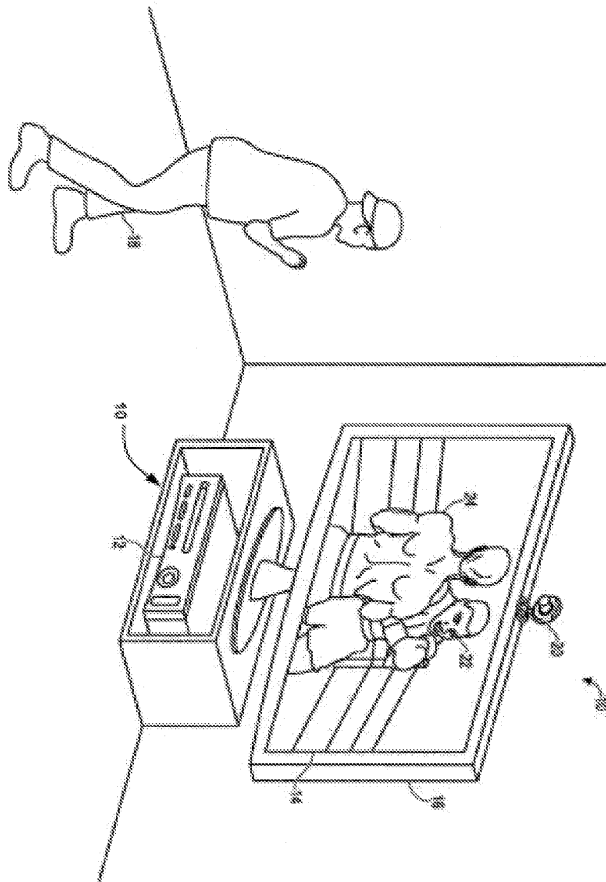
스처의 슷컷은 사용자의 손을 가지고 달리기 모션을 흉내 내는 것을 포함할 수 있다.

- [0089] 선택가능한 제스처(902)는 제스처의 슷컷을 규정하는 것에 해당하는 사용자 입력을 수신하는 것을 묘사한다. 예를 들어, 컴퓨팅 장치에 의해 재촉되거나 컴퓨팅 장치로 그렇게 하고자 하는 자신의 요망을 지시하는 일을 통해, 사용자는 캡처 장치에 의해 캡처되어 제스처를 수행하기 위한 양태로서 저장되는 모션이나 포즈를 만들 수 있다.
- [0090] 일 실시예에서, 사용자가 자신의 움직임이나 포즈를 통해 제스처 슷컷을 규정했을 경우, 사용자는 이제 시스템 상에서 제스처 슷컷을 세밀화할 수 있다. 예를 들어 필터들 및 대응 파라미터들을 사용해 제스처들이 인식되는 경우, 사용자는 위에 논의된 것과 같은 방식으로 자신의 제스처 슷컷의 파라미터들을 튜닝할 수 있다.
- [0091] 일 실시예에서, 제스처의 슷컷은 이차 제스처의 풀 버전에 해당한다. 제스처 슷컷은 복수의 풀 제스처들에 대응할 수 있고, 사용자가 어떤 슷컷을 규정하는 경우 그는 그 슷컷이 복수의 풀 제스처들에 대응하도록 지시할 수 있다. 예를 들어, 텍스트 파일을 인쇄하는 상황에서, 사용자는 묘사될 용지 방향을 선택하는 풀 제스처, 네 개의 인쇄 사본을 선택하는 풀 제스처, 및 인쇄할 특정 프린터를 선택하는 풀 제스처에 해당하는 하나의 제스처 슷컷을 규정할 수 있다.
- [0092] 제스처 필터들과 파라미터들을 통해 제스처들이 인식되는 일 실시예에서, 제스처의 슷컷 및 제스처의 풀 버전은 같은 제스처 필터를 사용할 수 있지만 한 개 이상의 파라미터들에 대해 다른 값을 사용할 수 있다. 예를 들어 풀 버전의 "공 던지기" 제스처는 사용자가 자신의 손을 몸통 뒤로부터 몸통 앞에 거의 팔 길이까지 이동할 것을 요할 수 있다. 슷컷은 손이 그렇게 멀리 뒤로나 앞으로 뻗을 필요가 없도록 손이 이동해야 하는 필요 거리를 줄일 수 있다. 이것은 "최소 손 거리"에 대한 것 같은 파라미터 값이나 값들을 변경하기 위해 실시될 수 있다.
- [0093] 제스처(904)는 캡처 장치에 의해 캡처되는 데이터를 수신하는 것을 묘사한 것으로, 그 데이터는 사용자가 수행한 제스처에 해당한다. 캡처 장치는 가령 사용자가 위치하는 거리에서 공간의 바닥부터 천정까지 그리고 모든 벽면까지 사용자의 모든 것을 포함하는 장면을 캡처할 수 있다. 캡처 장치는 또한 사용자가 책상에 앉아 있을 때 배 위쪽부터 보이는 사용자처럼 사용자의 일부분만을 포함하는 장면도 캡처할 수 있다. 캡처 장치는 또한 사용자가 자신의 손에 들고 있는 소품 카메라 같이 사용자에게 의해 통제되는 오브젝트를 캡처할 수도 있다.
- [0094] 선택가능한 제스처(906)는 애플리케이션으로부터 제스처의 슷컷을 이용해 데이터를 처리하도록 결정하는 것을 묘사한다. 애플리케이션은 입력으로서 사용되는 슷컷을 어떤 방식에 따라 제한할 수 있다. 예를 들어, 트랙 및 필드 게임에서, 달리기는 프로세스에 필수적인 것으로 간주될 수 있고, 애플리케이션은 사용자가 달리고 싶어할 때 달리기 제스처에 대한 슷컷을 불허하거나 불가능하게 하면서 사용자에게 풀 달리기 제스처를 만들도록 요구할 수 있다. 반대로, FPS 게임에서 달리기는 게임 이용에 보조적인 것으로 간주될 수 있으므로, 달리기 제스처에 대한 슷컷의 사용은 허락될 수 있다. 이러한 FPS 게임에서는 총을 쏘는 역학이 프로세스에 필수적인 것으로 간주될 수 있고, 애플리케이션은 겨누거나 쏘는 제스처에 대한 슷컷을 불허하거나 불가능하게 할 수 있다.
- [0095] 일 실시예에서 사용자는 앞서 논의된 것과 같이 사용자가 동시에 여러 제스처들을 수행하는 것과 같은 방식으로 제스처들의 슷컷들과 제스처들의 풀 버전들 모두를 수행할 수 있다. FPS 예를 사용할 때 사용자는 달리기 제스처의 슷컷과 겨누는 제스처의 풀 버전을 동시에 만들 수 있다.
- [0096] 일 실시예에서, 제스처의 슷컷을 이용해 데이터를 처리하고자 하는 이러한 결정은 사용자로부터 비롯된다. 예를 들어 어떤 슷컷을 처리할 것인가 하는 것은 사용자가 선택한 애플리케이션의 난이도에 상응할 수 있다. 사용자가 가장 낮은 난이도를 선택한 경우, 모든 슷컷들이 처리될 수 있다. 사용자가 난이도를 높이면, 아무 슷컷도 처리되지 않는 최고 난이도가 될 때까지 처리하도록 허락되는 슷컷들의 수가 줄어들 수 있다.
- [0097] 이러한 결정은 사용자 능력에 맞춰 시간에 따라 바뀔 수 있다. 예를 들어 허락된 슷컷들의 디폴트 설정은 어떤 애플리케이션 세션의 시작시에 구현될 수 있으며, 그 세션 도중에 사용자가 제스처들을 잘 수행하는 자신의 능력을 보이거나 그러한 능력이 모자란다는 것을 보일 때, 허락되는 슷컷들이 늘거나 줄 수 있다. 또한, 사용자가 세션 진행 중에 지치거나 역량이 높아질 때, 자신의 현재의 능력 상태에 상응하도록 허락되는 슷컷들이 늘거나 줄 수 있다.
- [0098] 제스처(908)는 사용자가 어느 제스처의 풀 버전에 해당하는 제스처의 슷컷을 수행했는지 여부에 해당하는 출력을 결정하도록 데이터를 처리하는 것을 보인다. 일 실시예에서 이 출력은 그 제스처가 발생했다는 신뢰 수준을 포함할 수 있다. 일 실시예에서 이것은 제스처의 풀 버전이 관찰되었는지 제스처 슷컷이 관찰되었는지 여부에 대한 표시를 포함할 수 있다.

- [0099] 제스처(910)는 제스처의 슛컷에 해당하는 출력을 애플리케이션으로 보내는 것을 묘사한다. 현재의 제스처들이 애플리케이션에 의해 수행되는 경우, 처리된 사용자 입력을 취하고 그것을 애플리케이션 내(in-application) 액션에 매핑하는 애플리케이션의 어느 구성요소로 출력이 보내질 수 있다.
- [0100] 선택 가능한 제스처(912)는 사용자가 풀 버전의 제스처를 수행했는지 여부에 해당하는 출력을 결정하도록 데이터를 처리하고, 풀 버전의 제스처에 해당하는 출력을 애플리케이션으로 전송하는 것을 묘사한다.
- [0101] 일 실시예에서, 제스처의 슛컷은 제스처의 풀 버전을 포함하는 사용자 움직임의 부분집합을 포함하는 사용자 움직임을 포함한다.
- [0102] 일 실시예에서, 제스처의 슛컷에 해당하는 출력은 사용자가 그 제스처를 일으켰을 높은 가능성에 해당하고, 제스처의 풀 버전에 해당하는 출력은 사용자가 그 제스처를 일으켰을 높은 가능성에 해당하며, 애플리케이션은 오직 한 개의 사용자 제스처만을 인식한다. 제스처의 슛컷이 제스처의 풀 버전의 부분집합을 포함하는 경우, 사용자가 제스처의 풀 버전을 수행할 때 그는 제스처의 슛컷 역시 수행할 것이다. 따라서, 의도된 하나의 제스처 입력에 대해 두 가지 제스처들이 인식될 수 있다. 일 실시예에서 제스처의 슛컷 및 제스처의 풀 버전 둘 모두가 소정 시간 안에 (제스처 및/또는 사용자 고유한 것일 수 있는 것으로) 인식되는 경우, 오직 하나만이 입력으로서 사용되고 나머지 하나는 버려진다.
- [0103] 일 실시예에서, 제스처의 슛컷에 해당하는 출력은 사용자가 그 제스처를 일으켰을 높은 가능성에 해당하고, 제스처의 풀 버전에 해당하는 출력은 사용자가 그 제스처를 일으켰을 높은 가능성에 해당하며, 애플리케이션은 제스처에 세부사항을 더하기 위해 제스처의 풀 버전에 해당하는 출력을 사용한다. 제스처의 슛컷이 애플리케이션에 의해 입력으로서 인식 및 (해당 애니메이션이나 결과가 디스플레이 장치 상에 디스플레이되는 것과 같이) 처리되고, 그런 다음 제스처의 슛컷이 처리되는 동안 제스처의 풀 버전이 인식되는 경우, 제스처의 풀 버전으로부터의 출력이 프로세싱 시에 사용될 수 있다.
- [0104] 예를 들어, 풀 버전 "점프" 제스처는 사용자가 점프하는 것을 포함할 수 있다. 그 "점프" 제스처의 슛컷은 제스처의 풀 버전-구부렀다 일어나기-의 초기 모션을 포함하고 그 제스처가 수행되었다는 신뢰 수준을 출력한다. 점프 제스처의 슛컷을 관찰한 결과로서, 애플리케이션은 사용자 아바타를 점프하는 것으로 디스플레이함으로써 그것을 처리할 수 있다. 이것이 일어나는 동안, 사용자가 계속해서 일어나 두 발을 땅에서 떠나게 함으로써 점프 제스처의 풀 버전을 이행하는 경우, 애플리케이션은 사용자가 물리적으로 점프한 높이를 사용해 아바타를 상응하는 높이 만큼 점프하는 것으로 디스플레이 하여 현재 처리된 점프 슛컷에 세부사항을 추가하도록 할 수 있다. 사용자가 점프 제스처의 슛컷 만을 수행했을 경우, 애플리케이션은 아바타 점프를 디폴트 높이로 할 수 있다.
- [0105] 일 실시예에서, 사용자가 제스처의 슛컷을 수행하는 경우, 그것은 사용자가 제스처의 풀 버전을 수행하는 경우보다 낮은 애플리케이션 실행에 해당할 수 있다. 예를 들어, 포인트가 득점되는 사용자의 퍼포먼스를 순위 매기는 스케이트보딩 게임에서, 사용자가 스케이트보딩 묘기 제스처의 슛컷을 이용해 소정 묘기를 수행한 경우, 사용자는 풀 버전의 스케이트보딩 묘기 제스처를 이용해 소정 묘기를 수행했을 때보다 적은 포인트를 받을 것이다.
- [0106] **결론**
- [0107] 본 개시사항은 여러 도면에 도시된 바와 같은 바람직한 양태와 관련해 기술되었으나, 본 개시사항에서 이탈하지 않고 그와 동일한 기능을 수행하기 위해, 기술된 양태들에 대한 변경과 추가가 이뤄지거나 다른 유사한 양태가 사용될 수 있다는 것을 알 수 있다. 따라서, 본 개시사항은 어느 한 양태에 국한되어서는 안되며, 그보다는 첨부된 청구범위에 따른 폭과 범위 안에서 해석되어야 한다. 예를 들어, 여기 기술된 다양한 절차들은 하드웨어나 소프트웨어, 또는 그 둘의 조합을 이용해 구현될 수 있다. 따라서, 개시된 실시예의 방법 및 장치, 또는 소정 양태나 그 일부는 플로피 디스켓, CD-ROM, 하드 드라이브, 또는 어떤 다른 기계 판독가능 저장 매체 같은 유형 매체 안에서 구현되는 프로그램 코드(즉, 명령어)의 형태를 취할 수 있다. 프로그램 코드가 컴퓨터 같은 기계장치 안에 로드되어 실행될 때, 기계장치는 개시된 실시예를 실시하도록 구성되는 장치가 된다. 여기 개시된 사양을 고려함에 따라, 여기에 명확히 언급된 특정 구현예들 이외에 다른 양태 및 구현예들도 자명해질 것이다. 명세서 및 예시된 구현예들은 예로서만 간주되도록 의도된다.

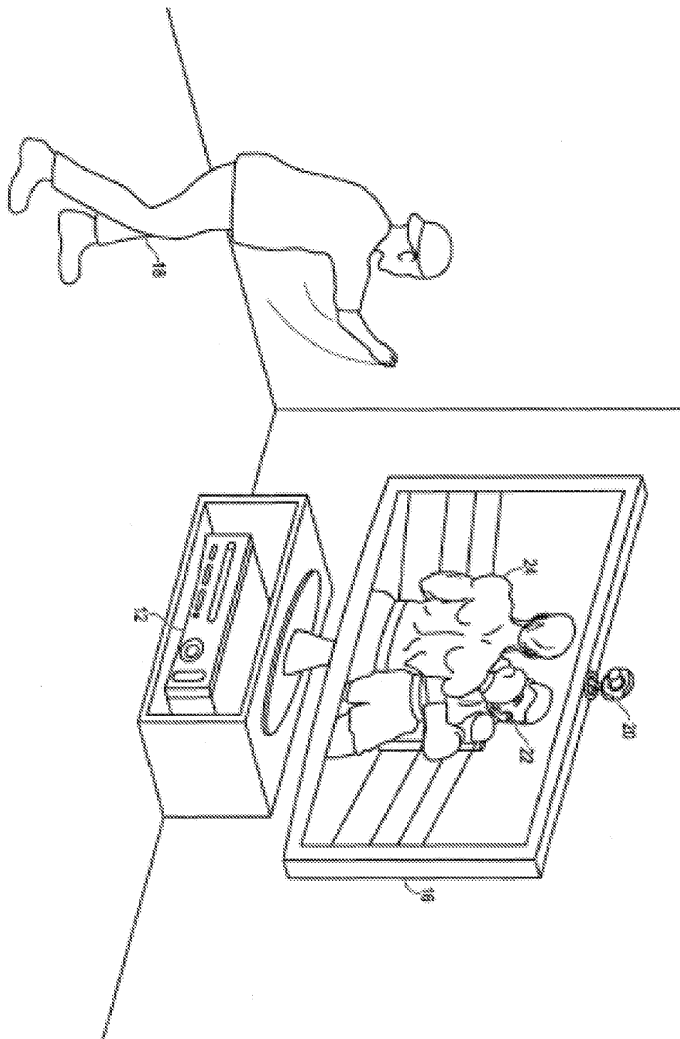
도면

도면1a

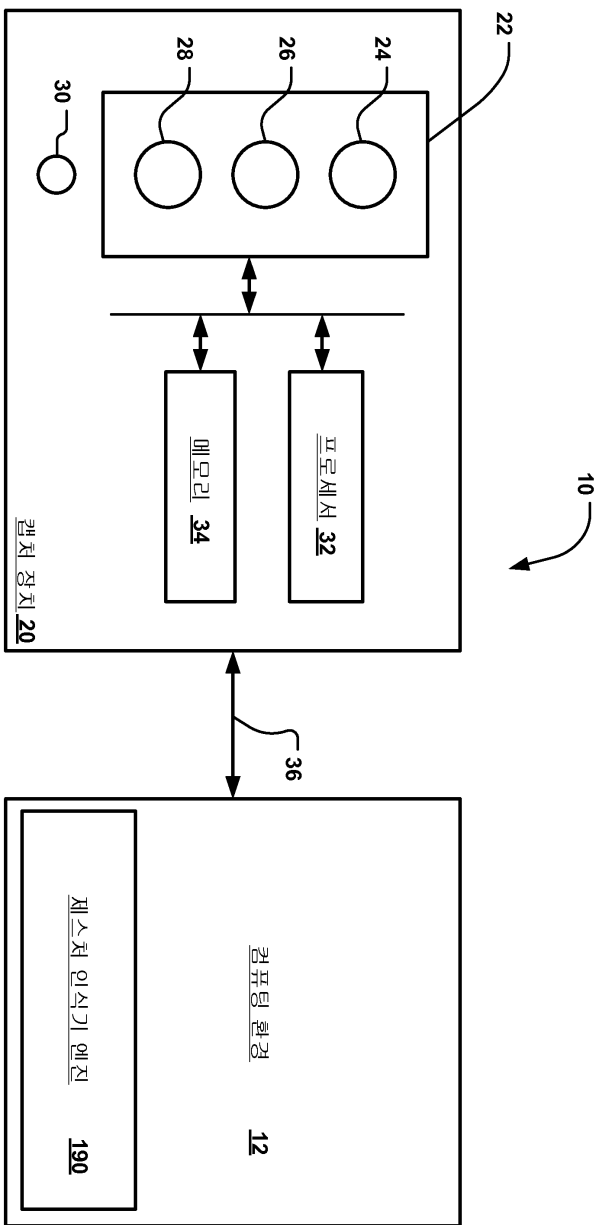




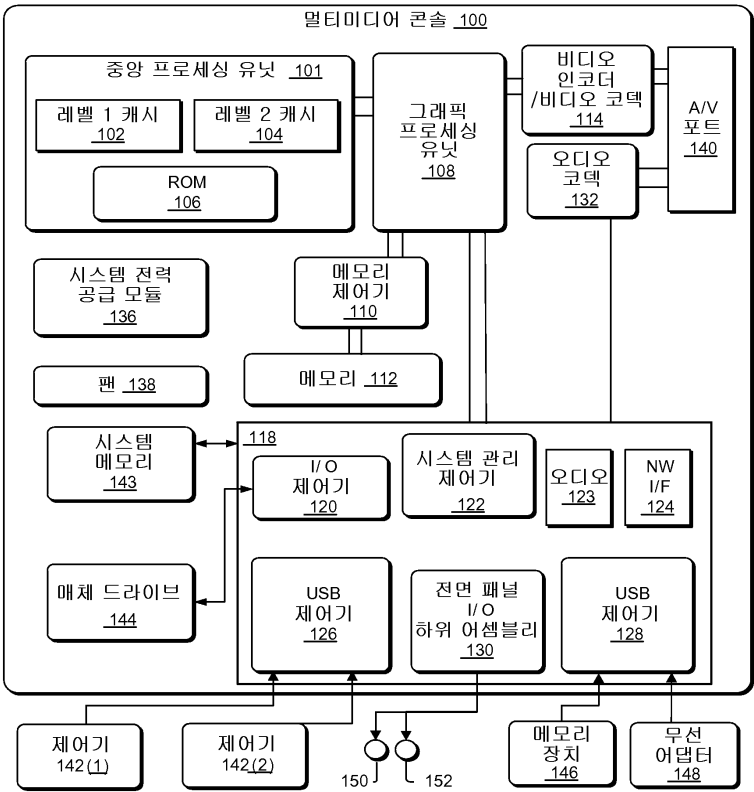
도면1b



도면2

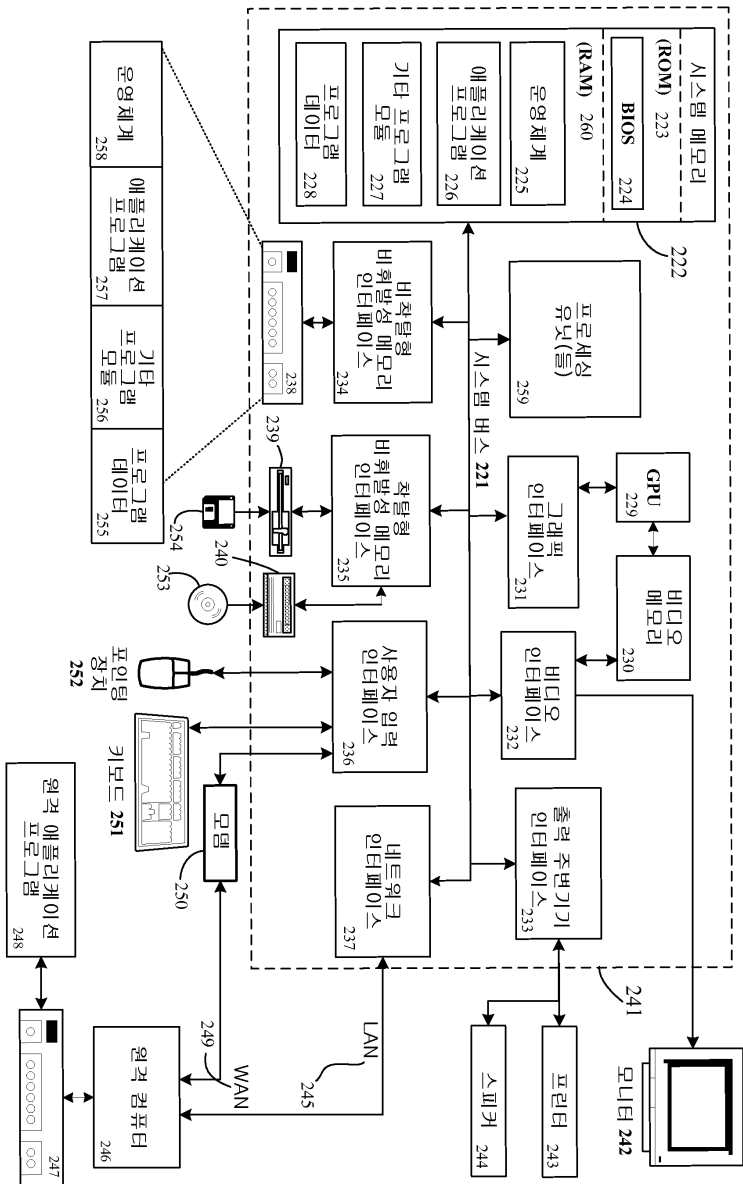


도면3a

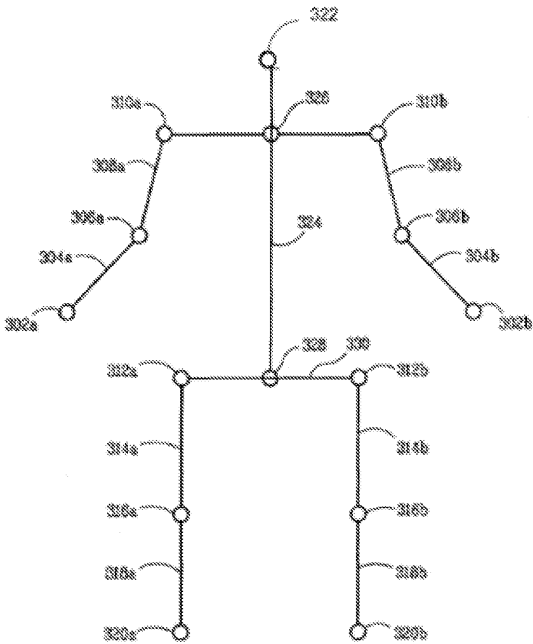




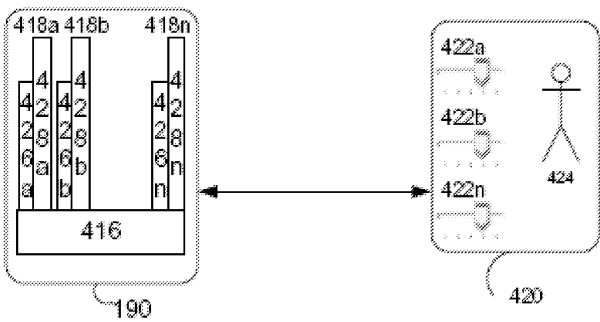
220 文學研究



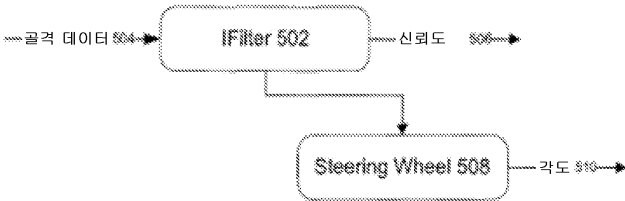
도면4a



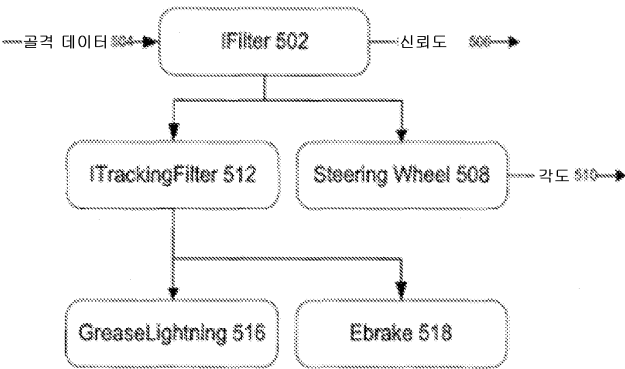
도면4b



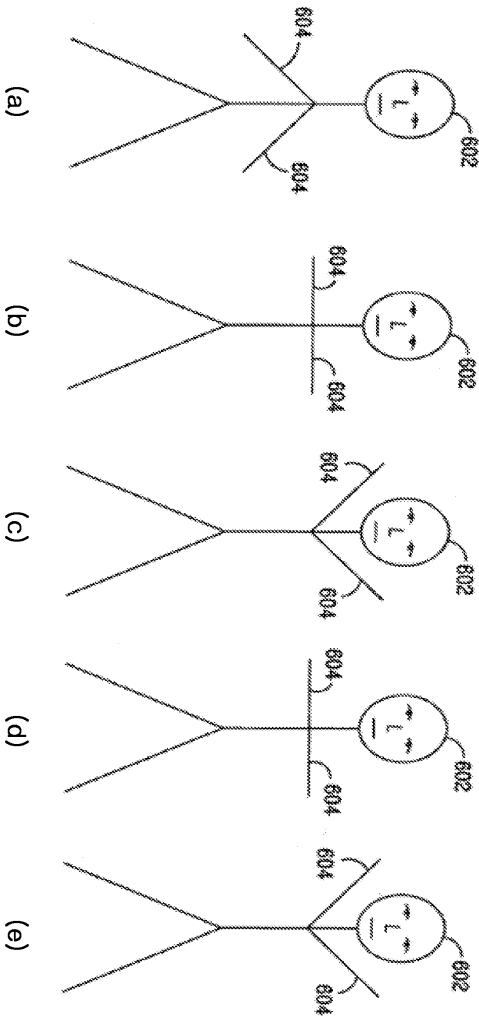
도면5a



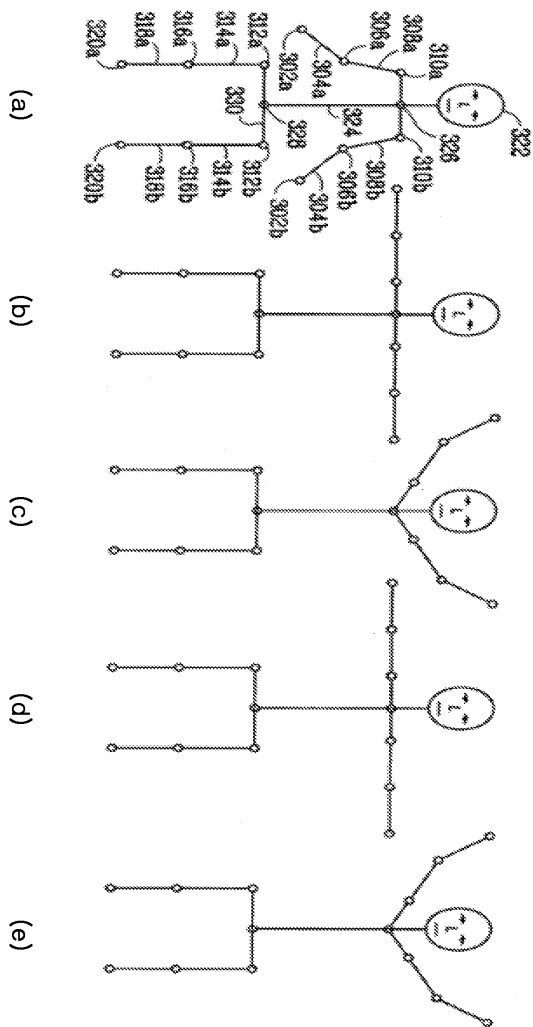
도면5b



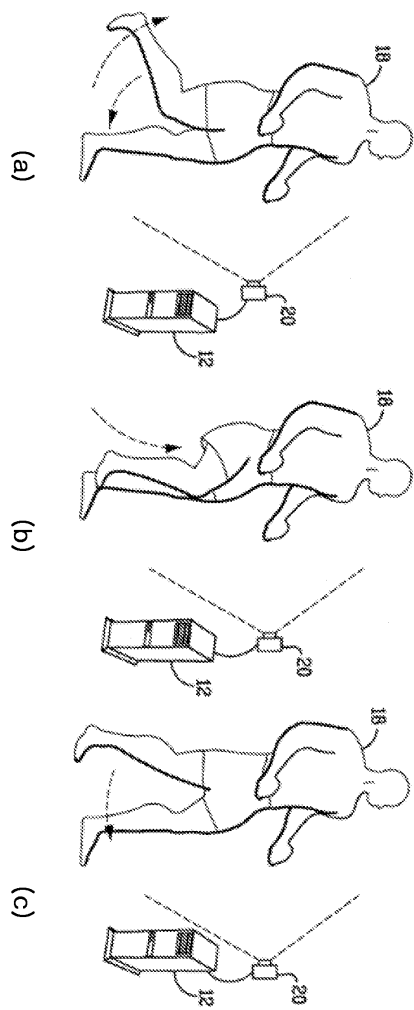
도면6



도면7



도면8



도면9

