



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2019년06월28일  
(11) 등록번호 10-1994500  
(24) 등록일자 2019년06월24일

(51) 국제특허분류(Int. Cl.)  
G06F 3/03 (2006.01)  
(21) 출원번호 10-2014-7010368  
(22) 출원일자(국제) 2012년10월20일  
심사청구일자 2017년09월20일  
(85) 번역문제출일자 2014년04월18일  
(65) 공개번호 10-2014-0078681  
(43) 공개일자 2014년06월25일  
(86) 국제출원번호 PCT/US2012/061225  
(87) 국제공개번호 WO 2013/059752  
국제공개일자 2013년04월25일  
(30) 우선권주장  
13/277,222 2011년10월20일 미국(US)  
(56) 선행기술조사문헌  
US20030222856 A1  
US20060244735 A1  
US20070126743 A1

(73) 특허권자  
마이크로소프트 테크놀로지 라이선싱, 엘엘씨  
미국 워싱턴주 (우편번호 : 98052) 레드몬드 원  
마이크로소프트 웨이  
(72) 발명자  
혼지 스코트  
미국 워싱턴주 98052-6399 레드몬드 원 마이크로  
소프트 웨이 엘씨에이 - 인터내셔널 페이턴츠 마  
이크로소프트 코포레이션  
(74) 대리인  
제일특허법인(유)

전체 청구항 수 : 총 26 항

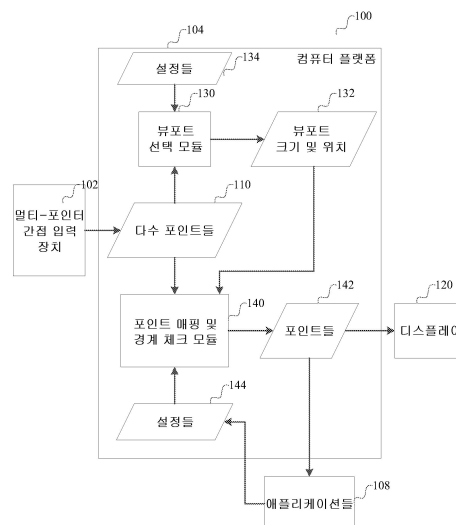
심사관 : 이상현

(54) 발명의 명칭 멀티-포인터 간접 입력 장치들을 위한 가속도-기반 상호작용 기법

(57) 요약

터치 센서와 같은, 하지만 이에만 제한되지는 않는, 간접 상호작용 입력 장치는 다수의 입력 포인트들을 제공할 수 있다. 그러면, 이들 다수의 포인트들은 디스플레이와 같은 출력 장치 상의 다수의 위치들로 매핑된다. 그러나, 다수의 입력 포인트들은 포인터 볼리스틱스의 적용과 입력 센서 및 타겟 디스플레이 사이의 해상도 차이를 더욱 관리하기 어렵게 만든다. 그러므로, 포인트들의 세트의 특성이 식별되고 포인트들 각각의 매핑을 조정하기 위해 사용된다. 예를 들어, 이 문제를 해결하기 위한 한가지 방식은, 그것의 이전 포인트로부터 또는 기준 포인트로부터이든 간에, 이전 프레임으로부터 최소 변위를 가진 입력 포인트를 식별하는 것이다. 이 변위는 입력 장치로부터 각자의 대응하는 디스플레이 좌표들로의 입력 포인트들의 세트의 매핑을 조정하기 위하여 사용된다.

대표도 - 도1



## 명세서

### 청구범위

#### 청구항 1

컴퓨터 구현 방법으로서,

복수의 프레임에 대하여 멀티-포인터 간접 입력 장치로부터 입력 포인트들을 기술하는 정보를 메모리로 수신하는 단계— 각각의 입력 포인트는 센서 좌표 시스템 내의 위치를 가짐 —와,

상기 센서 좌표 시스템 내의 위치들로부터 상기 프레임들 중 하나의 프레임에 대한 상기 입력 포인트들을, 컴퓨터 사용자 인터페이스에서 디스플레이의 디스플레이 좌표 공간 내 위치들로 매핑하는 단계와,

상기 복수의 프레임 중 또 다른 프레임 내의 하나 이상의 다른 포인트에 대해서 상기 프레임들 중 상기 하나의 프레임의 상기 입력 포인트들 각각의 변위를 결정하는 단계와,

가장 낮은 크기의 변위에 따라 오프셋을 결정하고 각각의 입력 포인트에 대해, 상기 입력 포인트의 상기 매핑된 위치에 상기 오프셋을 더함으로써, 상기 가장 낮은 크기의 변위를 갖는 선택된 입력 포인트에 대해 결정된 상기 변위에 따라 상기 프레임들 중 상기 하나의 프레임 내의 상기 매핑된 입력 포인트들의 위치를 수정하는 단계와,

상기 디스플레이에서, 상기 컴퓨터 사용자 인터페이스 내에서 상기 수정된 위치에 상기 입력 포인트들을 나타내는 포인트들을 디스플레이하는 단계

를 포함하는 컴퓨터 구현 방법.

#### 청구항 2

컴퓨터 구현 방법으로서,

복수의 프레임에 대하여 멀티-포인터 간접 입력 장치로부터 입력 포인트들을 기술하는 정보를 메모리로 수신하는 단계— 각각의 입력 포인트는 센서 좌표 시스템 내의 위치를 가짐 —와,

상기 센서 좌표 시스템 내의 위치들로부터 상기 프레임들 중 하나의 프레임에 대한 상기 입력 포인트들을, 컴퓨터 사용자 인터페이스에서 디스플레이의 디스플레이 좌표 공간 내 위치들로 매핑하는 단계와,

상기 복수의 프레임 중 또 다른 프레임 내의 하나 이상의 다른 포인트에 대해서 상기 프레임들 중 상기 하나의 프레임의 상기 입력 포인트들 각각의 변위를 결정하는 단계와,

가장 낮은 크기의 변위에 따라 상기 입력 포인트들에 관련된 기준 포인트에 오프셋을 적용하고 각각의 입력 포인트에 대해, 상기 기준 포인트로부터 상기 입력 포인트의 오프셋에 따라 상기 입력 포인트에 대한 수정된 위치를 결정함으로써, 상기 가장 낮은 크기의 변위를 갖는 선택된 입력 포인트에 대해 결정된 상기 변위에 따라 상기 프레임들 중 상기 하나의 프레임 내의 상기 매핑된 입력 포인트들의 위치를 수정하는 단계와,

상기 디스플레이에서, 상기 컴퓨터 사용자 인터페이스 내에서 상기 수정된 위치에 상기 입력 포인트를 나타내는 포인트들을 디스플레이하는 단계

를 포함하는 컴퓨터 구현 방법.

#### 청구항 3

제 1 항에 있어서,

상기 또 다른 프레임 내의 상기 다른 포인트는 상기 또 다른 프레임 내의 상기 입력 포인트의 위치인

컴퓨터 구현 방법.

#### 청구항 4

제 1 항에 있어서,

상기 또 다른 프레임 내의 상기 다른 포인트는 상기 또 다른 프레임에 대한 센서 로케이터인  
컴퓨터 구현 방법.

#### 청구항 5

제 1 항에 있어서,

상기 결정된 변위가 작은 변위인 것에 응답하여, 상기 매핑된 입력 포인트들의 위치를 수정하는 단계는 상기 입력 포인트들의 변위를 일 프레임에서 그 다음 프레임으로 넘어가면 감소시키는 단계를 포함하는

컴퓨터 구현 방법.

#### 청구항 6

제 1 항에 있어서,

상기 결정된 변위가 큰 변위인 것에 응답하여, 상기 매핑된 입력 포인트들의 위치를 수정하는 단계는 상기 입력 포인트들의 변위를 일 프레임에서 그 다음 프레임으로 넘어가면 증가시키는 단계를 포함하는

컴퓨터 구현 방법.

#### 청구항 7

제 1 항에 있어서,

상기 매핑된 입력 포인트들의 위치를 수정하는 단계는 각 축에 대하여 상이한 가속 함수를 적용하는 단계를 포함하는

컴퓨터 구현 방법.

#### 청구항 8

제 1 항에 있어서,

상기 매핑된 입력 포인트들의 위치를 수정하는 단계는 상기 변위의 크기에 따라 가속을 적용하는 단계를 포함하는

컴퓨터 구현 방법.

#### 청구항 9

제 7 항에 있어서,

상기 가속 함수는 가속화된 변위에 대한 변위의 구간적 선형 함수(piecewise linear function)인

컴퓨터 구현 방법.

#### 청구항 10

컴퓨터에 의해 처리되는 경우에 상기 컴퓨터로 하여금 프로세스를 수행하도록 지시하는 컴퓨터 프로그램 명령어가 저장된 컴퓨터 판독가능 저장 매체로서,

상기 프로세스는

복수의 프레임에 대하여 멀티-포인터 간접 입력 장치로부터 입력 포인트들을 기술하는 정보를 메모리로 수신하는 단계— 각각의 입력 포인트는 센서 좌표 시스템 내의 위치를 가짐 —와,

상기 센서 좌표 시스템 내의 위치들로부터 상기 프레임들 중 하나의 프레임에 대한 상기 입력 포인트들을, 컴퓨터 사용자 인터페이스에서 디스플레이의 디스플레이 좌표 공간 내 위치들로 매핑하는 단계와,

상기 복수의 프레임 중 또 다른 프레임 내의 하나 이상의 다른 포인트에 대해서 상기 프레임들 중 상기 하나의 프레임의 상기 입력 포인트들 각각의 변위를 결정하는 단계와,

가장 낮은 크기의 변위에 따라 오프셋을 결정하고 각각의 입력 포인트에 대해, 상기 입력 포인트의 상기 매핑된 위치에 상기 오프셋을 더함으로써, 상기 가장 낮은 크기의 변위를 갖는 선택된 입력 포인트에 대해 결정된 상기 변위에 따라 상기 프레임들 중 상기 하나의 프레임 내의 상기 매핑된 입력 포인트들의 위치를 수정하는 단계와,

상기 디스플레이에서, 상기 컴퓨터 사용자 인터페이스 내에서 상기 수정된 위치에 상기 입력 포인트들을 나타내는 포인트들을 디스플레이하는 단계

를 포함하는, 컴퓨터 판독가능 저장 매체.

### 청구항 11

컴퓨터에 의해 처리되는 경우에 상기 컴퓨터로 하여금 프로세스를 수행하도록 지시하는 컴퓨터 프로그램 명령어가 저장된 컴퓨터 판독가능 저장 매체로서,

상기 프로세스는

복수의 프레임에 대하여 멀티-포인터 간접 입력 장치로부터 입력 포인트들을 기술하는 정보를 메모리로 수신하는 단계— 각각의 입력 포인트는 센서 좌표 시스템 내의 위치를 가짐 —와,

상기 센서 좌표 시스템 내의 위치들로부터 상기 프레임들 중 하나의 프레임에 대한 상기 입력 포인트들을, 컴퓨터 사용자 인터페이스에서 디스플레이의 디스플레이 좌표 공간 내 위치들로 매핑하는 단계와,

상기 복수의 프레임 중 또 다른 프레임 내의 하나 이상의 다른 포인트에 대해서 상기 프레임들 중 상기 하나의 프레임의 상기 입력 포인트들 각각의 변위를 결정하는 단계와,

가장 낮은 크기의 변위에 따라 상기 입력 포인트들에 관련된 기준 포인트에 오프셋을 적용하고 각각의 입력 포인트에 대해, 상기 기준 포인트로부터 상기 입력 포인트의 오프셋에 따라 수정된 위치를 결정함으로써, 상기 가장 낮은 크기의 변위를 갖는 선택된 입력 포인트에 대해 결정된 상기 변위에 따라 상기 프레임들 중 상기 하나의 프레임 내의 상기 매핑된 입력 포인트들의 위치를 수정하는 단계와,

상기 디스플레이에서, 상기 컴퓨터 사용자 인터페이스 내에서 상기 수정된 위치에 상기 입력 포인트들을 나타내는 포인트들을 디스플레이하는 단계

를 포함하는, 컴퓨터 판독가능 저장 매체.

### 청구항 12

제 10 항에 있어서,

상기 매핑된 입력 포인트들의 위치를 수정하는 단계는 각 축에 대하여 상이한 가속 함수를 적용하는 단계를 포함하는

컴퓨터 판독가능 저장 매체.

### 청구항 13

제 12 항에 있어서,

상기 가속 함수는 가속화된 변위에 대한 변위의 구간적 선형 함수(piecewise linear function)인 컴퓨터 판독가능 저장 매체.

#### 청구항 14

컴퓨터로서,

하나 이상의 프로세서와,

상기 프로세서에 연결된 컴퓨터 판독가능 저장소를 포함하되,

상기 저장소에는 컴퓨터 프로그램 명령어가 저장되고, 상기 프로세서가 상기 명령어를 처리하는 경우, 상기 컴퓨터는

복수의 프레임에 대하여 멀티-포인터 간접 입력 장치로부터 입력 포인트들을 기술하는 정보를 메모리로 수신하고— 각각의 입력 포인트는 센서 좌표 시스템 내의 위치를 가짐 —,

상기 센서 좌표 시스템 내의 위치들로부터 상기 프레임들 중 하나의 프레임에 대한 상기 입력 포인트들을, 컴퓨터 사용자 인터페이스에서 디스플레이의 디스플레이 좌표 공간 내 위치들로 매핑하고,

상기 복수의 프레임 중 또 다른 프레임 내의 하나 이상의 다른 포인트에 대해서 상기 프레임들 중 상기 하나의 프레임의 상기 입력 포인트들 각각의 변위를 결정하고,

가장 낮은 크기의 변위에 따라 오프셋을 결정하고 각각의 입력 포인트에 대해, 상기 입력 포인트의 상기 매핑된 위치에 상기 오프셋을 더함으로써, 상기 가장 낮은 크기의 변위를 갖는 선택된 입력 포인트에 대해 결정된 상기 변위에 따라 상기 프레임들 중 상기 하나의 프레임 내의 상기 매핑된 입력 포인트들의 위치를 수정하며,

상기 디스플레이에서, 상기 컴퓨터 사용자 인터페이스 내에서 상기 수정된 위치에 상기 입력 포인트들을 나타내는 포인트들을 디스플레이하도록 구성되는

컴퓨터.

#### 청구항 15

컴퓨터로서,

하나 이상의 프로세서와,

상기 프로세서에 연결된 컴퓨터 판독가능 저장소를 포함하되,

상기 저장소에는 컴퓨터 프로그램 명령어가 저장되고, 상기 프로세서가 상기 명령어를 처리하는 경우, 상기 컴퓨터는

복수의 프레임에 대하여 멀티-포인터 간접 입력 장치로부터 입력 포인트들을 기술하는 정보를 메모리로 수신하고— 각각의 입력 포인트는 센서 좌표 시스템 내의 위치를 가짐 —,

상기 센서 좌표 시스템 내의 위치들로부터 상기 프레임들 중 하나의 프레임에 대한 상기 입력 포인트들을, 컴퓨터 사용자 인터페이스에서 디스플레이의 디스플레이 좌표 공간 내 위치들로 매핑하고,

상기 복수의 프레임 중 또 다른 프레임 내의 하나 이상의 다른 포인트에 대해서 상기 프레임들 중 상기 하나의 프레임의 상기 입력 포인트들 각각의 변위를 결정하고,

가장 낮은 크기의 변위에 따라 상기 입력 포인트들에 관련된 기준 포인트에 오프셋을 적용하고 각각의 입력 포인트에 대해, 상기 기준 포인트로부터 상기 입력 포인트의 오프셋에 따라 상기 입력 포인트에 대한 수정된 위치를 결정함으로써, 상기 가장 낮은 크기의 변위를 갖는 선택된 입력 포인트에 대해 결정된 상기 변위에 따라 상기 프레임들 중 상기 하나의 프레임 내의 상기 매핑된 입력 포인트들의 위치를 수정하며,

상기 디스플레이에서, 상기 컴퓨터 사용자 인터페이스 내에서 상기 수정된 위치에 상기 입력 포인트들을 나타내는 포인트들을 디스플레이하도록 구성되는

컴퓨터.

#### 청구항 16

제 14 항에 있어서,

상기 매핑된 입력 포인트들의 위치를 수정하기 위해, 상기 컴퓨터는 상기 변위의 크기에 따라 가속 함수를 적용하도록 더 구성되는

컴퓨터.

#### 청구항 17

제 16 항에 있어서,

상기 가속 함수는 가속화된 변위에 대한 변위의 구간적 선형 함수인

컴퓨터.

#### 청구항 18

제 14 항에 있어서,

상기 또 다른 프레임 내의 상기 다른 포인트는 상기 또 다른 프레임을 위한 센서 로케이터인

컴퓨터.

#### 청구항 19

제 14 항에 있어서,

작은 변위에 응답하여, 상기 컴퓨터는 상기 입력 포인트들의 변위를 일 프레임에서 그 다음 프레임으로 넘어가면 감소시킴으로써 상기 입력 포인트들의 매핑된 위치를 수정하도록 구성되는

컴퓨터.

#### 청구항 20

제 14 항에 있어서,

큰 변위에 응답하여, 상기 컴퓨터는 상기 입력 포인트들의 변위를 일 프레임에서 그 다음 프레임으로 넘어가면 증가시킴으로써 상기 입력 포인트들의 매핑된 위치를 수정하도록 구성되는

컴퓨터.

#### 청구항 21

제 15 항에 있어서,

상기 입력 포인트들의 매핑된 위치를 수정하기 위해, 상기 컴퓨터는 각 축에 대하여 상이한 가속 함수를 적용하도록 구성되는

컴퓨터.

#### 청구항 22

제 21 항에 있어서,

상기 가속 함수는 가속화된 변위에 대한 변위의 구간적 선형 함수인

컴퓨터.

### 청구항 23

제 2 항에 있어서,

상기 입력 포인트들의 매핑된 위치를 수정하는 단계는 각 축에 대하여 상이한 가속 함수를 적용하는 단계를 포함하는

컴퓨터 구현 방법.

### 청구항 24

제 23 항에 있어서,

상기 가속 함수는 가속화된 변위에 대한 변위의 구간적 선형 함수인

컴퓨터 구현 방법.

### 청구항 25

제 11 항에 있어서,

상기 입력 포인트들의 매핑된 위치를 수정하는 단계는 각 축에 대하여 상이한 가속 함수를 적용하는 단계는 포함하는

컴퓨터 판독가능 저장 매체.

### 청구항 26

제 25 항에 있어서,

상기 가속 함수는 가속화된 변위에 대한 변위의 구간적 선형 함수인

컴퓨터 판독가능 저장 매체.

## 발명의 설명

### 기술 분야

- [0001] 컴퓨팅 시스템의 네비게이션 및 공간적 제어를 위해 사용되는 수동 입력 장치들은 컴퓨터 시스템의 성능과 사용자의 전체 경험에 상당한 영향을 준다. 몇 가지 종류의 수동 입력 장치들이 존재한다. 개인용 컴퓨터를 위한 것들 중 가장 통상적인 것에는 마우스 또는 트랙패드(trackpad)와 같은 단일-포인트의 간접 상호작용 장치들, 및 터치 스크린과 같은 직접 상호작용 장치들을 포함한다.
- [0002] 단일-포인트의 간접 상호작용 장치 센서는 센서를 통해 사용자 상호작용을 검출하고 이 상호작용을 디스플레이 상의 어느 한 위치로 매핑한다. 입력 포인트들을 디스플레이로 매핑하는 한 방법은 센서 범위를 직접 디스플레이의 범위로 일대일 매핑하는 것을 포함하는데, 이것은 절대 매핑(absolute mapping)이라 불리운다. 절대 매핑을 채용하는 장치들의 예는 펜 및 터치 디지털라이저(digitizer)이다. 다른 방법은 디스플레이의 장치 센서 좌표를 이동가능 서브포션(subportion)으로 매핑하는 것을 포함하는데, 이것은 상대 매핑(relative mapping)이라 불리운다.
- [0003] 상대 매핑을 채용하는 장치들의 예는 마우스 및, 트랙패드와 같이 마우스를 에뮬레이트하는 장치이다. 마우스는 움직임을 감지하며, 감지된 장치와의 상호작용에 기초하여 정해진 시작 위치를 변경한다. 트랙패드는 통상적으로 마우스와 유사한 방식으로 사용된다. 트랙패드 상에서 접촉 움직임이 감지되면, 감지된 움직임은 마우스 입력과 유사한 방식으로 취급된다.
- [0004] 직접 상호작용 장치는 디스플레이와 시각적으로 정렬되어 있는 장치와의 상호작용을 허용한다. 직접 상호작용 장치는, 절대 매핑을 사용하여, 터치 민감형 표면 상의 위치들과 동일 크기의 디스플레이 상의 위치들 사이에 매핑한다. 예를 들어, 사용자가 터치스크린 상의 한 포인트를 터치하면, 입력 이벤트는 사용자에게 의해 터치된 디스플레이 상의 포인트에 대응하는 사용자 인터페이스 내 위치에서 명령 활성화와 같은 애플리케이션 응답을

트리거할 수 있다.

[0005] 디스플레이에 대한 멀티-포인터 입력 장치로부터 공간적인 입력의 절대 매핑 및 상대 매핑은, 입력 장치와 디스플레이 장치의 물리적인 속성들, 시스템의 성능들, 애플리케이션 사용자 인터페이스의 성질과 레이아웃, 사용자가 실행시키는 태스크의 종류, 및 다양한 인간공학적 팩터들에 따라, 선택적인 장점들과 단점들을 가진다.

[0006] 예를 들어, 상대 매핑 모드에서, 입력 장치 센서는 타겟 디스플레이 좌표 공간의 서브세트만을 스캔(span)할 수 있다. 그러므로, 하나의 디스플레이 위치에서 다른 하나의 위치로의 네비게이션은, 입력 포인트들의 움직임이 검출될 때 포인트들의 어떤 가속 형태가 적용되지 않는 한, 다수의 스트로크를 포함할 수 있다. 반대로, 픽셀-레벨의, 포인트-대-포인트 타겟팅 정밀도를 달성하기 위하여, 포인트들의 감속 형태가 적용될 수 있다. 이러한 가속 및 감속은 종종 "포인터 볼리스틱(pointer ballistics)"이라고 지칭된다. 다른 한 예를 들면, 절대 매핑 모드에서, 입력 센서는 디스플레이 좌표 공간에 대해서 훨씬 더 작은 해상도를 가지거나 또는 종횡비(aspect ratio)에서 더 큰 차이를 가질 수 있다. 그러므로, 픽셀-레벨 포인트-대-포인트 네비게이션의 정밀도 성취는 어떤 형태의 양의 및/또는 음의 가속을 고려하지 않으면 어려울 수 있다.

## 발명의 내용

[0007] 본 요약(summary)은, 아래의 발명을 실시하기 위한 구체적인 내용(Detailed Description)에서 더 기술되는 단순화된 개념을 소개하기 위하여 제공된다. 본 요약은 청구된 발명 주제의 주요 특징들 또는 본질적인 특징들을 식별하는 것으로 의도되지 않으며, 또한 청구된 발명 주제의 범위를 결정하는데 사용되는 것으로도 의도되지 않는다.

[0008] 간접 상호작용 입력 장치는 하나 이상의 입력 센서들을 포함할 수 있으며, 각 센서는 다수의 입력 포인트를 검출하고 제공할 수 있다. 해당 기술분야에서 알려져 있는 다양한 멀티-포인트 입력 센싱 기술들에는 용량성, 저항성, 및 압력-기반 터치 센서, 광 센서, 및 모션-비디오 골격 추적 시스템이 포함되지만 이들로 제한되지는 않는다. 디스플레이와 같은 출력장치로 센서 입력 포인트들을 매핑하는데 적용되는 계산은, 센서가 하나의 입력 포인트(2차원 좌표 공간 내의 하나의 개별 위치가 아닌)를 정의하는 방식 및 입력 포인트가 감지되는 방식에 대해 독립적이다. 예를 들어, 터치-민감형 센서는 사용자가 센서를 터치하는 좌표 공간 내에서 두 개 이상의 위치들을 지시하는 데이터를 제공할 수 있다. 이러한 센서는 사각형의 형상일 수 있지만, 그 외 다른 형상들을 가질 수도 있다. 센서는 트랙패드와 유사하지만, 하나의 단일 포인트의 추적 움직임이 아니라, 사용자에 의해 터치되는 다수의 포인트들을 검출하는 트랙패드와 유사하게 보여질 수 있다. 그 다음에 이들 다수의 포인트들은 디스플레이와 같은 출력 장치 상의 다수의 위치들로 매핑된다.

[0009] 그러나, 다수의 입력 포인트들은 입력 센서와 타겟 디스플레이 사이의 포인터 볼리스틱 및 해상도 차이의 적용을 관리하기 더 어렵게 만든다. 따라서, 포인트들로 된 세트의 특성이 식별되어 포인트 각각의 매핑을 적용하는데 사용된다. 예를 들어, 이러한 문제를 해결하는 한 가지 방식은, 그것의 이전 위치로부터이든 또는 기준 포인트로부터이든, 이전 프레임으로부터 최소 변위(the least displacement)를 가지는 입력 포인트를 식별하는 것이다. 이 변위는 센서의 입력 포인트들의 세트로부터 각각의 대응하는 디스플레이 좌표들의 매핑을 조정하는데 사용된다.

[0010] 따라서, 일 양상에서, 멀티-포인터 간접 입력 장치 센서 상의 입력 포인트들을 기술하는 정보가 메모리로 수신된다. 입력 포인트들은 디스플레이를 위한 디스플레이 좌표 공간 내의 위치들로 매핑된다. 시간적으로 두 개의 포인트들 사이에서 입력 포인트들의 변위가 결정된다. 입력 포인트들의 매핑은 결정된 변위에 따라 수정된다. 포인트들은 입력 포인트들의 수정된 매핑에 따라 디스플레이 상에 디스플레이된다.

[0011] 아래의 설명에서, 본 명세서의 일부를 형성하는 첨부된 도면들에 대한 참조가 이루어지며, 도면에는 본 기법의 특정한 예시적인 구현예들이, 예시적으로 도시되어 있다. 그 외 다른 실시예들이 사용될 수 있으며 개시된 내용의 범위로부터 벗어나지 않으면서 구조적인 변화가 이루어질 수 있다는 것이 이해된다.

## 도면의 간단한 설명

[0012] 도 1은 멀티-포인터 간접 입력 장치를 사용하는 시스템의 블록도이다.



도 2는 뷰포트(viewport) 배치의 예시적인 구현예를 도시하는 흐름도이다.

도 3은 입력 매핑의 예시적인 구현예를 도시하는 흐름도이다.

도 4는 입력 매핑의 다른 예시적인 구현예를 도시하는 흐름도이다.

도 5는 입력 가속의 예시적인 구현예를 도시하는 흐름도이다.

도 6는 스캔 조정의 예시적인 구현예를 도시하는 흐름도이다.

도 7은 그러한 시스템이 구현될 수 있는 예시적인 컴퓨팅 장치의 블록도이다.

### 발명을 실시하기 위한 구체적인 내용

- [0013] 아래의 섹션은 멀티-포인터 간접 입력 장치가 사용될 수 있는 예시적인 동작 환경을 제공한다.
- [0014]도 1을 참조하면, 컴퓨터 시스템(100)은, 센서를 구비하고 컴퓨터 플랫폼(104)(이것의 일 예의 디테일은 아래에서 기술된다)에 연결되어 있는 멀티-포인터 간접 입력 장치(102)를 포함한다. 이러한 컴퓨터 시스템은 개인용 컴퓨터, 홈 엔터테인먼트 시스템, 프로젝터, 키오스크 애플리케이션, 컴팩 개인용 전자장치, 또는 기타등등일 수 있다. 컴퓨터 플랫폼은 하나 이상의 애플리케이션(108)과, 멀티포인터 간접 입력 장치를 포함하는 그것의 주변 장치들과 같은, 컴퓨터 플랫폼(104)의 자원들 사이의 상호작용을 관리하는 운영 시스템을 가진다.
- [0015]운영 시스템 내에서, 다수의 감지된 입력 포인트들(110)을 기술하는 데이터가 멀티포인터 간접 입력 장치(102)의 센서로부터 수신된다. 이들 입력 포인트들은 그들을 디스플레이(120) 상의 포인트들에게 매핑하기 위해 프로세싱된다.
- [0016]이 매핑 프로세스는, 상대 매핑이거나 절대 매핑일 수 있는, 디스플레이 좌표 시스템으로의 장치 좌표 시스템의 초기 매핑을 결정하고, 이후 디스플레이 좌표 시스템으로의 장치 좌표 시스템의 각 포인트의 매핑을 결정하는 것을 포함한다. 이러한 초기 매핑은 각 입력 세션의 초기에 발생한다.
- [0017]하나의 입력 세션은 첫번째 입력이 센서에 의해 검출된 시간 포인트에서부터 마지막 입력이 센서로부터 제거된 시간 포인트까지이다. 하나의 입력 세션 동안, 입력 포인트들은 대개 이동한다. 입력 포인트들은 센서 좌표 시스템 내의 각자의 새로운 위치들로부터 디스플레이 좌표 시스템 내의 대응하는 새로운 위치들로 매핑된다. 이러한 이동 매핑(movement mapping)은 바운딩(bounding) 및 가속과 마찬가지로 고려되는 이슈가 될 수 있다.
- [0018]도 1에 도시된 바와 같이, 다수 포인트들(110)은 입력 세션의 초기에 뷰포트 선택 모듈(130)로 입력된다. 뷰포트 선택 모듈은 출력으로서 디스플레이 좌표 시스템 내의 뷰포트 크기 및 위치(132)를 제공한다. 뷰포트(viewport)는 센서 좌표 공간이 매핑되는 디스플레이 좌표 내의 일 영역을 정의한다. 다수 장치 센서들이 시스템에 연결되어 있는 구성에서, 각각의 센서는 각자 자신의 뷰포트를 가진다. 뷰포트는 입력 장치 센서의 형상에 대응하는 형상을 가질 수 있다. 그렇지만, 몇몇 구현예에서, 뷰포트는 센서에서와는 상이한 종횡비 또는 배향을 가질 수 있고, 또는 심지어 상이한 형상을 가질 수도 있다. 예를 들어, 타원 센서가 사각형 뷰포트로 매핑될 수 있다. 뷰포트의 형상은 통상적으로 호스트 시스템에 의해 정의되지만, 또한 장치에 의해 또는 사용자에게 의해 정의될 수도 있다. 뷰포트의 크기 및 위치는 센서에 의해 사용자 입력이 검출될 때 계산된다. 센서에 의해 사용자 입력이 검출되지 않은 때에는 뷰포트의 크기와 위치는 정의되지 않는다. 뷰포트는 통상적으로 사용자에게 디스플레이되지 않는다. 뷰포트의 형상, 크기 및 위치는 함께 센서 좌표 시스템의 디스플레이 좌표 시스템으로의 매핑을 나타낸다. 설정들(134)은, 예컨대 상대 매핑 또는 절대 매핑에 의해 이러한 매핑이 어떻게 수행되는지를 결정하며, 이것의 예들은 아래에서 더 상세히 기술된다.
- [0019]다수 포인트들(110)은 또한 입력 세션 전체에 걸쳐 입력 매핑 모듈(140)에 입력된다. 입력 매핑 모듈은 그것의 출력으로서 디스플레이 좌표 시스템 내의 다수 포인트들(142)을 제공한다. 설정들(144)은, 예컨대 상대적인 입력 위치들을 해석하기 위한 장치 및 디스플레이 기준 위치들을 결정하고, 입력 모션 가속, 스캔 조정, 및 바운딩 조건을 적용함으로써 각각의 포인트가 어떻게 매핑되는지를 결정하며, 이것의 예들은 아래에서 더 상세히 기술된다.
- [0020]다수 포인트들이 디스플레이 좌표 시스템으로 매핑되면, 다수 포인트들(142)이 디스플레이 상에 디스플레이될 수 있다. 각각의 포인트는 운영 시스템(106) 및/또는 애플리케이션들(108)에 의해 예컨대 디스플레이된 아이템의 선택에 있어서 임의의 단일 포인트와 유사한 방식으로, 또는 예컨대 호스트 시스템 사용자 인터페이스 내의 일 요소의 줌, 회전 또는 이동을 실행시키는데 있어서 직접 터치 입력 센서들로부터의 다수 포인트들과 유사한

방식으로 취급될 수 있다. 다수 포인트들의 가능한 사용 범위는, 일단 디스플레이로 매핑되면, 본 발명의 제한이 아니다.

- [0021] 이러한 맥락에서, 이제 도 2 내지 도 4를 참조하여 다수 포인트들의 디스플레이로의 매핑의 예시적인 구현예가 더 상세히 기술될 것이다.
- [0022] 도 2에서, 흐름도는 뷰포트 크기 및 위치가 뷰포트 선택 모듈에 의해 어떻게 선택될 수 있는지, 그리고 이어서 어떻게 포인트들이 매핑될 수 있는지에 대한 예시적인 구현예를 기술한다.
- [0023] 아래의 구현예는 바람직한 사용자 경험에 관한 특정 디자인 결정에 기초하고 있음이 주목되어야 한다. 예를 들어, 그 외 다른 물리적인 입력들에 대한 각 물리적인 입력의 상대적인 위치가 디스플레이로의 프로젝션시 유지된다고 간주된다. 또한 모든 입력들 사이의 거리들은 대칭적으로 스케일(scale)된다고 간주된다.
- [0024] 사용자 경험의 다른 한 양상은 입력 장치와 디스플레이 사이의 매핑 종류이다. 매핑은 상대적이거나 절대적일 수 있으며, 또한 각각의 축에 대하여 독립적일 수 있다. 예를 들어, 상대적인 매핑은 y축에 적용되고 다른 한편으로 x축에 절대 매핑이 적용될 수 있으며, 그 역도 마찬가지이다. 또한 두 축은 서로 다른 상대 매핑을 사용할 수 있다. 또한 매핑은 입력 장치 및 디스플레이의 논리적인 좌표에 기초될 수도 있고 또는 물리적인 차원에 기초될 수도 있다. 만약 매핑이 장치들의 물리적인 차원에 기초한다면, 공간적인 정확도가 향상되어, 더욱 직관적이고 인지적으로 효율적인 인터페이스를 제공할 수 있다. 이들 매핑 종류에 관한 결정은 시스템 내의 선택적인 설정일 수 있다.
- [0025] 사용자 경험의 다른 하나의 양상은 바운딩 정책(bounding policy)이다. 더 상세하게는, 장치 입력들은 시스템을 위한 디스플레이 바운딩 정책을 따를 수 있다. 예를 들어, 모든 장치 입력들이 디스플레이 내에 있도록 강제되거나, 또는 세트로부터 오직 하나의 장치 입력만이 디스플레이 내에 있도록 강제될 수 있다. 다른 하나의 구현예는 어떠한 바운딩 정책도 사용하지 않을 수 있다. 이들 바운딩 정책에 관한 결정은 시스템 내의 선택적인 설정일 수 있다.
- [0026] 뷰포트 크기 및 위치는 각 입력 세션의 초기에 결정된다. 예를 들어, 사용자 입력이 없는 기간 이후 센서에 의해 하나 이상의 입력 포인트들이 검출되면, 입력 세션의 시작이 검출된다(200). 각 축에서 뷰포트 차원들은 입력 장치, 호스트 시스템, 또는 사용자에게 의해 정의될 수 있다. 차원들은 거리의 타겟 디스플레이 장치의 퍼센티지나 또는 물리적인 단위들 중 어느 하나로 표현될 수 있다. 사용될 거리의 물리적인 단위들에 있어서, 입력 센서와 디스플레이 둘 모두의 물리적이고 논리적인(좌표) 범위는 예를 들어 장치에 의해, 사용자 입력에 의해 또는 그 외 다른 수단에 의해 제공된다. 그런 다음 디스플레이 좌표 공간 내에서 출력 로케이터의 위치가 획득된다(201). 이 구현예에서, 출력 로케이터 위치는 사용자 세션(사용자가 로그인하면 시작하고 사용자가 로그오프하면 종료하는) 전체에 걸쳐 적용된다. 출력 로케이터 위치는 시스템에 연결되어 있는 다수의 싱글-포인트 및 멀티-포인트 입력 장치들에 의해 공유되고 갱신된다. 출력 로케이터는 이전 입력 세션에서 저장된 위치일 수 있다. 만약 이전 입력 세션이 없다면, 디스플레이 장치의 중심, 마우스 또는 그 외 다른 장치의 마지막 위치, 또는 대안적인 디폴트 디스플레이 위치가 출력 로케이터 위치로서 사용될 수 있다.
- [0027] 그 다음에, 디스플레이 장치 및 입력 장치의 알려져 있는 파라미터들 즉 좌표들 및 경계들이 주어지면, 각각의 축에 대한 스케일링 팩터들이 결정된다(202). 이들 파라미터들은 통상적으로 메모리 내에 저장된다. 디스플레이 장치의 경우, 파라미터들은 시스템 API의 것을 사용하여 획득될 수 있다. 입력 장치의 경우, 파라미터들은 장치 조회(device interrogation)를 통해 획득될 수 있다. 디스플레이 및 입력 장치들의 좌표와 경계가 주어지면, 스케일링 팩터들이 결정된다. 만약 절대 매핑이 사용된다면, 물리적인 범위들에 기초하는 계산은 필요하지 않으며, 그리고 x축 및 y축 스케일 팩터는 장치와 디스플레이 좌표 범위들의 일-대-일 비율에 기초한다. 만약 상대 매핑이 사용되면, x축 및 y축 스케일 팩터들은 디스플레이 좌표들 내의 뷰포트 차원들에 대한 장치 자원들의 비율에 의해 결정된다. 스케일 팩터들은 한번 계산되고 메모리에 저장된 후 필요할 때면 획득될 수 있다.
- [0028] 디스플레이 좌표 공간 내의 뷰포트 범위, 즉 뷰포트 꼭지점들의 x좌표 및 y좌표는 결정된 스케일 팩터를 사용하여 결정된다(203). 뷰포트 범위는 초기에, 아래와 같이 새로운 출력 로케이터가 계산되기 전에, 저장된 출력 로케이터를 사용하여 입력 세션 동안 결정된다.
- [0029] 디스플레이의 픽셀 밀도를 사용하는 스케일된 뷰포트에 있어서, 스케일 팩터들  $S_v$ 는 비-제로(non-zero)이고 0과 1 사이의 양의 값이고, 뷰포트의 범위( $R_v$ )는 다음과 같다:

$$R_v = \{ \begin{aligned} &L_{V0x} - S_{Vx}/[2 * \text{extent}(R_{Dx})], \\ &L_{V0x} + S_{Vx}/[2 * \text{extent}(R_{Dx})], \\ &L_{V0y} - S_{Vy}/[2 * \text{extent}(R_{Dy})], \\ &L_{V0y} + S_{Vy}/[2 * \text{extent}(R_{Dy})] \end{aligned} \},$$

[0030]

[0031]

여기서  $L_{V0}$ 는 초기 뷰포트 로케이터로서, 통상적으로 타겟 디스플레이의 중심이고,  $S_v$ 는 스케일 팩터들이며 범위( $R_0$ )는 디스플레이의 x좌표 및 y좌표 범위, 즉 그것의 픽셀 폭 및 높이이고, 아래첨자 중 x 및 y는 x축 및 y축에서의 해당 값들을가리킨다.

[0032]

물리적인 차원을 사용하는 뷰포트에 있어서, 바람직한 크기  $S_v$ 는 비-제로이며 타겟 디스플레이의 물리적인 범위보다 크지 않는 양의 값이며, 디스플레이의 픽셀 밀도  $D$ 는 하드웨어 질문을 통해 알려지고, 뷰포트 범위( $R_v$ )는 다음과 같다:

$$R_v = \{ \begin{aligned} &[L_{V0x} - S_{Vx}/[2 * \text{extent}(R_{Dx})]] * D_x + R_{Dx.left}, \\ &[L_{V0x} + S_{Vx}/[2 * \text{extent}(R_{Dx})]] * D_x + R_{Dx.right}, \\ &[L_{V0y} - S_{Vy}/[2 * \text{extent}(R_{Dy})]] * D_y + R_{Dy.top}, \\ &[L_{V0y} + S_{Vy}/[2 * \text{extent}(R_{Dy})]] * D_y + R_{Dy.bottom} \end{aligned} \},$$

[0033]

[0034]

뷰포트의 초기 범위가 주어지면, 초기에 장치 좌표 시스템 내에서, 센서 로케이터가 결정된다(204). 센서 로케이터를 선택하는 많은 방식들이 존재하며, 선택된 특정 방식은 바람직한 사용자 상호작용에 종속한다. 예를 들어, 만약 센서에 의해 검출된 하나의 단일 입력이 존재한다면, 센서 로케이터는 이 단일 입력의 좌표들일 수 있다. 만약 다수의 입력들이 존재한다면, 센서 로케이터는 하나의 단일 "프라이머리(primary)" 입력의 위치이거나, 또는 모든 입력들의 지리적 중심과 같이, 그 외 다른 입력들에 대해 특별한 관계를 가지는 하나의 포인트의 위치일 수 있다. 센서 로케이터는 입력 포인트들이 검출되지 않으면 정의되지 않으며 입력 세션들 사이에 지속되지 않는다.

[0035]

프라이머리 입력의 위치가 센서 로케이터로서 사용되면, 입력에 대해 프라이머리 상태를 선택하고 할당하기 위하여 다양한 방법들 중 하나가 사용될 수 있다. 일반적으로, "프라이머리" 입력은 임의의 방법에 의하여 다른 것들 중에서도 선택된 하나의 입력 포인트이다. 예를 들어, 프라이머리 입력은 세션 내에서 검출된 첫번째 입력, 또는 마지막 입력일 수 있다. 이 방법은 동시에 도착하는 다수의 입력들의 경우 임의의 선택을 강제한다는 단점을 가진다. 해결책은 프라이머리 입력이 지리적인 정렬 공식(sorting formula)(상호작용 의존적일 수 있는)에 따라 가장 높은 순위의 입력과 같은 지리적인 순서의 형태에 의해 선택되도록 하는 것이다. 예를 들어, 정렬 공식은 모든 입력들의 지리적인 중심에서의 원점 및 기준 포인트에 대해 각 입력 포인트에 의해 형성된 각도들을 정렬할 수 있다. 기준 포인트는 예를 들어 사용자가 오른손잡이 또는 왼손잡이인 것에 기초하여 측정되는 각도를 가진 수직선일 수 있다.

[0036]

방법과는 상관없이, 센서 로케이터 결정은 입력들의 도착 및 출발 시간에 의해 영향을 받을 수 있다. 사용자가 복수의 입력들을 동시에 도착시키거나 출발시키려고 의도하지만 약간 서로 다른 때에 복수의 입력을 도착시키거나 출발시키는 상황을 방지하기 위하여, 센서 로케이터 계산을 지연시키기 위한 작은 시간 윈도우(예컨대, 10-60 ms)가 사용될 수 있다.

[0037]

그 다음, 센서 로케이터 위치는 장치 좌표들로부터 디스플레이 좌표들로 매핑된다(205). 그 결과는 프레임을 위한 새로운 출력 로케이터 위치이다. 이 위치는  $[L_s / \text{extent}(R_s) * \text{extent}(R_v)] + R_{V0}$ 에 의해 계산될 수 있으며, 여기서  $L_s$ 는 센서 로케이터의 x좌표 또는 y좌표이고,  $\text{extent}(R_s)$ 는 센서 좌표 공간의 폭 또는 높이이며,  $\text{extent}(R_v)$ 는 뷰포트의 폭 또는 높이이며,  $R_{V0}$ 는 초기 뷰포트의 폭 또는 높이이다. 이 새로운 출력 로케이터는 디스플레이의 경계들 내에 있도록 제약된다.

[0038]

새로운 출력 로케이터가 주어지면, 뷰포트는 뷰포트 로케이터를 획득(206)함으로써 디스플레이 좌표 공간 내에 위치된다. 세션의 첫번째 프레임에 있어서, 뷰포트 위치가 결정되고, 후속적인 프레임들 내에서 그것은 메모리로부터 획득된다. 뷰포트의 위치는 논리적으로 결정되며, 이것은 뷰포트를 디스플레이할 지 여부가 선택적임을 의미한다. 실제로, 대부분의 구현예들에서 실제로는 뷰포트를 디스플레이하지 않는 것이 바람직할 수 있다.

[0039]

위에서 언급된 바와 같이, 뷰포트는 디스플레이 상으로 입력 센서 좌표 공간의 프로젝션이며, 뷰포트 로케이터

위치는 디스플레이 좌표들 내에서 뷰포트의 지리적 중심이다. 또한 위에서 언급된 바와 같이, 출력 로케이터와는 달리, 뷰포트는 입력들이 센서에 의해 검출되지 않은 때에 정의되지 않는다. 그것은 하나의 특정한 장치 인스턴스와 연관되며(사용자 세션 전체와 연관되는 것이 아니라), 그것의 위치는 사용자가 센서 상에 입력들을 초기에 배치할 때 갱신된다. 하나의 입력 세션이 시작한 이후, 그리고 그 입력 세션이 종료될 때까지, 뷰포트는 프레임들 사이에 안정적으로 유지된다. 만약 하나의 프레임이 하나의 입력 세션의 연속(이전 및 현재 프레임 둘 모두로부터의 입력 포인트들의 리스트들은 비어 있지 않음)을 나타낸다면, 뷰포트 로케이터는 메모리로부터 획득된다. 만약 프레임이 하나의 새로운 입력 세션을 초기화한다면, 뷰포트 로케이터는 센서 로케이터(205 단계에서 결정된)와 출력 로케이터 위치(201 단계에서 결정된) 사이의 오프셋을 아래와 같이 결정함으로써 획득된다.

$$\Delta L_D = L_D - L_{D0}$$

$$L_V = [L_S / \text{extent}(R_S) * \text{extent}(R_V)] + L_{V0} + \Delta L_D$$

그런 다음  $L_V$ 는 타겟 디스플레이의 경계들로 구속되며, 위에서 결정된 뷰포트의 범위는 새로운 뷰포트 로케이터를 사용하여 재계산된다.

하나의 프레임에 대해 센서 로케이터, 뷰포트 로케이터, 및 출력 로케이터를 계산한 이후, 해당 프레임을 위한 센서 입력들은 아래에서 더 상세히 기술되는 방식으로 디스플레이 좌표들로 매핑된다(208). 만약 입력 세션이 210에서 결정되는 바와 같이 종료되면, 입력 세션에 관한 몇몇 정보(마지막 출력 로케이터와 같은)는 저장될 수 있다(212). 만약 입력 세션이 종료되지 않았고, 또 만약 갱신된 센서 입력 위치들이 수신되면(214에서 결정되는 바와 같이), 프로세스는 프레임에 대해 센서 로케이터를 결정하는 단계(204)로부터 이들 새로운 센서 입력들을 디스플레이로 매핑하는 단계(208)까지를 반복된다. 그렇지만, 프레임이 연속 세션의 일부라면, 뷰포트 로케이터는 단계(206)에서 결정되지 않고, 메모리로부터 획득된다.

도 3은, 뷰포트 크기 및 위치가 주어지면, 하나의 단일 디스플레이 상에서 바운딩 조건들의 적용(만약 상대 매핑이 수행된다면)을 포함하여, 센서 입력들이 뷰포트 내의 포인트들로 어떻게 매핑되는지를 기술한다. 도 3은 모든 입력들이 디스플레이 내에 존재하도록 구속된 경우를 기술한다.

시스템은 장치로부터, 각각 장치 좌표 공간 내의 좌표들을 가지는, 입력 포인트들의 리스트를 수신한다(300). 그 다음에, 입력 포인트들은 각자의 디스플레이 좌표 공간 내의 대응하는 포인트들로 매핑된다(302). 예를 들어, 장치 좌표 공간 내 포인트( $C_s$ )의 디스플레이 좌표 공간 내 좌표들( $C_D$ )은  $[C_s / \text{extent}(R_s) * \text{extent}(R_v)] + R_v$ 에 의해 계산될 수 있다.

입력 포인트들을 포함하는 바운딩 박스(bounding box)가 정의된다(304). 바운딩 박스의 코너들은 디스플레이의 비주얼 영역으로 매핑되고 비교된다(306). 만약 바운딩 박스의 코너들 중 어느 것도 디스플레이의 비주얼 영역 외부에 있지 아니하면, 입력 매핑은 유지된다(310). 그렇지 않으면, 바운딩 박스가 디스플레이의 비주얼 영역 내에 있도록 이동시키기 위한 오프셋이 결정된다(312). 최소 정정 오프셋을 계산하는데 있어서, 입력 바운딩 박스의 각 비-일치 코너 또는 개별 입력의 이전 프레임과 현재 프레임 사이의 변위 벡터는 하나의 경로 및 볼 수 있는 디스플레이 경계와 경로의 교차 포인트를 정의한다. 정정 오프셋은 경로의 원점과 교차 포인트 사이의 변위이다. 이 오프셋은 디스플레이의 비주얼 영역 내에서 새로운 위치들로 그들을 재-매핑하기 위해 포인트들에게 적용된다(314).

다른 하나의 구현예에서, 포인트들은 장치로부터의 적어도 하나의 입력 포인트의 디스플레이가 유지되도록 구속된다. 도 4에서, 시스템은 장치로부터, 각각 장치 좌표 공간 내의 좌표들을 가지는, 입력 포인트들의 리스트를 수신한다(400). 그 다음에, 입력 포인트들은 각자의 디스플레이 좌표 공간 내의 대응하는 포인트들로 매핑된다(402). 입력 포인트들을 포함하는 바운딩 박스가 정의된다(404). 바운딩 박스의 코너들은 디스플레이의 비주얼 영역에 비교된다(406). 만약 바운딩 박스의 코너들 중 적어도 하나가 디스플레이의 비주얼 영역 내에 있으면, 입력 매핑은 유지된다(410). 그렇지 않으면, 바운딩 박스의 적어도 하나의 포인트가 디스플레이의 비주얼 영역 내에 있도록 이동시키기 위한 정정 오프셋이 결정된다(412). 그 다음, 구속된 코너와 가장 가까운 입력의 오프셋이 결정되고 정정 오프셋으로 적용된다(414). 이 갱신된 정정 오프셋은 디스플레이의 비주얼 영역 내에서 새로운 위치들로 그들을 재-매핑하기 위해 포인트들에게 적용된다(416).

다수 모니터 디스플레이들에 있어서, 프로세스는 유사하다. 디스플레이의 볼 수 있는 영역들의 결합이 내부적인 빈 공간없이 하나의 단일하고, 사각형이며, "가상의" 디스플레이인 정규 디스플레이 토폴로지들이 존재한다. 정규 디스플레이 토폴로지들에 있어서, 가상 디스플레이 표면의 경계들에 대하여 다수 입력들의 경계는 하나의 단일 디스플레이에 대한 것과 동일하다. 또한 비정규 디스플레이 토폴로지들도 존재할 수 있는데, 여기서 디스플



레이의 볼 수 있는 영역들의 결합은 볼록한 또는 오목한 내부적인 빈 공간을 가진 직선으로 둘러싸인 가상 디스플레이이다. 이러한 디스플레이 토폴로지들에 있어서, 상술한 방법들은 정정 오프셋들을 계산하고 적용하기 위해 사용될 수 있다.

[0048] 그렇지만, 추가 실패 경우는 하나의 포인트가 볼록 또는 오목 내부 빈 공간 중 하나에 놓일 때, 디스플레이의 비주얼 영역의 외부에서만 이들 포인트들을 포함하는 바운딩 박스가 계산되고, 정정 오프셋을 계산하기 위하여 사용될 수 있는 경우이다. 이런 경우, 바운딩 박스는 디스플레이의 비주얼 영역들로 매핑하지 않는 입력 포인트들을 포함하도록 계산되며, 여기서 비-일치(non-conforming) 바운딩 박스라고 불리운다. 비-일치 바운딩 박스의 적어도 하나의 코너가 디스플레이의 비주얼 부분 내에 포함하는 것을 보장하는 최소 정정 오프셋이 계산된다. 이 정정 오프셋은 모든 입력들을 위한 장치-디스플레이 변환에 적용된다.

[0049] 다수 모니터들을 위한 바운딩 조건들의 더 특정한 예시적인 구현예가 이제 기술될 것이다.

[0050] 이 예에서, 각각의 입력에 대해, 하나의 타겟 바운딩 디스플레이( $R_{D, \text{타겟}}$ )이 아래의 방식으로 결정된다. 첫째, 입력 위치( $C_D$ )는 가상 디스플레이 표면의 볼 수 있는 영역 내에 포함되지 않는지 여부가 결정된다. 만약 아니라면, 이전 프레임을 위한 입력의 디스플레이( $R_{D0}$ )의 좌표들이 획득된다. 새로운 세션을 나타내는 프레임에 있어서, 이들 좌표들은 출력 로케이터 위치( $L_D$ )를 포함하는 디스플레이의 좌표들과 대체된다. 그 다음에, 입력( $C_D$ )이 x축 또는 y축 중 어느 하나에서  $R_{D0}$ 에 의해 계속 바운딩되어 있는지 여부가 결정된다. 만약 둘 중 어느 하나의 축에서 긍정적인 테스트가 관찰되면, 타겟 바운딩 디스플레이는 디스플레이( $R_{D0}$ )이다. 그렇지 않은 경우, 입력은 디스플레이( $R_{D0}$ )의 경계들 외부에 있다. 그런 다음 센서 좌표들 내의 변위 벡터  $\Delta S_s$ 가 이 입력에 대해 결정된다:  $\Delta S_s = C_s - C_{s0}$ . 센서의 범위, extent( $R_s$ )가 획득된다. 변위의 주축(dominant axis)이 결정된다. 만약  $|\Delta S_{sx} / \text{extent}(R_{sx})| \geq |\Delta S_{sy} / \text{extent}(R_{sy})|$ 이면, X축이 주축이다. 그렇지 않은 경우에는 Y축이 주축이다.

[0051] 그런 다음 입력 변위의 주축은 타겟 바운딩 디스플레이를 결정하기 위하여 사용된다. 만약 X축이 주축이라면, 타겟 바운딩 디스플레이( $R_{D, \text{타겟}}$ )은 다음의 조건들을 만족하는 디스플레이이다: 1. 입력이 디스플레이의 수평 범위 내에 떨어진다; 2. 타겟 디스플레이는 입력의 일차 이동 방향 쪽에 있고, 마지막 디스플레이와 해당 경계를 공유한다; 및 3. 마지막 입력 위치는 디스플레이의 수직 범위 내로 떨어진다. 만약 Y축이 주축이라면, 타겟 바운딩 디스플레이( $R_{D, \text{타겟}}$ )은 다음의 조건들을 만족하는 디스플레이이다: 1. 입력이 디스플레이의 수직 범위 내에 떨어진다; 2. 타겟 디스플레이는 입력의 일차 이동 방향 쪽에 있고, 마지막 디스플레이와 해당 경계를 공유한다; 및 3. 마지막 입력 위치는 디스플레이의 수평 범위 내로 떨어진다.

[0052] 만약 타겟 바운딩 디스플레이가 주축 방향을 사용하여 결정될 수 없다면, 검색이 비-주축 방향에서 수행된다. 만약 타겟 바운딩 디스플레이가 여전히 발견되지 않는다면, 타겟 디스플레이는 입력의 이전 디스플레이이다.

[0053] 입력을 위한 타겟 바운딩 디스플레이가 주어지면, 입력은 해당 디스플레이에 클램프(clamp)되고, 클램핑 오프셋이 계산되어 저장된다. 이 클램핑 오프셋은 모든 입력들에게 적용됨으로써 그들 사이의 상대적인 거리가 유지되도록 한다. 이런 방식으로 입력들을 조정한 후, 그들은 모두 다시 테스트되어 그들이 디스플레이의 볼 수 있는 부분 상에 있다는 것을 보장한다.

[0054] 몇몇 상호작용 모드들에서, 센서를 이용하여 다수의 입력들을 동시에 만들려고 하는 사용자의 의도를 실현하기 위한 작은 양의 시간이 허용된다. 세션의 첫번째 입력이 관찰된 때, 타이머가 활성화되고 도착하는 입력들은 비활성으로 표시되고, 센서 로케이터 결정은 타이머가 종료될 때까지 연기되거나 또는 도착하는 입력들이 제거되면 끝난다. 유사하게, 사용자는 입력들이 동시에 출발할 것을 의도할 수 있다. 센서 로케이터 위치에 영향을 미치지 않으면서 이런 의도를 실현하기 위하여, 타이머가 사용될 수 있다. 타이머가 활성화되고, 출발하는 입력들은 타이머가 종료될 때까지 센서 로케이터 계산에 계속해서 포함된다.

[0055] 상술한 기술에서, 상대 매핑 모드 및 절대 매핑 모드 둘 모두에 있어서, 입력 포인트들은 직접 디스플레이 좌표들로 매핑된다. 그렇지만, 상대 매핑 모드에서, 입력 장치는 타겟 디스플레이 좌표 공간의 서브세트만을 스캔할 수 있다. 그러므로, 한 디스플레이 위치에서 다른 한 위치로의 네비게이션은, 입력 포인트들의 움직임이 검출될 때 포인트들의 어떤 가속의 형태가 적용되지 않는 한, 다수의 스트로크를 포함할 수 있다. 반대로, 픽셀-레벨 포인트-대-포인트 타겟팅 정밀도를 달성하기 위하여, 포인트들의 감속의 형태가 적용될 수 있다. 이러한 가속 및 감속은, 종종 "포인트 볼리스틱스"라고 지칭되며, 아래의 방식으로 다수 입력, 간접 입력 장치에 적용될 수 있다. 입력 장치 상의 입력 포인트들의 변위는, 이 경우에 있을 수 있는, 디스플레이 상의 포인트들의 움직임을

가속 또는 감속하기 위하여, 장치 좌표 공간으로부터 디스플레이 좌표공간으로의 입력 포인트들의 매핑 내에서 고려된다. 일반적으로, 입력 포인트들을 위한 변위의 측정값이 결정된다. 이 변위는 입력 장치 포인트들의 각각의 대응하는 디스플레이 좌표들로의 매핑을 어떻게 변화시킬 것인가를, 변위에 기초하여, 결정하는 함수에 대한 입력이다.

[0056] 일 구현예에서, 각 입력 포인트의 변위가 결정된다. 가장 낮은 좌표 변위 벡터를 가진 입력의 센서 픽셀들 내의 물리적인 변위는, 하나의 단일한 가속된 디스플레이 변위를 생성하기 위한 가속 곡선 변형을 통해 전달되고, 이것은 출력 로케이터 및 모든 포인트들의 디스플레이 변위들로 적용된다. 가속 함수로의 입력들은 벡터 크기일 수 있거나 또는 각각의 축을 위한 하나의 값이 두 개의 서로 다른 가속 함수들로 입력될 수 있다. 이 구현예는 이제 도 5와 관련하여 기술될 것이다.

[0057] 먼저, 입력 센서 상에서 입력 포인트들이, 시간적으로 첫번째 및 두번째 포인트들로부터, 수신된다(500). 이동하거나 또는 정지된 입력들을 어떻게 고유하게 식별하며 추적할 수 있는가는, 해당 기술분야에서 "입력 인식 및 추적"으로서 알려져 있으며, 장치 및 센서 특징이라는 것에 주목하라. 본 발명은 임의의 특정 입력 인식 및 추적 기법으로 제한되지 않는다. 해당 기술 분야에서 적합할 수 있는 것으로 발견된 그러한 인식 및 추적을 위한 임의의 기법이 사용될 수 있다.

[0058] 그런 다음, 하나의 시간 인터벌 내에서 각 입력의 각 차원 내에서 장치 좌표들(즉, 픽셀들) 내 변위가 결정된다(502). 만약 시간 인터벌들이 일정하다고 알려져 있다면, 변위 홀로 사용될 수 있다. 그렇지 않다면, 시간 인터벌은 속도를 계산하는데 사용될 수 있다.

[0059] 입력들의 각각의 시간 인터벌, 또는 "프레임"에 있어서, 가장 작은 크기의 변위, 또는 속도를 가진 입력이 식별된다(504). 입력 센서 상에서 정지되어 있는 입력들이 디스플레이로 매핑된 때에 정지된 채로 유지되도록 하기 위하여 가장 작은 크기(예를 들어, 평균 또는 최대값이 아니라)를 가진 입력이 선택된다.

[0060] 식별된 입력의 변위는 픽셀 형태의 변위에서, 센서의 픽셀 밀도를 사용하여, 물리적인 변위로 변환될 수 있다. 변위 값은 가속된 변위로 그 값을 변형시키는 가속 함수의 입력으로서 사용된다(506). 본 발명은 사용되는 특정한 가속 공식에 의해 제한되지 않는다. 마우스 포인터 가속을 위해 사용되는 것과 같이, 현재 해당 기술 분야에서 사용되는 임의의 합리적인 기법이 사용될 수 있다. 본 발명은 일반적으로 각 좌표 축(x, y 또는 z)의 독립적인 가속을 허용하는 임의의 가속 공식에 적용될 수 있다. 적절한 변형은 변위 값을 가속된 변위 값으로 매핑하는 피스-와이즈(piece-wise) 선형 함수를 사용하여 구현될 수 있다. 가속된 변위 값은, 만약 물리적인 차원들에 기초하고 있다면, 픽셀 좌표들로 다시 변환될 수 있다.

[0061] 그런 다음, 가속된 변위는 디스플레이 좌표 공간 내의 가속된 변위로 변환된다(508). 예를 들어, 변환은 다음:  $\Delta C_D = \Delta C_S / \text{extent}(R_S) * \text{extent}(R_V)] + R_V$  와같이 표현될 수 있다. 그런 다음, 디스플레이 좌표들로 매핑된 각각의 입력 위치는 가속된 변위에 의해 조정된다(510).

[0062] 절대적으로 매핑된 차원들에 있어서, 도 6과 관련해서 기술되는 바와 같이, 스캔 조정(span adjustment)이라고 불리는 유사한 프로세스가 사용될 수 있다. 도 6에서, 센서 로케이터로부터의 각 입력의 변위가 장치 좌표 공간 내의 픽셀들의 형태로 결정된다(600). 최소 변위가 선택된다(602). 이 최소 변위 값은 장치의 픽셀 밀도를 사용하여 물리적인 차원들로 변환된다(604). 물리적인 차원들 형태의 최소 변위 값은 임의의 적절한 변형을 사용하여 스캔 조정 값으로 변형된다(606). 적절한 변형은 변위 값을 스캔 조정 값으로 매핑하는 피스-와이즈 선형 함수와 같은 가속 변형과 유사할 수 있다. 이 스캔 조정 값은 다시 픽셀 값들로 변환된다(608). 그런 다음, 가속과 유사하게, 스캔 조정 값은 디스플레이 픽셀 값들로 변형되고(610), 각각의 입력 포인트는 그 값을 사용하여 조정된다(612).

[0063] 입력 포인트들에 대한 가속 및 스캔 조정 수정은, 포인트들이 볼 수 있는 디스플레이 영역 내에 유지되는 것을 보장하는 경계 조건들을 적용하기 이전에 이루어져야 한다는 점이 주목되어야 한다.

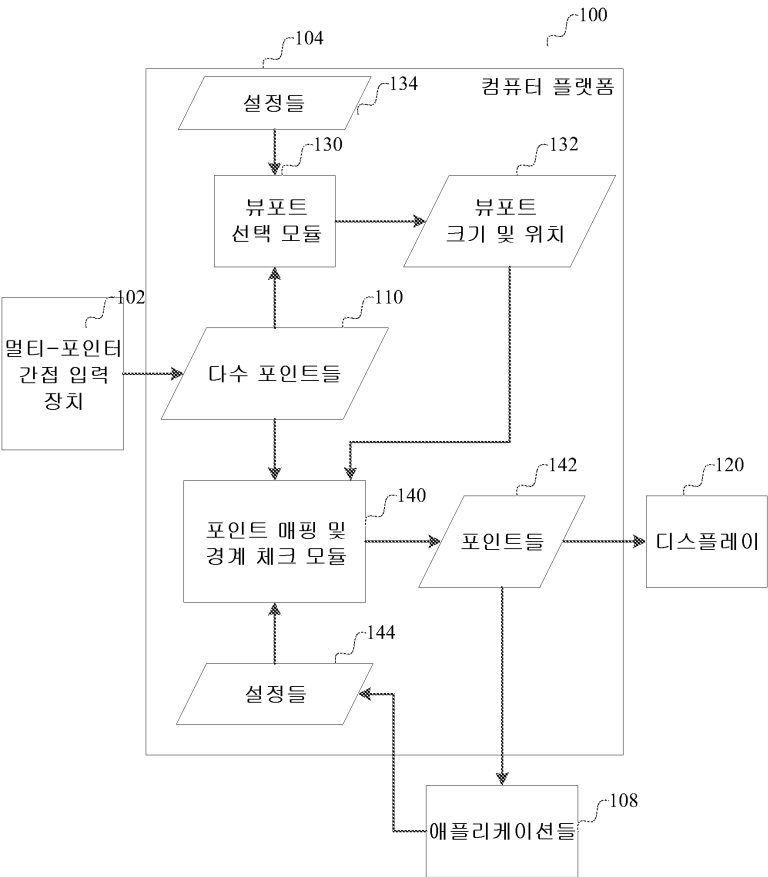
[0064] 이제 예시적인 구현예가 기술되었고, 그러한 시스템이 동작하도록 설계되는 컴퓨팅 환경이 이제 기술될 것이다. 아래의 기술은 이 시스템이 구현될 수 있는 적절한 컴퓨터 환경에 대한 간략하고 일반적인 기술을 제공하려는 것이다. 시스템은 다양한 범용 또는 특정 목적의 컴퓨팅 하드웨어 구성들을 통해 구현될 수 있다. 잘 알려진 컴퓨팅 장치의 예들에는 개인용 컴퓨터, 서버 컴퓨터, 핸드-헬드 또는 랩탑 장치들(예들 들어, 미디어 플레이어, 노트북 컴퓨터, 셀룰러 폰, 퍼스널 데이터 어시스턴트, 음성 레코더), 멀티프로세서 시스템, 마이크로프로세서-기반 시스템, 셋탑 박스, 게임 콘솔, 프로그래머블 소비자 가전, 네트워크 PC, 미니컴퓨터, 메인프레임 컴퓨터, 위의 시스템들이나 장치들 중 임의의 것을 포함하는 분산 컴퓨팅 환경 및 기타 등등이 포함되지만 이들로 제한

되지 않는다.

- [0065] 도 7은 적절한 컴퓨팅 시스템 환경의 일 예를 도시한다. 컴퓨팅 시스템 환경은 적절한 컴퓨팅 환경의 단지 하나의 예일 뿐이며 이러한 컴퓨팅 환경의 이용 범위나 기능성에 대한 어떠한 제한을 암시하려고 의도되지 않는다. 또한 컴퓨팅 환경이 예시적인 동작 환경에서 도시하고 있는 임의의 한 컴포넌트 또는 컴포넌트들의 조합과 관련되어 어떠한 종속성이나 필요성을 가지는 것으로 해석되어서도 안된다.
- [0066] 도 7을 참조하면, 예시적인 컴퓨팅 환경은 컴퓨팅 머신(700)과 같은 컴퓨팅 머신을 포함한다. 가장 기본적인 구성에서, 컴퓨팅 머신(700)은 통상적으로 적어도 하나의 프로세싱 유닛(702) 및 메모리(704)를 포함한다. 컴퓨팅 장치는 다수의 프로세싱 유닛들 및/또는 그래픽 프로세싱 유닛(720)과 같은 추가적인 코-프로세싱 유닛들을 포함할 수 있다. 컴퓨팅 장치의 정확한 구성 및 타입에 종속하여, 메모리(704)는 휘발성(RAM과 같은), 비-휘발성(ROM, 플래시 메모리, 등과 같은) 또는 이 둘의 몇몇 조합일 수 있다. 이 가장 기본적인 구성은 도 7에 점선(706)으로 도시되어 있다. 추가적으로 컴퓨팅 머신(700)은 또한 추가적인 특징들/기능성을 가질 수 있다. 예를 들어, 컴퓨팅 머신(700)은 또한 자기 또는 광 디스크 또는 테이프를 포함하지만, 이에만 제한되는 것이 아닌, 추가적인 스토리지(탈거가능 및/또는 비-탈거가능)를 포함할 수 있다. 그러한 추가적인 스토리지는 도 7에 탈거가능 스토리지(708) 및 비-탈거가능 스토리지(710)에 의해 도시된다. 컴퓨터 스토리지 매체는 컴퓨터 프로그램 인스트럭션, 데이터 구조, 프로그램 모듈 또는 그 외 다른 데이터와 같은 정보의 스토리지를 위한 임의의 방법이나 기술에 의해 구현되는 휘발성 및 비휘발성, 탈거가능 및 비-탈거가능 매체를 포함한다. 메모리(704), 탈거가능 스토리지(708) 및 비-탈거가능 스토리지(710)는 모두 컴퓨터 스토리지 매체의 예들이다. 컴퓨터 스토리지 매체에는 RAM, ROM, EEPROM, 플래시 메모리 또는 그 외 다른 메모리 기술, CD-ROM, 디지털 다기능 디스크(DVD) 또는 그 외 다른 광 스토리지, 자기 카세트, 자기 테이프, 자기 디스크 스토리지 또는 그 외 다른 자기 스토리지 장치, 또는 원하는 정보를 저장하기 위해 사용될 수 있고 컴퓨팅 머신(700)에 의해 액세스될 수 있는 임의의 그 외 다른 매체가 포함되지만, 이것들로 제한되지 않는다. 임의의 그러한 컴퓨터 스토리지 매체는 컴퓨팅 머신(700)의 일부일 수 있다.
- [0067] 컴퓨팅 머신(700)은 또한 장치가 그 외 다른 장치들과 통신하는 것을 허용하는 통신 커넥션(들)(712)을 포함할 수 있다. 통신 커넥션(들)(712)은 통신 매체의 일 예이다. 통신 매체는 통상적으로 반송파 또는 그 외 다른 전송 메커니즘과 같이 변조된 데이터 신호의 형태로 컴퓨터 프로그램 인스트럭션, 데이터 구조, 프로그램 모듈 또는 그 외 다른 데이터를 운반하며, 임의의 정보 전달 매체를 포함한다. "변조된 데이터 신호"는 하나 이상의 특성 세트를 가지거나 정보를 신호 내에 인코딩하는 방식으로 변화된 신호를 의미하며, 그에 의해 신호의 수신 장치의 구성이나 상태를 변화시킨다. 제한이 아니라 예시로서 말하면, 통신 매체는 유선 네트워크 또는 직접-유선 커넥션과 같은 유선 매체, 및 음향, RF, 적외선 및 그 외 다른 무선 매체와 같은 무선 매체를 포함한다.
- [0068] 컴퓨팅 머신(700)은 키보드, 마우스, 펜, 카메라, 터치 입력 장치 및 기타 등등과 같은 다양한 입력 장치(들)(714)를 구비할 수 있다. 디스플레이, 스피커, 프린터, 및 기타 등등과 같은 출력 장치(들)(716)도 또한 포함될 수 있다. 이들 장치들 모두는 해당 기술 분야에서 잘 있으므로 여기서 길게 논의할 필요는 없다.
- [0069] 시스템은 컴퓨터 머신에 의해 프로세스되는, 프로그램 모듈과 같은 컴퓨터-실행가능 인스트럭션 및/또는 컴퓨터-해석 인스트럭션을 포함하여, 소프트웨어의 일반적인 맥락으로 구현될 수 있다. 일반적으로, 프로그램 모듈은 프로세싱 유닛에 의해 프로세싱되면 프로세싱 유닛으로 하여금 특정 태스크를 수행하거나 특정 추상적인 데이터 타입을 구현하도록 지시하는 루틴, 프로그램, 오브젝트, 컴포넌트, 데이터 구조, 및 기타 등등을 포함한다. 이 시스템은 통신 네트워크를 통해 연결되어 있는 원격 프로세싱 장치들에 의해 태스크가 수행되는 분산 컴퓨팅 환경에서 실시될 수 있다. 분산 컴퓨팅 환경에서, 프로그램 모듈은 메모리 스토리지 장치들을 포함하여 로컬 및 원격 컴퓨터 스토리지 매체 둘 모두에 위치될 수 있다.
- [0070] 첨부된 청구범위의 전제부의 용어 "제조품(article of manufacture)", "프로세스", "머신" 및 "물질의 조합"은 35 U.S.C. § 101에서 이들 용어의 사용에 의해 정의되는 특허가능한 발명 주제의 범위 내에 포함되는 발명 주제로 청구범위를 제한하는 것으로 의도된다.
- [0071] 본 명세서에서 기술된 앞서 언급된 다른 실시예들 중 임의의 것 또는 모든 것은 추가적인 혼합 실시예들을 형성하기 위하여 원하는 임의의 조합으로 사용될 수 있다. 첨부된 청구범위에서 한정된 발명 주제는 위에서 기술된 특정 구현예들로 제한되어야 할 필요가 없다는 점이 이해되어야 한다. 위에서 기술된 특정 구현예들은 오직 예시로서만 개시된다.

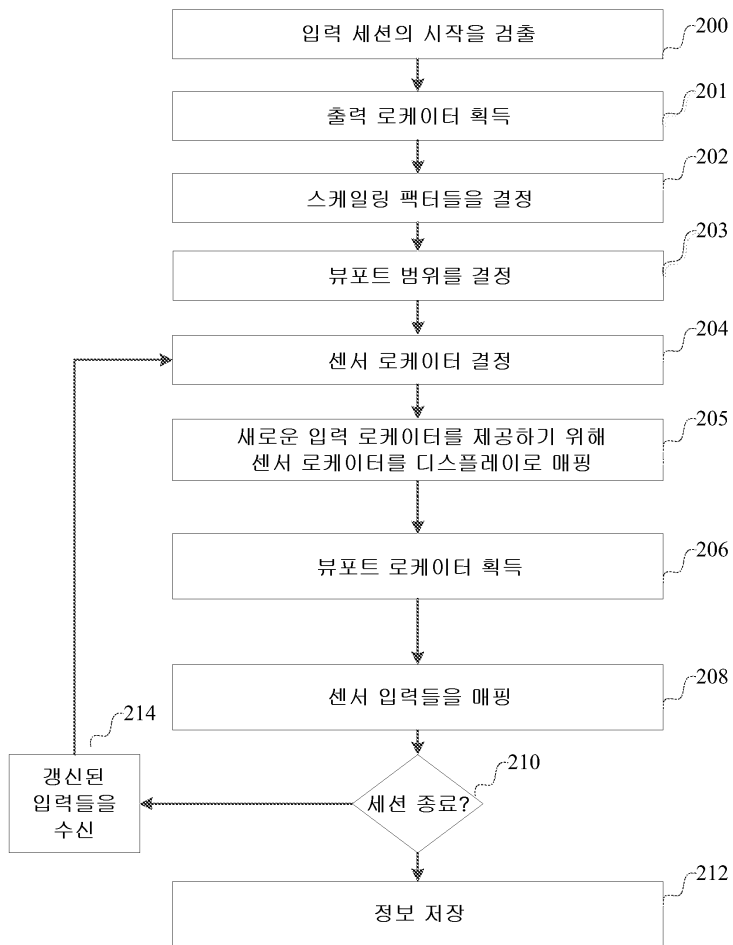
도면

도면1

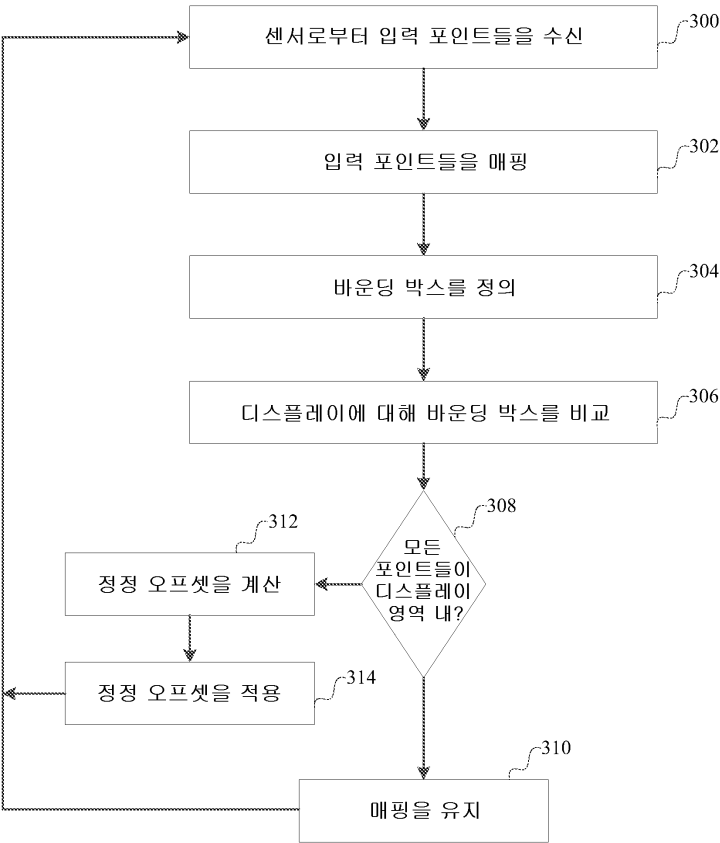




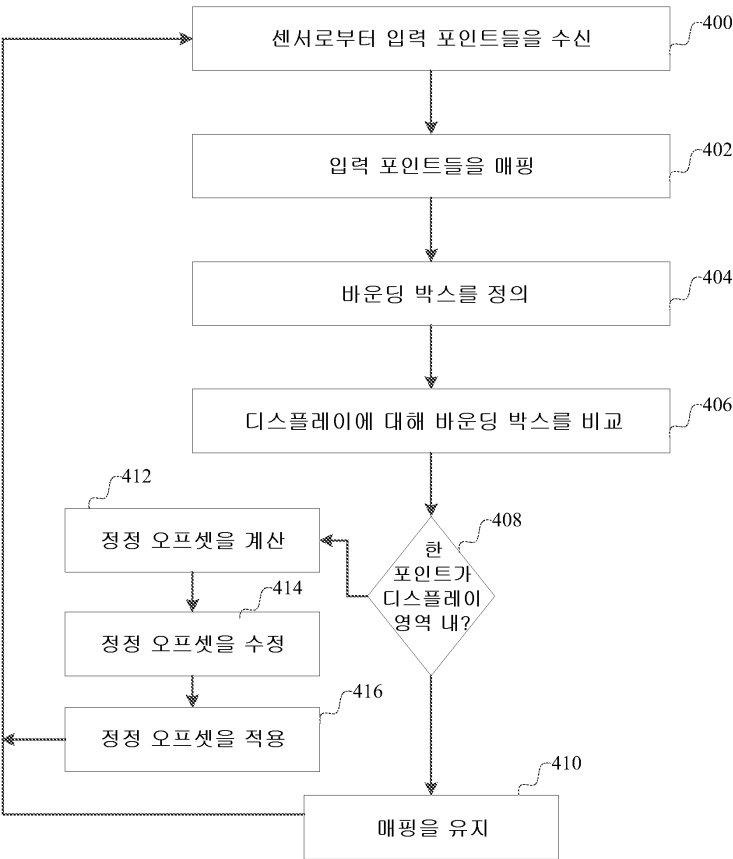
도면2



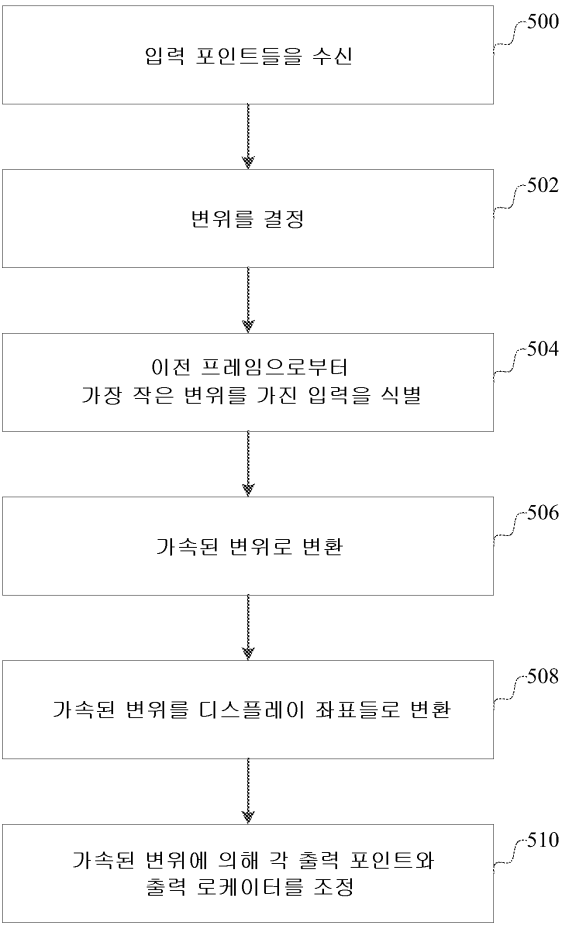
도면3



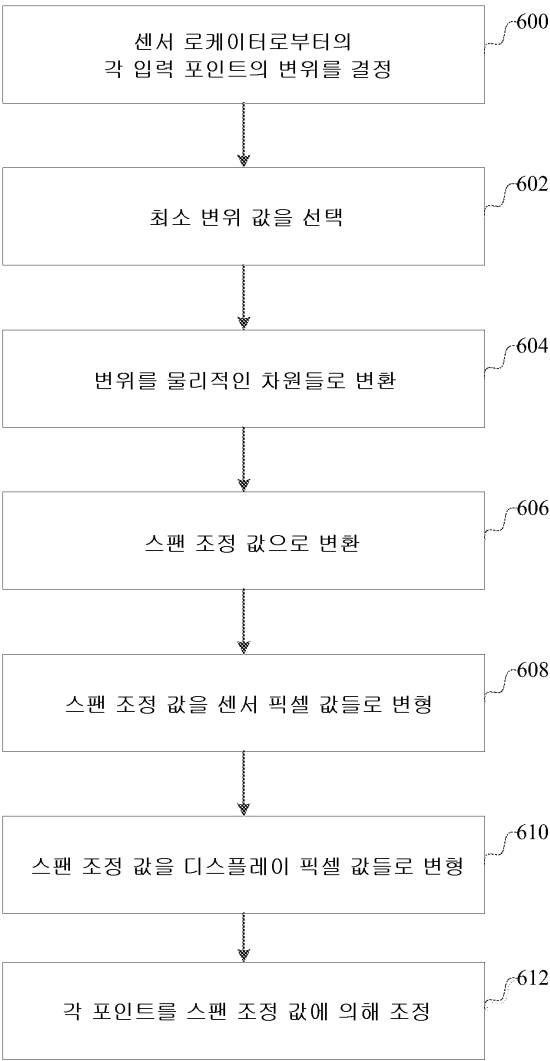
도면4



도면5



도면6



도면7

