



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0088982  
(43) 공개일자 2023년06월20일

- |  |  |
|--|--|
| <p>(51) 국제특허분류(Int. Cl.)<br/>G06F 3/0488 (2022.01) G06F 3/0481 (2022.01)<br/>G06F 3/0484 (2022.01)</p> <p>(52) CPC특허분류<br/>G06F 3/04883 (2022.01)<br/>G06F 3/0481 (2013.01)</p> <p>(21) 출원번호 10-2021-0177277</p> <p>(22) 출원일자 2021년12월13일<br/>심사청구일자 2021년12월13일</p> | <p>(71) 출원인<br/>연세대학교 산학협력단<br/>서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)</p> <p>(72) 발명자<br/>이병주<br/>서울특별시 서대문구 연희로24길 16<br/>연공아트빌라, 301호</p> <p>김현철<br/>대전광역시 유성구 어은로 57 한빛아파트 137동 204호</p> <p>(74) 대리인<br/>정부연</p> |
|--|--|

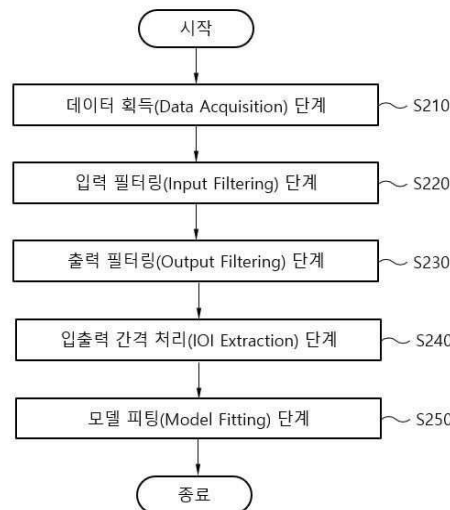
전체 청구항 수 : 총 16 항

(54) 발명의 명칭 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치 및 방법

(57) 요약

본 발명은 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치 및 방법에 관한 것으로, 상기 장치는 사용자의 입력 및 화면의 출력을 기록하는 데이터 획득부; 상기 사용자의 입력에 대한 입력 타이밍을 추출하는 입력 필터링부; 상기 화면의 출력에 대한 특정 기준 이상의 시각적 변경을 개시하는 출력 타이밍을 추출하는 출력 필터링부; 상기 사용자의 입력 및 상기 화면의 출력에 대한 타이밍들을 통해 입출력 간격의 분포를 생성하는 입출력 간격 처리부; 및 상기 입출력 간격의 분포를 선제적 분포 및 반응적 분포로 조절하여 선제성 및 반응성을 추정하는 모델 피팅부를 포함한다.

대표도 - 도2



(52) CPC특허분류

**G06F 3/0484** (2022.01)

G09G 2340/04 (2013.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711126082
과제번호	2020-0-01361-002
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신방송혁신인재양성(R&D)
연구과제명	인공지능대학원지원(연세대학교)
기 여 율	1/3
과제수행기관명	연세대학교 산학협력단
연구기간	2021.01.01 ~ 2021.12.31

이 발명을 지원한 국가연구개발사업

과제고유번호	1375027292
과제번호	R2021040105-0001
부처명	문화체육관광부
과제관리(전문)기관명	한국콘텐츠진흥원
연구사업명	문화기술연구개발(R&D)
연구과제명	Game Now : e-스포츠 서비스를 위한 인공지능 기반 실시간 게임 분석 기술 개발
기 여 율	1/3
과제수행기관명	한국전자통신연구원
연구기간	2021.06.01 ~ 2021.12.31

이 발명을 지원한 국가연구개발사업

과제고유번호	1375027277
과제번호	R2019020010-0003
부처명	문화체육관광부
과제관리(전문)기관명	한국콘텐츠진흥원
연구사업명	문화기술연구개발(R&D)
연구과제명	시니어 게임 플레이 지원 기술 및 게임 서비스 모델 개발
기 여 율	1/3
과제수행기관명	연세대학교 산학협력단
연구기간	2021.01.01 ~ 2021.12.31

---

## 명세서

### 청구범위

#### 청구항 1

사용자의 입력 및 화면의 출력을 기록하는 데이터 획득부;

상기 사용자의 입력에 대한 입력 타이밍을 추출하는 입력 필터링부;

상기 화면의 출력에 대한 특정 기준 이상의 시각적 변화를 개시하는 출력 타이밍을 추출하는 출력 필터링부;

상기 사용자의 입력 및 상기 화면의 출력에 대한 타이밍들을 통해 입출력 간격의 분포를 생성하는 입출력 간격 처리부; 및

상기 입출력 간격의 분포를 선제적 분포 및 반응적 분포로 조절하여 선제성 및 반응성을 추정하는 모델 피팅부를 포함하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 2

제1항에 있어서, 상기 데이터 획득부는

특정 구간에 대해 상기 사용자의 시계열 입력 액션들을 입력 엔티티별로 기록하고 상기 화면의 시계열 출력을 프레임 이미지별로 기록하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 3

제1항에 있어서, 상기 입력 필터링부는

상기 사용자의 입력이 입력 이벤트의 생성을 기초로 이산적 타이밍을 추출하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 4

제1항에 있어서, 상기 출력 필터링부는

(1) 상기 사용자의 입력에 의해 촉발된 화면상의 시각적 변화, (2) 상기 사용자의 반응적 입력을 촉발시키는 화면상의 시각적 변화, (3) 상기 사용자의 선제적 입력을 촉발시키는 화면상의 시각적 변화, (4) 상기 사용자의 입력과 전혀 무관한 화면상의 시각적 변화를 통한 상기 화면의 출력을 기초로 상기 출력 타이밍을 추출하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 5

제1항에 있어서, 상기 출력 필터링부는

상기 출력 타이밍의 추출 전에 상기 화면의 출력을 기록하는 프레임 이미지의 해상도 또는 프레임레이트를 다운스케일 하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 6

제5항에 있어서, 상기 출력 필터링부는

상기 사용자의 입력에 대한 평균 간격을 기초로 상기 다운스케일의 인자를 설정하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 7

제6항에 있어서, 상기 출력 필터링부는

상기 화면의 출력을 구성하는 프레임 이미지들 간의 차이에 대한 사용자 시각 인지성을 기초로 특정 기준을 설정하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 8

제1항에 있어서, 상기 입출력 간격 처리부는

상기 사용자의 입력 및 상기 화면의 출력에 대한 타이밍들 간의 차이를 산출하고 상기 차이를 기초로 복수의 막대들로 구성된 분포를 생성하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 9

제8항에 있어서, 상기 입출력 간격 처리부는

상기 분포에서 음의 영역에 존재하는 최빈도 막대를 제거하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 10

제1항에 있어서, 상기 모델 피팅부는

상기 입출력 간격의 분포를 가우시안 분포로 생성하여 상기 선제성을 산출하고 상기 입출력 간격의 분포를 엑스-가우시안 분포를 포함하는 반응적 분포로 생성하여 상기 반응성을 산출하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 11

제10항에 있어서, 상기 모델 피팅부는

상기 선제성 및 상기 반응성을 기초로 사용자 인터페이스의 특성을 조절하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 12

제11항에 있어서, 상기 모델 피팅부는

상기 사용자 인터페이스의 시각적 레이아웃 및 동작 기능을 조절하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치.

#### 청구항 13

사용자의 입력 및 화면의 출력을 기록하는 데이터 획득단계;

상기 사용자의 입력에 대한 입력 타이밍을 추출하는 입력 필터링단계;

상기 화면의 출력에 대한 특정 기준 이상의 시각적 변경을 개시하는 출력 타이밍을 추출하는 출력 필터링단계;

상기 사용자의 입력 및 상기 화면의 출력에 대한 타이밍들을 통해 입출력 간격의 분포를 생성하는 입출력 간격 처리단계; 및

상기 입출력 간격의 분포를 선제적 분포 및 반응적 분포로 조절하여 선제성 및 반응성을 추정하는 모델 피팅단계를 포함하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 방법.

#### 청구항 14

제13항에 있어서, 상기 데이터 획득단계는

특정 구간에 대해 상기 사용자의 시계열 입력 액션들을 입력 엔티티별로 기록하고 상기 화면의 시계열 출력을 프레임 이미지별로 기록하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 방법.

#### 청구항 15

제13항에 있어서, 상기 입력 필터링 단계는

상기 사용자의 입력이 입력 이벤트의 생성을 기초로 이산적 타이밍을 추출하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 방법.

#### 청구항 16

제13항에 있어서, 상기 출력 필터링 단계는

상기 출력 타이밍의 추출 전에 상기 화면의 출력을 기록하는 프레임 이미지의 해상도 또는 프레임레이트를 다운스케일 하는 것을 특징으로 하는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 방법.

### 발명의 설명

#### 기술 분야

[0001] 본 발명은 사용자 인터페이스 기술에 관한 것으로, 보다 상세하게는 컴퓨터 화면 비디오와 사용자의 입력 로그 데이터를 분석하여 사용자의 입력 행동이 얼마나 반응적(reactive)이거나 예측적(proactive)이었는지를 정량화할 수 있는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치 및 방법에 관한 것이다.

#### 배경 기술

[0003] HCI(Human-Computer Interaction)는 인간(사용자)과 컴퓨터 간의 상호작용을 뜻하며, 여기에서 상호작용은 사용자와 컴퓨터 사이에 있는 사용자 인터페이스에서 발현되는 작용을 의미한다.

[0004] 일상적인 상호작용에서 컴퓨터의 수많은 출력이 사용자에게 제공되고 사용자도 여러 버튼 입력을 생성한다. 이 과정에서, 사용자는 컴퓨터의 출력에 반응하거나 또는 컴퓨터의 출력을 예상하여 능동적으로 행동하는 두가지 전략 중 하나를 따른다. 사용자가 반응 또는 사전 버튼 입력 전략에 의존하는 정도를 각각 사용자 반응 및 사용자 사전 대응으로 통계적으로 정량화할 경우에 반응성의 정량화는 상호작용 기술의 효율성을 나타낼 수 있다. 시간 경과에 따른 반응성의 변화는 습관 형성 및 학습을 나타내는 역할을 한다. 마찬가지로 선제성은 일반적인 예측 가능성과 일관성을 나타낸다. 시간 경과에 따른 주도성의 변화는 습관 형성 및 학습을 나타낸다. 이 두 구성에 대한 정량적 분석 방법은 존재하지 않지만 HCI의 많은 질문과 관련이 있다. 예를 들어, 인터랙션 디자

이너는 애니메이션 전환이 실제로 사용자가 향후 입력을 계획하고 실행하는 데 도움이 되는지 궁금해할 수 있다. 또한, 게임 디자이너는 게임의 시각적 디자인이 플레이어의 성능에 어떤 영향을 미치는지 궁금해할 수 있다.

## 선행기술문헌

### 특허문헌

[0006] (특허문헌 0001) 한국공개특허 제10-2021-0002713호 (2021.01.08)

## 발명의 내용

### 해결하려는 과제

[0007] 본 발명의 일 실시예는 컴퓨터 화면 비디오와 사용자의 입력 로그 데이터를 분석하여 사용자의 입력 행동이 얼마나 반응적(reactive)이거나 예측적(proactive)이었는지를 정량화할 수 있는 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치 및 방법을 제공하고자 한다.

### 과제의 해결 수단

[0009] 실시예들 중에서, 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치는 사용자의 입력 및 화면의 출력을 기록하는 데이터 획득부; 상기 사용자의 입력에 대한 입력 타이밍을 추출하는 입력 필터링부; 상기 화면의 출력에 대한 특정 기준 이상의 시각적 변화를 개시하는 출력 타이밍을 추출하는 출력 필터링부; 상기 사용자의 입력 및 상기 화면의 출력에 대한 타이밍들을 통해 입출력 간격의 분포를 생성하는 입출력 간격 처리부; 및 상기 입출력 간격의 분포를 선제적 분포 및 반응적 분포로 조절하여 선제성 및 반응성을 추정하는 모델 피팅부를 포함한다.

[0010] 상기 데이터 획득부는 특정 구간에 대해 상기 사용자의 시계열 입력 액션들을 입력 엔티티별로 기록하고 상기 화면의 시계열 출력을 프레임 이미지별로 기록할 수 있다.

[0011] 상기 입력 필터링부는 상기 사용자의 입력이 입력 이벤트의 생성을 기초로 이산적 타이밍을 추출할 수 있다.

[0012] 상기 출력 필터링부는 (1) 상기 사용자의 입력에 의해 촉발된 화면상의 시각적 변화, (2) 사용자의 반응적 입력을 촉발시키는 화면상의 시각적 변화, (3) 상기 사용자의 선제적 입력을 촉발시키는 화면상의 시각적 변화, (4) 상기 사용자의 입력과 전혀 무관한 화면상의 시각적 변화를 통한 상기 화면의 출력을 기초로 상기 출력 타이밍을 추출할 수 있다.

[0013] 상기 출력 필터링부는 상기 출력 타이밍의 추출 전에 상기 화면의 출력을 기록하는 프레임 이미지의 해상도 또는 프레임레이트를 다운스케일 할 수 있다.

[0014] 상기 출력 필터링부는 상기 사용자의 입력에 대한 평균 간격을 기초로 상기 다운스케일의 인자를 설정할 수 있다.

[0015] 상기 출력 필터링부는 상기 화면의 출력을 구성하는 프레임 이미지들 간의 차이에 대한 사용자 시각 인지성을 기초로 특정 기준을 설정할 수 있다.

[0016] 상기 입출력 간격 처리부는 상기 사용자의 입력 및 상기 화면의 출력에 대한 타이밍들 간의 차이를 산출하고 상기 차이를 기초로 복수의 막대들로 구성된 분포를 생성할 수 있다.

[0017] 상기 입출력 간격 처리부는 상기 분포에서 음의 영역에 존재하는 최빈도 막대를 제거할 수 있다.

[0018] 상기 모델 피팅부는 상기 입출력 간격의 분포를 가우시안 분포로 생성하여 상기 선제성을 산출하고 상기 입출력 간격의 분포를 엑스-가우시안 분포를 포함하는 반응적 분포로 생성하여 상기 반응성을 산출할 수 있다.

[0019] 상기 모델 피팅부는 상기 선제성 및 상기 반응성을 기초로 사용자 인터페이스의 특성을 조절할 수 있다.

- [0020] 상기 모델 피팅부는 상기 사용자 인터페이스의 시각적 레이아웃 및 동작 기능을 조절할 수 있다.
- [0021] 실시예들 중에서, 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 방법은 사용자의 입력 및 화면의 출력을 기록하는 데이터 획득단계; 상기 사용자의 입력에 대한 입력 타이밍을 추출하는 입력 필터링단계; 상기 화면의 출력에 대한 특정 기준 이상의 시각적 변경을 개시하는 출력 타이밍을 추출하는 출력 필터링단계; 상기 사용자의 입력 및 상기 화면의 출력에 대한 타이밍들을 통해 입출력 간격의 분포를 생성하는 입출력 간격 처리단계; 및 상기 입출력 간격의 분포를 선제적 분포 및 반응적 분포로 조절하여 선제성 및 반응성을 추정하는 모델 피팅단계를 포함한다.
- [0022] 상기 데이터 획득단계는 특정 구간에 대해 상기 사용자의 시계열 입력 액션들을 입력 엔티티별로 기록하고 상기 화면의 시계열 출력을 프레임 이미지별로 기록할 수 있다.
- [0023] 상기 입력 필터링단계는 상기 사용자의 입력이 입력 이벤트의 생성을 기초로 이산적 타이밍을 추출할 수 있다.
- [0024] 상기 출력 필터링단계는 상기 출력 타이밍의 추출 전에 상기 화면의 출력을 기록하는 프레임 이미지의 해상도 또는 프레임레이트를 다운스케일 할 수 있다.

### 발명의 효과

- [0026] 개시된 기술은 다음의 효과를 가질 수 있다. 다만, 특정 실시예가 다음의 효과를 전부 포함하여야 한다거나 다음의 효과만을 포함하여야 한다는 의미는 아니므로, 개시된 기술의 권리범위는 이에 의하여 제한되는 것으로 이해되어서는 아니 될 것이다.
- [0027] 본 발명에 따른 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치 및 방법은 사용자의 입력 타이밍들과 화면상의 시각적 변화들 사이의 시간간격 분포를 측정하고 해당 분포를 반응적 분포, 예측적 분포 그리고 비상관 분포의 합으로서 어떻게 구성해낼 수 있는지 추정하여 사용자의 입력이 얼마나 반응적이었는지 예측적이었는지 알아낼 수 있으며, 그것이 화면의 특정 위치에 대해서 어떻게 측정되는지도 알아낼 수 있다.

### 도면의 간단한 설명

- [0029] 도 1은 본 발명에 따른 사용자 인터페이스 장치의 물리적 구성을 설명하는 도면이다.
- 도 2는 본 발명에 따른 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 방법을 설명하는 순서도이다.
- 도 3은 특정 기준 이상의 시각적 변경을 개시하는 출력 타이밍을 설명하는 도면이다.
- 도 4는 커서 이동을 동반한 버튼 입력에 따른 화면의 출력을 설명하는 도면이다.
- 도 5의 (a) 및 (b)는 사용자의 입력 및 화면의 출력에 대한 타이밍들 간의 차이인 IOI 산출 및 분포를 설명하는 도면이다.
- 도 6은 사용자의 버튼 입력 전략에 따른 IOI 분포를 설명하는 도면이다.
- 도 7은 Monte Carlo 방법을 사용하여 IOI 분포를 얻는 방법을 설명하는 도면이다.
- 도 8은 각 픽셀-게임 유형 조합에 대한 평균  $w_I$  (모든 참가자)를 히트 맵으로 보여주는 도면이다.
- 도 9는 각 게임에 대한 추정된 IOI 분포 매개변수와 게임 점수 간의 선형 회귀 결과를 보여주는 도면이다.
- 도 10은 모델 피팅에 대한 실험 결과를 설명하는 도면이다.
- 도 11은 연구 2에서 사용자가 클릭한 자체 확장 대상의 디자인을 설명하는 도면이다.
- 도 12는 연구 2의 결과로서 시행 완료 시간 및 오류율을 설명하는 도면이다.
- 도 13은 연구 2의 결과로서 반응성 및 선제성을 설명하는 도면이다.
- 도 14는 본 발명에 따른 사용자 인터페이스 장치의 시스템 구성을 설명하는 도면이다.
- 도 15는 본 발명에 따른 사용자 인터페이스 시스템을 설명하는 도면이다.

## 발명을 실시하기 위한 구체적인 내용

- [0030] 본 발명에 관한 설명은 구조적 내지 기능적 설명을 위한 실시예에 불과하므로, 본 발명의 권리범위는 본문에 설명된 실시예에 의하여 제한되는 것으로 해석되어서는 아니 된다. 즉, 실시예는 다양한 변경이 가능하고 여러 가지 형태를 가질 수 있으므로 본 발명의 권리범위는 기술적 사상을 실현할 수 있는 균등물들을 포함하는 것으로 이해되어야 한다. 또한, 본 발명에서 제시된 목적 또는 효과는 특정 실시예가 이를 전부 포함하여야 한다거나 그러한 효과만을 포함하여야 한다는 의미는 아니므로, 본 발명의 권리범위는 이에 의하여 제한되는 것으로 이해되어서는 아니 될 것이다.
- [0031] 한편, 본 출원에서 서술되는 용어의 의미는 다음과 같이 이해되어야 할 것이다.
- [0032] "제1", "제2" 등의 용어는 하나의 구성요소를 다른 구성요소로부터 구별하기 위한 것으로, 이들 용어들에 의해 권리범위가 한정되어서는 아니 된다. 예를 들어, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다.
- [0033] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결될 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다고 언급된 때에는 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 한편, 구성요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.
- [0034] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한 복수의 표현을 포함하는 것으로 이해되어야 하고, "포함하다" 또는 "가지다" 등의 용어는 실시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이며, 하나 또는 그 이상의 다른 특징이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0035] 각 단계들에 있어 식별부호(예를 들어, a, b, c 등)는 설명의 편의를 위하여 사용되는 것으로 식별부호는 각 단계들의 순서를 설명하는 것이 아니며, 각 단계들은 문맥상 명백하게 특정 순서를 기재하지 않는 이상 명기된 순서와 다르게 일어날 수 있다. 즉, 각 단계들은 명기된 순서와 동일하게 일어날 수도 있고 실질적으로 동시에 수행될 수도 있으며 반대의 순서대로 수행될 수도 있다.
- [0036] 본 발명은 컴퓨터가 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 코드로서 구현될 수 있고, 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피 디스크, 광 데이터 저장 장치 등이 있다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.
- [0037] 여기서 사용되는 모든 용어들은 다르게 정의되지 않는 한, 본 발명이 속하는 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한 이상적이거나 과도하게 형식적인 의미를 지니는 것으로 해석될 수 없다.
- [0039] 본 발명은 5단계로 수행될 수 있다. 첫번째 단계는 사용자의 입력 로그와 컴퓨터 화면을 기록하는 데이터 수집이다. 두번째 단계는 입력 필터링으로 주어진 입력 로그에서 사용자가 의도한 버튼 입력 타이밍을 추출한다. 세번째 단계는 기록된 화면 비디오의 각 프레임에 대해 사용자가 감지할 수 있는 시각적 변화를 식별하는 출력 필터링이다. 네번째 단계는 버튼 입력 및 출력 쌍에서 IOI 분포를 얻는 IOI 추출이다. 마지막 단계는 획득한 IOI 분포에 사후적 분포와 사전적 분포를 피팅하여 반응성과 선제성을 추정하는 모델 피팅이다.
- [0041] 도 1은 본 발명에 따른 사용자 인터페이스 장치의 물리적 구성을 설명하는 도면이다.
- [0042] 도 1을 참조하면, 사용자 인터페이스 장치(100)는 데이터 획득부(110), 입력 필터링부(120), 출력 필터링부(130), 입출력 간격 처리부(140), 모델 피팅부(150) 및 제어부(도 1에 미도시함)를 포함할 수 있다.
- [0043] 데이터 획득부(110)는 사용자의 입력 및 화면의 출력을 기록할 수 있다. 데이터 획득부(110)는 특정 구간에 대



해 사용자의 시계열 입력 액션들을 입력 엔티티별로 기록하고 화면의 시계열 출력을 프레임 이미지별로 기록할 수 있다. 여기에서, 입력 엔티티는 버튼에 해당할 수 있다.

[0044] 일 실시예에서, 데이터 획득부(110)는 사용자의 입력 및 화면의 출력을 특정 구간(T) 동안 기록할 수 있다. 기록된 입력 동작은 버튼 입력  $b^{type}$  (누른 경우 1, 누르지 않은 경우 0)과 커서 좌표  $x$  및  $y$  (사용 가능한 경우 픽셀 단위)의 시계열이다. 사용자가 다른 버튼을 누르면 데이터가 각 버튼에 대해 별도로 기록될 수 있다. 예를 들면, 마우스 왼쪽 버튼의 경우  $b^l$ , 키 “a”의 경우  $b^a$ 로 기록될 수 있다. 로깅된 컴퓨터의 출력은 로깅 기간 동안 시각적 디스플레이에서 사용자에게 표시되는 모든 프레임 이미지( $v$ ) (여기서는 24비트 RGB)일 수 있다. 프레임 이미지( $v$ )의 픽셀 해상도는 대상 애플리케이션 및 하드웨어에 따라 다를 수 있다.

[0045] 입력 로깅의 프레임 속도를  $f$ , 로깅된 프레임 수  $N$ 를  $f \cdot T$ 라고 하면 최종 로깅 데이터는 다음의 수학적 식 1로 표현될 수 있다.

[0046] [수학적 식 1]

$$\begin{aligned} b^{type} &= [b_1^{type}, b_2^{type}, \dots, b_N^{type}] \\ x &= [x_1, x_2, \dots, x_N] \\ y &= [y_1, y_2, \dots, y_N] \\ v &= [v_1, v_2, \dots, v_N] \\ t &= [1/f, 2/f, \dots, T] \end{aligned}$$

[0047]

[0048] 여기에서,  $t$ 는 타임스탬프이다.

[0049] 본 실시예에서는 OBS(Open Broadcaster Software)( $f=60\text{Hz}$ )를 이용하여 화면 영상을 기록하였고, pyuput 라이브러리를 이용하여 OBS 콜백 함수가 실행되는 순간 버튼 입력과 커서 좌표를 함께 기록하였다. 전체 스크립트는 Python으로 구현되고, 비디오는 mp4 형식으로 기록되고 버튼 입력 및 커서 좌표는 csv 형식으로 기록된다. 일반적인 해상도 즉,  $1680 \times 1050$ 에서 30분동안 기록하면 원시 데이터 파일은 약 3GB가 된다.

[0050] 입력 필터링부(120)는 사용자의 입력에 대한 입력 타이밍을 추출할 수 있다. 입력 필터링부(120)는 사용자가 발생시킨 모든 입력 이벤트의 생성을 기초로 이산적 타이밍을 추출할 수 있다. 여기에서, 입력 이벤트의 생성은 특정 아이콘을 클릭해서 실행하는 행위를 포함하고, 빈공간을 클릭해서 아무 입력 이벤트가 발생하지 않은 것을 제외할 수 있다.

[0051] 일 실시예에서, 입력 피터링부(120)는 특정 유형의 버튼  $b^{type}$ 의 입력 로그 데이터에서 사용자가 버튼 입력 이벤트를 생성하려고 의도한 개별 타이밍을 추출할 수 있다. 사용자가 버튼을 누른 순간부터 손을 떼는 순간까지 약 수십 ms가 소요되며 그 사이 사용자가 의도한 입력 타이밍에 가장 가까운 순간이 있다. 여기서는 버튼 입력 로그가 0에서 1로 변경되는 순간, 즉 버튼이 눌리기 시작하는 순간을 사용자가 의도한 버튼 입력 활성화 순간으로 가정할 수 있다. 즉, 사용자가 의도한 버튼 입력 타이밍은 다음의 수학적 식 2와 같이 결정될 수 있다.

[0052] [수학적 식 2]

$$\text{Find all } i \text{ where, } (b_i^{type} - b_{i-1}^{type}) = 1$$

[0053]

[0054] 결과적으로, 특정 유형의 버튼  $b^{type}$ 을 사용하는 사용자가 의도한 입력 타이밍의 타임스탬프는 다음의 수학적 식 3과 같이 정의될 수 있다.

[0055] [수학적 식 3]

$$t_{input}^{type} = [t_1^{type}, t_2^{type}, \dots, t_{B^{type}}^{type}]$$

[0056]

[0057] 여기에서,  $t_{input}^{type}$ 는 특정 유형의 버튼을 사용하는 사용자가 의도한 입력 타이밍이고,  $B^{type}$ 는 해당 버튼으로 감지

된 입력 이벤트의 수이다.

- [0058] 출력 필터링부(130)는 화면의 출력에 대한 특정 기준 이상의 시각적 변경을 개시하는 출력 타이밍을 추출할 수 있다. 출력 필터링부(130)는 사용자의 입력에 의해 촉발된 화면상의 시각적 변화, 사용자의 반응적 입력을 촉발시키는 화면상의 시각적 변화, 사용자의 선제적 입력을 촉발시키는 화면상의 시각적 변화, 사용자의 입력과 전혀 무관한 화면상의 시각적 변화를 통한 화면의 출력을 기초로 출력 타이밍을 추출할 수 있다. 출력 필터링부(130)는 출력 타이밍의 추출 전에 화면의 출력을 기록하는 프레임 이미지의 해상도 또는 프레임레이트를 다운스케일 할 수 있다. 출력 필터링부(130)는 사용자의 입력에 대한 평균 간격을 기초로 다운스케일의 인자를 설정할 수 있다. 예를 들면, 출력 필터링부(130)는 평균 간격이 적으면 고해상도로, 평균 간격이 넓으면 저해상도로 프레임 이미지의 해상도를 다운스케일 할 수 있다. 출력 필터링부(130)는 화면의 출력을 구성하는 프레임 이미지들 간의 차이에 대한 사용자 시각 인지성을 기초로 특정 기준을 설정할 수 있다.
- [0059] 기록한 화면의 해상도가  $W \times H$ 이면 총  $W \cdot H$  픽셀이 사용자에게 정보를 전달한다. 출력 필터링부(130)는 보다 효율적인 분석을 위해 스케일 인자  $K$ 를 사용하여 화면의 출력을 기록하는 프레임 이미지의 해상도를 다운스케일하여 해상도를 축소할 수 있다. 이때, 이미지의 해상도는  $W/K \times H/K$ 가 된다. 다운스케일 인자  $K$ 는 축소 후에도 화면에서 중요한 시각적 이벤트를 완전히 인지할 수 있는 값으로 설정할 수 있으며, 여기서는 32로 결정할 수 있다.
- [0060] 입출력 간격 처리부(140)는 사용자의 입력 및 화면의 출력에 대한 타이밍들을 통해 입출력 간격의 분포를 생성할 수 있다. 입출력 간격 처리부(140)는 사용자의 입력 및 화면의 출력에 대한 타이밍들 간의 차이를 산출하고 차이를 기초로 복수의 막대들로 구성된 분포를 생성할 수 있다. 분포에서 음의 영역에 존재하는 최빈도 막대는 일정하고 짧은 레이턴시를 가지는 사용자 입력에 의해 단순하게 트리거되는 시각적 출력에 해당하기 때문에, 입출력 간격 처리부(140)는 분포에서 음의 영역에 존재하는 최빈도 막대를 제거할 수 있다.
- [0061] 모델 피팅부(150)는 입출력 간격의 분포를 선제적 분포 및 반응적 분포로 조절하여 선제성 및 반응성을 추정할 수 있다. 모델 피팅부(150)는 입출력 간격의 분포를 가우시안 분포로 생성하여 선제성을 산출하고 입출력 간격의 분포를 엑스-가우시안 분포를 포함하는 반응적 분포로 생성하여 반응성을 산출할 수 있다. 모델 피팅부(150)는 선제성 및 반응성을 기초로 사용자 인터페이스의 특성을 조절할 수 있다. 여기에서, 모델 피팅부(150)는 사용자 인터페이스의 시각적 레이아웃 및 동작 기능을 조절할 수 있다.
- [0062] 제어부(도 1에 미도시함)는 사용자 인터페이스 장치(100)의 전체적인 동작을 제어하고, 데이터 획득부(110), 입력 필터링부(120), 출력 필터링부(130), 입출력 간격 처리부(140) 및 모델 피팅부(150) 간의 제어 흐름 또는 데이터 흐름을 관리할 수 있다.
- [0064] 도 2는 본 발명에 따른 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 방법을 설명하는 순서도이다.
- [0065] 도 2를 참조하면, 사용자 인터페이스 장치(100)는 데이터 획득부(110)를 사용자의 입력 및 화면의 출력을 기록할 수 있다(단계 S210). 사용자 인터페이스 장치(100)는 입력 필터링부(120)를 통해 사용자의 입력에 대한 입력 타이밍을 추출할 수 있다(단계 S220).
- [0066] 또한, 사용자 인터페이스 장치(100)는 출력 필터링부(130)를 통해 화면의 출력에 대한 특정 기준 이상의 시각적 변경을 개시하는 출력 타이밍을 추출할 수 있다(단계 S230). 사용자 인터페이스 장치(100)는 입출력 간격 처리부(140)를 통해 사용자의 입력 및 화면의 출력에 대한 타이밍들을 통해 입출력 간격의 분포를 생성할 수 있다(단계 S240).
- [0067] 사용자 인터페이스 장치(100)는 모델 피팅부(150)를 통해 입출력 간격의 분포를 선제적 분포 및 반응적 분포로 조절하여 선제성 및 반응성을 추정할 수 있다(단계 S250).
- [0069] 이하, 도 3 내지 13을 참조하여, 본 발명에 따른 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 장치 및 방법에 대해 보다 자세히 설명한다.
- [0070] 본 발명에 따른 사용자 인터페이스 방법은 상호작용에서 얻은 사용자의 입력 및 화면의 출력 간 시간 간격(IOI)을 분석하여 사용자 반응성 및 선제성을 평가할 수 있다.
- [0071] 버튼은 사용자의 지속적인 움직임을 불연속적인 신호로 변환하여 컴퓨터로 전송하는 입력장치이다. 사용자가

버튼을 누르는 목적은 컴퓨터에 자신의 의사를 전달하는 것이다. 버튼을 누르기 위해서는 사용자가 먼저 컴퓨터의 현재 상태를 인지하고 버튼을 누를 시기를 결정해야 한다. 버튼을 언제 누를지 결정한 후 버튼 누름 동작이 실행된다. 컴퓨터의 상태를 인지한 후 결정을 내리는 일련의 과정에 비해 버튼을 누르는 동작 자체는 매우 짧고 사용자 성능에 거의 영향을 미치지 않으므로 일반적으로 무시할 수 있다.

[0072] 컴퓨터에서 예상치 못한 피드백을 받은 후 최대한 빨리 버튼을 누르는 작업을 반응이라고 한다. 반응작업의 IOI는 전통적으로 응답시간 또는 반응시간이라고 하며 IOI의 분포는 기하급수적으로 수정된 가우시안(ex-Gaussian) 또는 로그 정규(log-normal)와 같이 긴 오른쪽 꼬리를 가진 단봉 분포를 따르는 것으로 알려져 있다. Hick-Hyman 법칙에 따르면 인간의 정보 처리 능력은 일정하기 때문에 반응 과제에서 IOI의 평균은 누를 수 있는 대체 버튼 수의 로그에 비례하여 증가한다. 반면에 드리프트-확산 모델은 반응을 순차적 샘플링 프로세스로 설명한다.

[0073] 이 모델에 따르면 사용자는 실시간으로 의사 결정을 위한 증거를 축적한다. 도중에 축적된 증거의 양이 미리 정해진 결정 경계를 초과하면 사용자는 해당 버튼을 누른다. Hick-Hyman 법칙과 달리 드리프트-확산은 반응 IOI의 분포를 시뮬레이션할 수 있다.

[0074] 사용자에게 버튼을 누르는 정확한 타이밍을 묻는 메시지가 표시되면 작업을 예상(사전 버튼 입력)이라고 한다. 예상에는 깜박이는 표적 획득 및 움직이는 표적 획득 작업이 포함된다. 전자의 작업은 깜박이는 대상이 다시 나타날 때를 예상하여 그 순간에 버튼을 누르는 것이다. 후자의 작업은 움직이는 표적이 조준 영역에 들어갈 때를 예상하여 그 순간에 버튼을 누르는 것이다.

[0075] 반응 작업에서 사용자의 목표는 주어진 자극에 가능한 빨리 반응하는 것이다. 따라서 반응 작업의 주요 성능 매트릭은 IOI의 평균이며 일반적으로 좋은 인터페이스는 낮은 평균을 가능하게 하는 인터페이스 즉, 사용자가 목표를 더 빨리 달성할 수 있도록 도와주는 인터페이스이다. 드리프트-확산 모델에 따르면 사용자의 평균 IOI가 낮기 위해서는 의사 결정의 증거가 축적되는 비율 즉, 드리프트 비율이 높아야 한다. 반면에, 예측 작업에서 사용자의 버튼 입력 타이밍은 평균적인 작업에서 필요한 버튼 입력 타이밍에 가깝기 때문에 IOI 분포의 평균은 더 이상 중요한 성능 매트릭이 아니다. 대신, 사용자는 자신의 버튼 입력 타이밍이 작업에서 요구되는 버튼 입력 타이밍에 대해 낮은 변동성을 갖기를 원한다. 즉, 예측 작업의 주요 성능 매트릭은 IOI 분포의 분산(또는 표준 편차)이다.

[0076] IOI 분포의 분산이 크면 사용자는 컴퓨터에서 요구하는 버튼 입력 타이밍을 추정하기 어렵다. 예를 들어, 깜박이는 표적 획득에서 깜박임의 반복 주기가 길수록 사용자가 다음 버튼 입력의 타이밍을 추정하기 어려워지며 이를 사용자 내부 시계의 스칼라 속성이라고 한다. 한편, 움직이는 표적 획득 작업에서는 표적을 관찰할 수 있는 시간 즉, 큐 시청 시간이 짧을수록 사용자의 버튼 입력 타이밍 추정 성능이 낮아진다. 특히, 시스템 대기 시간이 길면 사용자가 끝까지 목표 이동을 관찰하지 않고 버튼을 눌러야 하기 때문에 큐 시청 시간이 더욱 단축될 수 있다.

[0077] 사용자의 두 가지 대표적인 버튼 입력 전략(반응형 및 선제형), 수학적 모델 및 각각의 성능에 영향을 미치는 요소는 다음의 표 1로 정리할 수 있다.

[0078] [표 1]

버튼 입력 전략	IOI의 분포	품질측정	품질에 영향을 미치는 인자
반응	ex-가우시안 또는 로그-정규	분포의 평균	자극 친숙도, 강도, 호환성, 시각적 혼란, 대안의 수
예기	가우시안	분포의 분산	큐 시청 시간, 깜박이는 시간, 시스템 레이턴시, 감각 큐의 신뢰성, 사용자 내부 시계의 정확도

[0079]

[0080] 본 발명에서는 제어되지 않은 자극 및 무제한 사용자 버튼 누름과 관련된 일상적인 상호작용에서 사용자의 버튼 누름 성능을 정량화하는 기술을 제안 및 구현한다.

[0081] 이를 위한 첫번째 단계는 데이터 획득 단계로, 사용자의 입력 로그와 컴퓨터 화면을 기록한다.

[0082] 데이터 수집 단계에서 사용자 입력 작업 및 컴퓨터 출력은 특정 기간 동안 기록된다. 기록된 입력 동작은 버튼 입력  $b^{type}$  (누른 경우 1, 누르지 않은 경우 0)과 커서 좌표 및 (사용 가능한 경우 픽셀 단위)의 시계열이다. 사용자가 다른 버튼을 누르면 데이터가 각 버튼에 대해 별도로 기록되어야 한다(예: 마우스 왼쪽 버튼의 경우  $b^{ml}$ , 키 "a"의 경우  $b^a$ ). 로깅된 컴퓨터의 출력은 로깅 기간 동안 시각적 디스플레이에서 사용자

에게 표시되는 모든 프레임 이미지 (이 연구에서는 24비트 RGB)이다. 의 픽셀 해상도는 대상 애플리케이션 및 하드웨어에 따라 다를 수 있다.

[0083] 입력 로깅의 프레임 속도를  $f$ , 로깅된 프레임 수  $N$ 를  $f \cdot T$  라고 하고 최종 로깅 데이터는 다음과 같다.

$$\begin{aligned} b^{type} &= [b_1^{type}, b_2^{type}, \dots, b_N^{type}] \\ x &= [x_1, x_2, \dots, x_N] \\ y &= [y_1, y_2, \dots, y_N] \\ v &= [v_1, v_2, \dots, v_N] \\ t &= [1/f, 2/f, \dots, T] \end{aligned}$$

[0084]

[0085] 여기서  $t$ 는 타임스탬프이다. 본 연구에서는 OBS(Open Broadcaster Software)( $f=60$  Hz)를 이용하여 화면 영상을 기록하였고, pynput 라이브러리를 이용하여 OBS 콜백 함수가 실행되는 순간 버튼 입력과 커서 좌표를 함께 기록하였다. 전체 스크립트는 Python으로 구현된다. 비디오는 mp4 형식으로 기록되고 버튼 입력 및 커서 궤적은 csv 형식으로 기록된다. 일반적인 해상도(즉,  $1680 \times 1050$ )에서 이 방법으로 30분 동안 기록하면 원시 데이터 파일은 약 3GB가 된다.

[0086] 두 번째 단계는 입력 필터링으로 주어진 입력 로그에서 사용자가 의도한 버튼 입력 타이밍을 추출한다.

[0087] 입력 필터링 단계는 특정 유형의 버튼  $b^{type}$ 의 입력 로그 데이터에서 사용자가 버튼 입력 이벤트를 생성하려고 의도한 개별 타이밍을 추출해야 한다. 선행 연구에 따르면 버튼을 누른 순간부터 손을 떼는 순간까지 약 수십 ms가 소요되며 그 사이 사용자가 의도한 입력 타이밍에 가장 가까운 순간이 있다. 본 연구에서는 버튼 입력 로그가 0에서 1로 변경되는 순간, 즉 버튼이 눌리기 시작하는 순간을 사용자가 의도한 버튼 입력 활성화 순간으로 가정하였다. 즉, 사용자가 의도한 버튼 입력 타이밍은 다음과 같이 결정된다.

$$\text{Find all } i \text{ where, } (b_i^{type} - b_{i-1}^{type}) = 1$$

[0088]

[0089] 결과적으로 버튼을 사용하는 사용자가 의도한 입력 타이밍  $t_{input}^{type}$ 의 타임스탬프는 다음과 같이 쓸 수 있다.

$$t_{input}^{type} = [t_1^{type}, t_2^{type}, \dots, t_{B^{type}}^{type}]$$

[0090]

[0091] 여기서  $B^{type}$ 는 해당 버튼 으로 감지된 입력 이벤트의 수이다.

[0092] 세번째 단계는 기록된 화면 비디오의 각 프레임에 대해 사용자가 감지할 수 있는 시각적 변화를 식별하는 출력 필터링이다.

[0093] 도 3 및 도 4를 참조하면, 기록한 디스플레이의 해상도가  $W \times H$  이면 총  $W \cdot H$  픽셀이 사용자에게 정보를 전달한다. 보다 효율적인 분석을 위해 스케일 팩터  $K$ 를 사용하여 비디오의 해상도를 축소했다. 그러면 동영상의 해상도는  $W/K \times H/K$ 가 된다.  $K$ 은 축소 후에도 화면에서 중요한 시각적 이벤트를 완전히 감지할 수 있는 값으로 설정해야 한다. 이 연구에서  $K$ 의 값은 모든 후속 연구에서 32로 결정되었다. 다음으로, 각 프레임의 각 픽셀의 (R, G, B) 값을 CIELAB 형식(L, a, b)으로 변환한다. CIELAB 공간에서 색상 벡터 간의 거리는 인간이 실제로 인지하는 색상 차이에 더 가깝기 때문에 화면에서 발생하는 중요한 시각적 이벤트를 더 잘 감지할 수 있다. 본 연구에서는 colormath 라이브러리의 convert\_color 함수를 이용하여 CIELAB 형식으로 변환하였다.

[0094] 지금까지  $M(=W/K \cdot H/K)$  픽셀 각각에 대한 (L, a, b) 값의 시계열이 있다.  $i$ 번째 시간 단계에서  $j$ 번째 픽셀의 색상 값을  $(L_i^j, a_i^j, b_i^j)$ . 그런 다음 인접한 프레임 간의 색상 차이에 대한 시계열을 다음의 수학적 식 4와 같이 얻는다.

[0095] [수학식 4]

$$(\Delta L_i^j, \Delta a_i^j, \Delta b_i^j) = (L_i^j - L_{i-1}^j, a_i^j - a_{i-1}^j, b_i^j - b_{i-1}^j)$$

[0096]

[0097] 이 시계열에서는 L, a 및 b 중 적어도 하나의 채널에서 색상 값이 특정 임계값  $\delta$  이상으로 변경되는 모든 순간을 감지한다. 도 3의 경우, 감지된 각 순간(i/f)에서 사용자의 버튼 누름을 유발할 수 있는 중요한 시각적 이벤트가 해당 픽셀에서 사용자에게 전달된다. 본 연구에서는 CIELAB 공간에서 약 5 유클리드 거리의 색차를 인간이 충분히 인지할 수 있다고 하여  $\delta$ 를 5로 설정하였다. 결과적으로 각 j번째 픽셀에 대해 얻은 중요한 시각적 출력  $t_{output}^j$ 의 타임스탬프는 다음의 수학식 5와 같이 표현될 수 있다.

[0098] [수학식 5]

$$t_{output}^j = [t_1^j, t_2^j, \dots, t_{pj}^j]$$

[0099]

[0100] 여기에서,  $P^j$ 는 j번째 픽셀에서 감지된 시각적 이벤트의 수이다.

[0101] 상기 수학식 4 및 5는 디스플레이의 특정(절대) 위치에서 발생하는 시각적 이벤트가 사용자의 버튼 입력을 트리거할 수 있다는 전제를 기반으로 한다. 그러나 이 가정이 성립하지 않는 특별한 상황이 있을 수 있다. 예를 들면, 사용자가 컴퓨터 마우스를 사용하여 디스플레이의 왼쪽 하단 모서리에서 커서를 이동하고 오른쪽 상단 모서리에 나타난 대상을 클릭하는 상황이다. 이때, 사용자의 버튼 입력이 화면 우측 상단에 대상이 나타난 이벤트에 의해 발생했다고 보기는 어렵다. 오히려 커서가 타겟에 도달했거나 상대적으로 타겟이 커서에 가까워졌기 때문에 사용자의 버튼 입력이 발생했다고 보는 것이 더 자연스럽다. 따라서 마우스 버튼 입력을 트리거할 수 있는 출력을 추출하는 과정에서 커서 고정 좌표계를 참조로 사용했다. 보다 구체적으로, 이는 상기 수학식 4 및 5의 과정을 수행하기 전에 도 4와 같이, 스크린 비디오의 각 프레임  $v_i$ 의 중심을 커서 위치( $x_i, y_i$ )로 옮기는 것으로 이루어진다. 변환으로 비워진 픽셀의 (L, a, b) 값은 모두 (0,0,0)으로 채워진다.

[0102] 네번째 단계는 버튼 입력 및 출력 쌍에서 IOI 분포를 얻는 입출력 간격 처리단계이다.

[0103] 본 연구에서 IOI는 출력이 발생한 시간  $t_{out}$ 과 사용자가 해당 버튼 입력  $t_{input}$ 을 수행한 시간의 차이로 정의될 수 있다. 일반적으로 다중 입력과 다중 출력이 관찰되며, 이들의 조합으로 다중 IOI, 즉 IOI 분포를 정의할 수 있다. 그러나 특정 픽셀에서 관찰되는 시각적 출력의 수가 너무 적으면 IOI 분포를 얻는 것이 의미가 없으므로 픽셀에 대한 분석을 건너뛸 수 있다. 보다 구체적으로, 본 연구에서는 모든 픽셀에서 시각적 출력 수가  $P_j$ 의 최댓값의 1/10 미만인 픽셀을 건너뛰었다.

[0104] 그런 다음 건너뛰지 않은 픽셀의 경우 각 버튼 type-픽셀 번호 j조합을 살펴보면, 도 5를 참조하여, 하나의 버튼 type-픽셀 조합의 경우 버튼 입력 타이밍  $t_{input}^{type}$ 과 시각적 출력 타이밍  $t_{output}^j$ 축을 따라 정렬할 수 있다. 이때 IOI는  $t^{type}$ 와  $t^j$ (수학식 3 및 5 참조)의 각 조합에 따라 계산할 수 있지만, 인간의 버튼 입력 과정에 대해 한 가지 가정을 하여 샘플링된 IOI의 수를 줄였다. 결과적으로 각 q번째 시각적 출력은 시간 축에서 가장 가까운 버튼 입력  $t_{k*}^{type}$ 과 결합될 수 있다. IOI는 다음의 수학식 6과 같이 계산할 수 있다.



[0105] [수학식 6]

$$IOI_q^{type,j} = (t_{k^*}^{type} - t_q^j)$$

$$\text{where } k^* = \arg \min_k |t_k^{type} - t_q^j|$$

[0106]

[0107] 그런 다음 다음의 수학식 7과 같이 하나의 버튼 type-픽셀 조합에 대한 총  $P^j$  (시각적 출력 수) IOI로 구성된 IOI 분포를 얻을 수 있다.

[0108] [수학식 7]

$$IOI^{type,j} \xleftarrow{\text{normalize}} [IOI_1^{type,j}, IOI_2^{type,j}, \dots, IOI_{P^j}^{type,j}]$$

[0109]

[0110] 상기 수학식 7을 통해 도 4의 (b)와 같이, 확률 질량 함수로 변화되도록 정규화될 수 있다.

[0111] 마지막 단계는 획득한 IOI 분포에 사후적 분포와 사전적 분포를 피팅하여 반응성과 선제성을 추정하는 모델 피팅 단계이다.

[0112] 상기 수학식 7에서 구한 IOI 분포는 여러 하위 분포의 합으로 표현될 수 있다. 도 6을 참조하면, (a)는 사용자가 시각적 출력에 대한 응답으로 버튼을 능동적으로 누를 때 관찰되는 IOI 분포이다. 인간의 능동적인 버튼 입력 성능에 대한 이전 연구를 참조하면 하기 수학식 8과 같이 가우스 분포로 가정할 수 있다.

[0113] [수학식 8]

$$IOI_{proactive} = \frac{1}{\sigma_P \sqrt{2\pi}} \exp \left( -\frac{1}{2} \left( \frac{t - \mu_P}{\sigma_P} \right)^2 \right)$$

[0114]

[0115] 여기에서,  $\mu_P$ 는 분포의 평균이고,  $\sigma_P$ 는 분포의 표준편차이다. 이 두 가지는 함께 그룹화되어 사전 배포의 매개 변수  $\theta_P$ 로 정의될 수 있다.

[0116] 도 6의 (b)는 사용자가 시각적 출력에 대한 응답으로 버튼을 반응적으로 누를 때 관찰되는 IOI 분포이다. 인간의 반응 시간에 대한 이전 연구를 참조하면 하기 수학식 9의 ex-Gaussian 분포로 가정될 수 있다.

[0117] [수학식 9]

$$IOI_{reactive} = \frac{1}{2K_R} \exp \left( \frac{1}{2K_R^2} - \frac{t - \mu_R}{\sigma_R K_R} \right) \text{erfc} \left( -\frac{(t - \mu_R)/\sigma_R - 1/K_R}{\sqrt{2}} \right), \text{ where } K > 0 \text{ and } \text{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt$$

[0118]

[0119] 여기에서,  $\mu_R$ ,  $\sigma_R$ ,  $K_R$ 은 분포의 모양을 결정하는 모수이다. 이 세 가지는 함께 그룹화되어 반응 분포의 모수  $\theta_R$ 로 정의될 수 있다. 반응 분포의 평균은  $\mu_R$ 가 아니라  $(\mu_R + \sigma_R K_R)$ 이다. 이하, 반응 분포의 실제 평균을  $\hat{\mu}_R (= \mu_R + \sigma_R K_R)$ 으로 표시한다.

[0120] 도 7은 사용자의 버튼 입력과 무관한 시각적 출력에서 얻은 IOI의 분포를 나타낸다. 사용자의 버튼 입력과 무관한 시각적 출력은 시간 축에 균일하게 분포되어 있다고 가정할 수 있다. 사용자의 버튼 입력 타이밍을 이미 알고 있으므로 Monte Carlo 방법을 사용하여 균일하게 분포된 관련 없는 출력에서 IOI 분포를 얻는 방법을 시뮬레이션할 수 있다. 이렇게 얻은 분포를 정규화하고  $I_{irrelevant}$ 으로 가정하였다.

[0121] 마지막으로 일부 출력은 사용자 입력을 트리거하지 않지만 단순히 사용자 입력에 의해 트리거될 수 있다. 예를

들어, 사용자가 닫기 버튼을 누르면 애플리케이션 창이 닫히고, 슈팅 게임에서 사용자가 적을 쏘면 적이 사라진다. 분석을 단순화하기 위해 이 연구에서는 사용자에게 의해 트리거된 컴퓨터 출력이 짧고 일정한 대기 시간으로 생성된다고 가정한다. 이는 수학적 7에서 구한 IOI 분포에서 음의 영역에서 가장 높은 피크로 관찰될 수 있다. 이는 사용자의 반응성 및 선제성을 추정하는 것과는 무관하므로 분포에서 피크를 제거하고 후속 분석을 수행한다(도 5 참조).

[0122] 본 발명의 최종 목표는 다음의 수학적 10과 같이  $I_{proactive}$ ,  $I_{reactive}$  및  $I_{irrelevant}$ 의 합으로 피크 제거 IOI 분포를 표현하는 것이다.

[0123] [수학적 10]

$$IOI^{type,j} = w_R \cdot IOI_{reactive} + w_P \cdot IOI_{proactive} + w_I \cdot IOI_{irrelevant}$$

[0124] where,  $w_R + w_P + w_I = 1$

[0125] 여기에서,  $w_R$ ,  $w_P$ ,  $w_I$ 는 각 하위 분포의 가중치를 나타낸다.

[0126] 상기 수학적 10에서 각 분포의 가중치( $w_R$ ,  $w_P$ ,  $w_I$ )와 반응 및 사전 분포의 매개변수( $\theta_R$ ,  $\theta_P$ )를 찾아야 한다. 이는 혼합 분포 모델을 경험적 분포에 최적으로 맞추는 문제이며, EM(expectation-maximization) 방식을 통해 수행할 수 있다. 보다 구체적으로, EM을 통해 주어진 IOI 분포를 관찰할 가능성을 최대화하는 매개변수와 가중치를 얻을 수 있다. 이는 다음의 수학적 11로 표현될 수 있다.

[0127] [수학적 11]

$$\theta_R^*, \theta_P^* = \arg \max_{\theta_R, \theta_P} [\mathcal{L}(\theta_R, \theta_P; IOI^{type,j})]$$

[0128]

[0129] 이때 특정  $IOI^{type,j}$  인스턴스가 어느 부분에서 유래하는지 모르기 때문에 위의 수학적 11에서 가능도(likelihood)를 계산할 수 없다. EM 체계에서 우리는 각 IOI 인스턴스가 특정 분포에서 잠재적인 변수 Z로 온 것으로 믿어지는 확률을 정의한다. 3개의 하위 분포가 있으므로 Z은 크기가  $3 \times P^j$ 인 행렬이다. EM 기법은  $\theta_R$ ,  $\theta_P$  및 Z를 추정하기 위해 교대로 수행되는 기대 단계와 최대화 단계의 두 단계로 구성될 수 있다. 각 하위 분포에 대해 추정된 을 평균하면 해당 하위 분포의 가중치이다(즉,  $w_R$ ,  $w_P$ ,  $w_I$ ). 건너뛰지 않은 모든 버튼 type-픽셀 조합에 대해 이 프로세스를 반복한다.

[0130] 마지막의 모델 피팅단계까지 완료하면, 손에 있는 각 버튼 유형 픽셀에 대해 다음의 정보가 있을 수 있다.

[0131]  $w_R$ 는 픽셀이 다른 유형의 연관과 비교하여 사용자의 반응 입력 프로세스와 얼마나 강하게 연관되어 있는지 나타낸다.

[0132]  $w_P$ 는 픽셀이 다른 유형의 연결과 비교하여 사용자의 사전 입력 프로세스와 얼마나 강하게 연결되어 있는지 나타낸다.

[0133]  $w_I$ 는 픽셀이 다른 유형의 연관과 비교하여 사용자의 입력 프로세스와 얼마나 관련이 없는지를 나타낸다.

[0134]  $\theta_R$ 는 사용자가 시도한 반응성 입력의 특성과 성능을 설명하며, 사용자의 반응 성능을 가장 잘 설명하는 값은  $I_{reactive}$  분포의 평균이므로  $\hat{\mu}_R (= \mu_R + \sigma_R K_R)$ 이다.  $\hat{\mu}_R$ 가 높을수록 사용자가 예상치 못한 컴퓨터 출력에 빠르게 반응하는 것이 더 어려웠다.

[0135]  $\theta_P$ 는 사용자가 시도하는 능동적 입력의 특성과 성능을 설명하며,  $\theta_P$ 에 속하는 두 개의 파라미터( $\mu_P$ ,  $\sigma_P$ ) 중  $\sigma_P$ 가 가장 중요하다.  $\sigma_P$ 가 높을수록 사용자가 컴퓨터의 출력을 예상하기가 더 어려웠다.

[0136] 위의 해석에 따르면  $w_R$  과  $w_P$  는 사용자의 반응성과 선제성을 나타낼 수 있다. 그러나  $w_R$ ,  $w_P$  은  $w_R + w_P + w_I = 1$ , 즉 관련 없는 출력이 많을 때  $w_I$  가 증가할수록 감소한다. 그러나 반응성 및 선제성은 사용자 성능을 특성화하는 것을 목표로 하므로 관련 없는 출력의 유무에 관계없이 계산할 수 있어야 한다. 따라서 대응하는 버튼 유형-픽셀 조합에 대한 반응성 R 및 선제성 P를 다음의 수학적 식 12와 같이 정의하고 계산할 수 있다.

[0137] [수학적 식 12]

$$R^{type,j} = \frac{w_R}{w_R + w_P} \text{ and } P^{type,j} = \frac{w_P}{w_R + w_P}$$

[0138]

[0139] 여기에서, R 및 P는 각 픽셀에 대해 얻어지기 때문에 특정 버튼 유형에 대해 R 및 P의 히트맵(heatmap)을 그릴 수 있다. 히트맵에서 참가자가 더 반응적이거나 능동적으로 반응한 화면의 픽셀을 이해할 수 있다. 또한, 모든 픽셀에서 구한 R과 P를 평균하여 특정 버튼 종류에 대한 R과 P의 대표값을 구하는 것도 가능하다.

[0141] 연구(STUDY) 1: 실시간 비디오 게임하기

[0142] 연구 1에서 참가자들은 두 개의 실시간 비디오 게임을 한다. 그 게임들은 버튼입력으로 하는 리듬게임 장르에 속한다. 이때 본 발명에 의해 제안된 기법을 통해 기록된 화면 영상과 버튼 입력 로그를 분석한다. 그런 다음 추정된 반응성 R, 선제성 P, IOI 분포의 매개변수( $\theta_R$  및  $\theta_P$ )가 각 참가자의 게임 점수와 어떻게 상관되는지 조사한다.

[0143] 1.1 데이터 획득

[0144] 참가자 20명(여자 7명, 남자 13명)을 모집했다. 이들의 평균 나이는 23.9세( $\sigma=3.16$ )였다. 이 중 18명의 참가자는 리듬 게임을 해본 경험이 있었다.

[0145] 실험은 단일 독립 변수인 게임 유형을 사용하여 피험자 내 설계를 따랐다. 독립 변수의 세 가지 수준은 다음과 같다.

[0146] 게임 유형: Dino, Tabsonic 1(사운드 켜짐), Tabsonic 2(사운드 끄기)

[0147] 참가자들은 다이노 조건에서 구글 크롬 브라우저에 기본으로 설치되는 게임인 크롬 다이노를 플레이했다. 게임에서 공룡 캐릭터는 왼쪽에 배치된다(도 8 참조). 장애물은 오른쪽에서 왼쪽으로 오고 게임이 진행됨에 따라 더 빨라진다. 공룡이 장애물에 부딪히면 게임이 끝난다. 참가자가 버튼(공백 또는 위쪽 화살표)을 누르면 공룡이 점프하여 장애물을 피한다. 실험에서 참가자들에게 단순성을 위해 스페이스 키만 사용하여 플레이하도록 하였다.

[0148] Tabsonic 1 및 Tabsonic 2 조건에서 참가자들은 Neowiz에서 개발한 리듬 게임인 Tabsonic Bold를 플레이했다. 게임에서 플레이어는 대상 영역에 떨어지는 음표가 있을 때 해당 버튼을 눌러야 한다(도 8 참조). 플레이어가 특정 임계값보다 더 많은 음표를 놓치면 게임이 종료된다. 플레이어가 끝까지 살아 있으면 게임 점수가 제공된다. 음표가 날아갈 수 있는 6개의 레인이 있으며 각각 s, d, f, j, k 및 l 키에 매핑된다. 음표의 순서는 게임 디자이너가 배경 음악과 조화를 이루도록 설계되었다. 각 참가자는 실험자가 선택한 5곡("Jelly Fish", "Do It!", "Emoji Love", "This is You", "Iridescent")을 모두 연주했다. 그러나 참가자들은 Tabsonic 2 조건에서 배경 음악을 끈 상태에서 연주했다. 모든 참가자는 Dino 조건을 먼저 플레이한 후 Tabsonic 조건을 플레이했다. Tabsonic 1과 Tabsonic 2가 주어지는 순서와 각 조건 내에서 5곡이 주어지는 순서는 라틴 스캐어를 통해 균형을 이루었다. 통계 분석은  $\alpha$  수준 0.05로 반복 측정 ANOVA를 통해 수행되었다. 구형도 가정을 위반한 경우 Greenhouse-Geisser 보정을 적용했다.

[0149] 참가자들은 게임용 의자에 앉았고 모니터는 참가자의 눈높이에 설치되었다. 참가자들은 사전 설문지를 작성하고 동의서에 서명했다. 본 실험에 앞서 실험자는 참가자들에게 과제를 간략하게 소개하였다. 우리는 참가자들에게 가능한 한 높은 점수를 받기 위해 최선을 다할 것을 요청했다. 첫째, 참가자들은 10분 이상 다이노 조건을 하였다. 그런 다음 참가자들은 공식 Tabsonic 튜토리얼을 플레이했다. 다음으로, 참가자들은 Tabsonic의 각 조건에서 5곡을 연주했다. 조건 간의 전환은 실험자가 수동으로 수행했다. 재생 전후에 실험자는 화면 기록 및 입력



로깅을 실행하고 중지했다. 각 개인에 대한 실험은 약 60분이 소요되었다.

[0150] 모든 실험은 단일 Windows 데스크톱 컴퓨터(3.6GHz Intel Core i9-9900k, 32GB RAM, NVIDIA RTX2060)에서 수행되었다. 24인치 모니터(Dell)를 사용했으며, 해상도는 1680×1050 픽셀이었다. 일반적인 유선 물리적 키보드가 사용되었다. 참가자들은 Tapsonic 1 조건에서 소리를 듣기 위해 헤드폰(EPOS Sennheiser GSP 600)을 착용했다.

## [0151] 1.2 분석

[0152] 연구 1 실험에서 버튼 입력 로그(Dino의 경우  $b^{space}$ , Tabsonic 1 및 2의 경우  $b^s$ ,  $b^d$ ,  $b^f$ ,  $b^j$ ,  $b^k$ ,  $b^l$ ) 및 3가지 게임 조건에서 참가자의 화면 기록(v)을 얻는다. 이 연구의 목적은 다양한 버튼에 대한 반응성과 사전 대응성을 추정하는 것이 아니므로 각 Tabsonic 조건에 대한 6개의 독립적인 버튼 입력 로그를 하나로 집계했다 ( $b^{sdfjkl} = b^s \cup b^d \cup b^f \cup b^j \cup b^k \cup b^l$ ).

[0153] 먼저 실험자는 모든 비디오를 하나씩 시청하고 각 참가자가 경험한 게임 오버 수와 각 참가자가 얻은 점수를 기록했다. 그런 다음 본 발명에 제시된 단계들을 통해 반응성 및 선제성을 분석했다. 분석을 위한 Dino 조건의 축 소된 비디오는 53×33 해상도이고 분석을 위한 Tabsonic 조건의 비디오는 53×33 해상도였다.

[0154] 분석 결과, 각 게임 유형 조건에 대한 각 픽셀에 대해  $w_R$ ,  $w_P$ ,  $w_I$ ,  $\theta_R$  및  $\theta_P$ 을 얻었다. 이 값은 참가자의 버튼 누름과 해당 픽셀의 시각적 자극 간의 연관성을 나타낸다. 또한 이러한 값은 각 참가자에 대해 얻어진다.

[0155] 모든 분석은 단일 Windows 워크스테이션 컴퓨터(2.90GHz 64코어 AMD Ryzen Threadripper 3990X, 128GB RAM, NVIDIA Quadro RTX4000)에서 수행되었다. 참가자가 다이노 조건과 탭소닉 조건에서 10분 동안 재생한 데이터의 분석에는 약 10분 50분이 소요되었다. Dino 조건에서는 충분한 수의 시각적 이벤트가 관찰되지 않는 건너편 픽셀이 많기 때문에 분석 시간이 덜 소요되었다.

## [0156] 1.3 결과

[0157] 다이노 조건에서 참가자들은 평균 10.1( $\sigma=4.994$ ) 게임 오버를 경험했고 평균 1102.75( $\sigma=618.415$ ) 포인트를 기록했다. Tabsonic 1 조건과 Tabsonic 2 조건에서 참가자는 각각 평균 370,178( $\sigma=32,389$ ) 및 352,963( $\sigma=23,316$ )의 점수를 얻었다(5곡 모두 고려). 게임 유형이 게임 점수에 미치는 영향은 유의했다( $F_{2,38} = 2397.705$ ,  $p<0.001$ ,  $\eta_p^2 = 0.992$ ). 특히, Tabsonic 1과 Tabsonic 2의 차이는 post-hoc 분석에서 유의미한 것으로 나타났다( $p<0.01$ ).

[0158] 게임 유형이 평균  $w_I$ (모든 픽셀에 대해)에 미치는 영향은 통계적으로 유의했다( $F_{2,38} = 2397.705$ ,  $p<0.001$ ,  $\eta_p^2 = 0.992$ ). 평균  $w_I$ 는 Dino 조건의 경우 0.328( $\sigma=0.027$ ), Tabsonic 1 조건의 경우 0.768( $\sigma=0.023$ ), Tabsonic 2 조건의 경우 0.783( $\sigma=0.022$ )이었다. 사후 분석에서 Dino의  $w_I$ 는 Tabsonic 1 및 Tabsonic 2와 크게 다르다( $p=0.001$ ).

[0159] 게임 유형이 평균 반응성 R(모든 픽셀 및 버튼 유형에 대해)에 미치는 영향은 통계적으로 유의했다( $F_{2,38} = 61.493$ ,  $p<0.001$ ,  $\eta_p^2 = 0.765$ ). 평균 R의 평균은 Dino 조건의 경우 0.459( $\sigma=0.021$ ), Tabsonic 1 조건의 경우 0.589( $\sigma=0.050$ ), Tabsonic 2 조건의 경우 0.578( $\sigma=0.058$ )이었다. 사후 분석에서 Dino의 R은 Tabsonic 1 및 Tabsonic 2와 크게 다르다( $p=0.001$ ).

[0160] 다음으로 각 픽셀에 대해 얻은  $\theta_R$  매개변수를 분석했다. 특히 리액티브 버튼 입력에서 사용자의 성능을 설명하는 데 가장 중요한 값인  $\hat{\mu}_R$ 을 분석했다. 게임 유형이 평균  $\hat{\mu}_R$ (모든 픽셀에 대해)에 미치는 영향은 통계적으로 유의했다( $F_{2,38} = 54.148$ ,  $p<0.001$ ,  $\eta_p^2 = 0.740$ ). 평균  $\hat{\mu}_R$ 의 평균은 Dino 조건의 경우 0.357( $\sigma=0.022$ ), Tabsonic 1 조건의 경우 0.835( $\sigma=0.199$ ), Tabsonic 2 조건의 경우 0.847( $\sigma=0.204$ )이었다. 사후 분석에서 Dino의  $\hat{\mu}_R$ 는 Tabsonic 1 및 Tabsonic 2와 크게 다르다( $p<0.001$ ).

[0161] 픽셀에 대해 측정된 반응성  $R$ 과 적극성  $P$ 의 합은 항상 1이므로 독립변수가 적극성에 미치는 영향에 대한 추가 통계 분석을 수행할 필요가 없다(반응성 검정이면 충분함). 능동적인 히트맵도 1에서 반응성 값을 빼서 구하므로 따로 그럴 필요가 없다. 대신 각 픽셀에 대해 얻은  $\theta_P$  매개변수를 분석했다. 특히, 능동적인 버튼 입력에서 사용자의 성능을 설명하는 데 가장 중요한 매개변수인  $\sigma_P$ 를 분석했다. 게임 유형이 평균  $\sigma_P$ (모든 픽셀에 대해)에 미치는 영향은 통계적으로 유의했다( $F_{2,38} = 8.577, p < 0.01, \eta_p^2 = 0.333$ ). Dino, Tabsonic 1, Tabsonic 2의 평균  $\sigma_P$ 는 0.088( $\sigma = 0.004$ ), 0.101( $\sigma = 0.016$ ), 0.106( $\sigma = 0.019$ )이었다. 사후 분석에서 Dino의  $\sigma_P$ 는 Tabsonic 1 및 Tabsonic 2와 크게 다르다( $p < 0.01$ ).

[0162] 각 참가자의 인지 특성 매개변수  $\theta_R$  및  $\theta_P$ 가 참가자의 게임 점수를 얼마나 잘 설명하는지 선형 회귀 분석을 수행했다. 먼저 각 게임 유형 참가자 조합에서 모든 픽셀의  $\theta_R$  및  $\theta_P$ 를 평균화하였다. 그런 다음 scikit-learn 라이브러리를 사용하여 결과 5가지 매개변수( $\mu_R, \sigma_R, K_R, \mu_P, \sigma_P$ )와 참가자의 점수 간의 선형 회귀 분석을 수행하였다. 그 결과, 참가자들의 게임 점수는 Dino( $R^2 = 0.52$ ), Tabsonic 1( $R^2 = 0.70$ ), Tabsonic 2( $R^2 = 0.65$ )의 높은 결정 계수로 5가지 매개변수에서 예측되었다. 각 회귀 방정식과 플롯은 도 9에 나타난 바와 같다.

#### [0163] 1.4 토론

[0164] 실시간 게임을 할 때 플레이어는 다양한 방법으로 버튼을 누르도록 요청받는다. 예를 들어 Dino의 경우 플레이어는 버튼을 눌러 공룡이 움직이는 장애물에 부딪히는 것을 방지한다. Tabsonic의 경우 플레이어는 움직이는 표적이 특정 획득 영역 내에 위치할 때 버튼을 눌러야 한다. 그러나 결정적인 사건(장애물이 공룡에 닿거나 표적이 획득 영역에 있음)이 실제로 발생하는 순간에 플레이어가 계획을 세우고 실행하기 시작하면 너무 늦은 경우가 많다. 따라서 플레이어는 이벤트가 발생하기 전에 플레이어에게 제공되는 감각 신호에 의존하면서 버튼 누름을 트리거해야 한다. 연구 1에서는 화면의 픽셀에서 발생한 시각적 이벤트가 Dino 및 Tabsonic 조건에서 참가자의 버튼 입력을 어떻게 트리거했는지 알아내는 데 성공했다.

[0165] 먼저 다이노 상태의  $w_I$  히트맵을 보면(도 8 참조) 공룡이 상하로 움직이는 영역과 화면의 오른쪽 끝 영역에서 발생한 시각적 이벤트는 버튼 입력과 크게 관련이 없는 것으로 보인다. 이러한 결과를 얻은 이유는 다음과 같다. (1) 공룡의 점프 동작은 참가자가 버튼 누름을 계획하고 실행하도록 하는 감각 큐가 아니라 참가자의 버튼 누름에 의해 유발되는 이벤트이다. (2) 맨 오른쪽 영역에 장애물이 나타나면 공룡과 충돌하거나 버튼을 누를까 걱정할 정도로 긴급한 상황은 아니다.

[0166] 다음으로 다이노 조건에서 반응성(또는 능동적)의 히트맵을 보면 공룡 앞의 픽셀에서 발생한 시각적 이벤트는 참가자의 능동적인 버튼 입력을 유발하고, 조금 더 떨어진 픽셀에서 발생한 시각적 이벤트는 참가자의 반응 버튼 입력. 장애물이 공룡에 가까워질수록 참가자가 장애물의 접근에만 반응하면 공룡이 장애물에 부딪히는 것을 막을 수 없기 때문이다.

[0167] 기본적으로 Tabsonic과 Dino는 모두 움직이는 표적(또는 장애물)이 있는 게임이기 때문에 Tabsonic의  $w_I$  히트맵과 R 히트맵에서도 비슷한 경향이 관찰되었다. 목표물이 이동하는 라인 밖의 영역에서는  $w_I$ 의 값이 높았다. 획득 영역 바로 앞에서 발생한 시각적 이벤트는 참가자의 사전 버튼 입력을 유발하고, 조금 더 떨어진 영역에서 발생한 시각적 이벤트는 참가자의 반응 버튼 입력을 유발했다.

[0168] 반면 Tabsonic 1 조건과 달리 Tabsonic 2 조건은 배경 음악이 꺼졌다. 일반적으로 Tabsonic과 같은 리듬 게임에서 배경 음악의 비트는 목표물이 획득 영역에 도달하는 순간과 동기화된다. 따라서 배경 음악은 참가자가 버튼 입력 타이밍을 추정할 수 있는 추가 감각 신호를 제공한다. 큐 통합 이론에 따르면 추가 감각 큐는 항상 참가자의 버튼 누름 성능의 신뢰성을 향상시킨다. 실제로 Tabsonic 1 조건의 참가자 점수는 Tabsonic 2 조건 참가자의 점수보다 훨씬 높았다.  $\sigma_P$ 의 경우 본 발명의 기법에서 추정한 능동적 버튼 입력의 품질을 나타내는 분포 매개변수인 Tabsonic 2가 Tabsonic 1보다 높은 값을 얻었지만 그 차이는 통계적으로 유의하지 않았다.

[0169] 본 발명의 분석 기법에 의해 추정된 각 참가자에 대해 총 5개의 분포 매개변수( $K_R, \mu_R, \sigma_R, \mu_P$  및  $\sigma_P$ )가 있

다. 각 참가자의 분포 매개변수는 높은 결정 계수로 참가자의 게임 점수와 상관관계가 있었다(도 9 참조).

도 10은 연구 1 및 연구 2에서 일부 대표 픽셀에 대한 모델 피팅 결과를 보여주는 도면이다.

## 연구 2: 자기 확장 목표를 가리키기

연구 2에서 참가자는 스스로 크기가 커지는 대상을 클릭하고 가리키는 작업을 수행한다. 여기에서 참가자들은 트랙패드로 커서를 제어한다. 커서의 위치가 중앙이 되도록 비디오 프레임을 이동시킨 후 제안된 기법을 사용하여 참가자의 반응성과 주도성을 분석한다. 타겟 확장의 다양한 패턴을 부여하고, 참가자들의 반응성과 주도성, 각 버튼 입력의 품질이 어떻게 변했는지 탐구한다.

### 2.1 방법

12명의 참가자(남자 8명, 여자 4명)를 모집했으며, 이들의 평균 나이는 24.83세( $\sigma=3.24$ 세)였고, 이중 9명의 참가자는 매일 트랙패드를 사용한다고 보고했다.

연구 2에서 참가자들은 트랙패드로 커서를 제어하여 띠 모양의 표적을 클릭하도록 하였다. 참가자가 미리 정의된 시작 위치로 커서를 이동하고 일정 시간(0.5초) 동안 머무르면 대상이 일정 거리(900픽셀)에 나타난다. 대상의 초기 너비는 1픽셀이다. 대상이 나타날 때  $t=0$ 이라고 가정하면 대상이  $t=t_1$ 에서 폭 방향으로 확장되기 시작한다.  $t=t_2$ 까지 확장이 계속되다가 멈춘다. 참가자가 목표물 내부 또는 외부를 클릭하면 시도가 종료되고 목표물이 사라진다. 대상이 완전히 확장되면 너비는 50픽셀이다.

대상 확장이 종료되는 시간  $t_2$ 는 참가자마다 다르게 설정하였다. 좀더 구체적으로,  $t_2$ 는 참가자가 900픽셀 거리에서 50픽셀 너비로 0.8을 곱한(즉,  $t_2 = 0.8\bar{T}$ ) 일반 대상을 획득하는 데 걸리는 평균 시간( $\bar{T}$ )으로 설정하였다. 이를 위해 각 참가자에게 본 실험 전에 목표물(50픽셀 너비, 900픽셀 거리)을 최대한 빠르고 정확하게 60번 클릭하도록 요청했다. 본 실험의 과제와 마찬가지로 시작 위치에 커서가 0.5초 동안 머물렀을 때만 시도의 대상이 나타났다. 평균 시간( $\bar{T}$ )을 계산할 때 실패한 시도와 성공한 시도를 모두 고려했다.

주요 실험은 대상이 확장되는 패턴과 관련된 두 개의 독립 변수인 Easing 및 Duration이 있는  $3 \times 4$  개체 내 설계를 따랐다. 수준은 다음과 같다.

· Easing : In, Out, Pop

· Duration : 0.2, 0.4, 0.6, 0.8

여기서 Duration은  $t_1 = \text{Duration} \cdot \bar{T}$  과 같이 대상이 확장되기 시작하는 시간인  $t_1$ 을 결정한다. 즉, 대상이 확장되는 데 필요한 시간은  $(t_2 - t_1) = (0.8 - \text{Duration})\bar{T}$  이다. 완화는 대상이 확장되는 패턴을 나타낸다. 각 Easing 조건에 대해 시간 함수로서의 목표 너비  $W$ 는 다음과 같이 표현될 수 있다.

$$\text{For In condition, } W(t) = 1 + 49 \left( \frac{t - t_1}{t_2 - t_1} \right)^3 \text{ where } t \in [t_1, t_2]$$

$$\text{For Out condition, } W(t) = 1 + 49 \left( 1 - \left( \frac{t_2 - t}{t_2 - t_1} \right)^3 \right) \text{ where } t \in [t_1, t_2]$$

$$\text{For Pop condition, } W(t) = 50 \text{ where } t \in [t_1, t_2]$$

$t$ 가  $t_1$ 보다 작으면 대상의 너비는 모든 조건에서 1픽셀이고  $t$ 이  $t_2$ 보다 크면 대상의 너비는 모든 조건에서 50픽셀이다. 각 조건도 도 12에 시각화되어 있다.

실험 설계에 관계없이 모든 참가자에게 Baseline 조건이 주어졌다. Baseline 조건은 Easing 변수가 Pop으로 설정되고 Duration 변수가 0으로 설정됨을 의미한다. Baseline 조건의 경우 시간의 함수로 대상 너비  $W$ 는 다음과 같이 표현할 수 있다.

For In condition,  $W(t) = 1$  where  $t \in [t_1, t_2]$

[0185]

[0186]

Baseline 조건을 포함하여 각각의 고유한 Easing-Duration 조합에 대해 참가자는 60개의 표적 획득 시도를 수행하였다. 모든 작업 조합은 무작위 순서로 참가자에게 주어졌다. 통계 분석은  $\alpha$  수준 0.05로 반복 측정 ANOVA를 통해 수행되었다.

[0187]

모든 실험은 단일 Windows 데스크톱 컴퓨터(3.6GHz Intel Core i9-9900k, 32GB RAM, NVIDIA RTX2060)에서 수행되었다. 24인치 모니터(Dell)를 사용했으며, 해상도는  $1680 \times 1050$  픽셀이었다. Apple 트랙패드(Magic Trackpad)를 포인팅 장치로 사용하였다. 작업 응용 프로그램은 Python 언어(약 60Hz에서 실행)로 작성되었다.

[0188]

## 2.2 분석

[0189]

연구 2 실험에서 기준 조건을 포함하여 각 Easing-Duration 조합에 대한 참가자의 마우스 버튼 입력 로그 ( $b^{click}$ ), 화면 기록(v) 및 커서 좌표(x 및 y)를 얻었다. 전체 데이터 세트의 크기는 약 852MB였다.

[0190]

연구 1에서와 같이 본 발명에 설명된 단계를 따라 데이터 세트의 반응성과 선제성을 분석하였다. 축소된 화면 비디오는 픽셀 단위로  $55 \times 33$  해상도를 갖는다. 분석 결과, 각 Easing-Duration 조합의 각 픽셀에 대해  $w_R$ ,  $w_P$ ,  $w_I$ ,  $\theta_R$  및  $\theta_P$ 를 얻었다. 이 값은 참가자의 마우스 버튼 누르기와 해당 픽셀의 시각적 자극 간의 연관성을 나타낸다.

[0191]

## 2.3 결과

[0192]

작업 완료 시간은 대상이 나타난 시점과 버튼을 클릭한 시점의 시간 차이이다. 작업 완료 시간에 대한 완화의 효과는 통계적으로 유의했다( $F_{2,22} = 15.515$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.585$ ). In, Out, Pop의 시도 완료 시간 평균은 1.07( $\sigma = 0.119$ ), 1.005( $\sigma = 0.143$ ), 0.995( $\sigma = 0.148$ )였다. 작업 완료 시간에 대한 기간의 효과는 통계적으로 유의했다( $F_{3,33} = 20.690$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.653$ ). 0.2, 0.4, 0.6, 0.8에 대한 시도 완료 시간의 평균은 1.104( $\sigma = 0.138$ ), 1.021( $\sigma = 0.135$ ), 1.002( $\sigma = 0.143$ ), 0.969( $\sigma = 0.110$ )였다. 작업 완료 시간에 대한 Easing 및 Duration의 상호 작용 효과는 통계적으로 유의하지 않았다( $F_{6,66} = 1.135$ ,  $p = 0.347$ ,  $\eta_p^2 = 0.094$ ).

[0193]

오류율은 실패한 시도 횟수를 총 시도 횟수로 나눈 값이다. 오류율에 대한 완화의 효과는 통계적으로 유의하지 않았다( $F_{2,22} = 2.333$ ,  $p = 0.121$ ,  $\eta_p^2 = 0$ ). 또한 Duration이 오류율에 미치는 영향은 통계적으로 유의하지 않았다( $F_{3,33} = 2.112$ ,  $p = 0.118$ ,  $\eta_p^2 = 0.25$ ). 오류율에 대한 Easing 및 Duration의 상호 작용 효과는 통계적으로 유의하지 않았다( $F_{6,66} = 1.369$ ,  $p = 0.240$ ,  $\eta_p^2 = 0$ ).

[0194]

관련성 없음  $w_I$ 에 대한 Easing 효과는 통계적으로 유의했다( $F_{2,22} = 7.497$ ,  $p < 0.01$ ,  $\eta_p^2 = 0.406$ ). 평균 Irrelevance의 평균은 In 상태의 경우 0.134( $\sigma = 0.048$ ), Out 상태의 경우 0.103( $\sigma = 0.054$ ), Pop 상태의 경우 0.108( $\sigma = 0.050$ )이었다. Duration이  $w_I$ 에 미치는 영향은 통계적으로 유의했다( $F_{3,33} = 10.032$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.480$ ). 0.2, 0.4, 0.6, 0.8의 평균은 0.141( $\sigma = 0.054$ ), 0.122( $\sigma = 0.045$ ), 0.106( $\sigma = 0.054$ ), ( $\sigma = 0.054$ ), ( $\sigma = 0.49$ )  $w_I$  ( $F_{6,66} = 1.222$ ,  $p = 0.306$ ,  $\eta_p^2 = 0.099$ )에서 Easing과 Duration 사이에는 유의미한 상호작용 효과가 없었다.

[0195]

반응성 R에 대한 Easing의 효과는 통계적으로 유의했다( $F_{2,22} = 54.403$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.832$ ). In, Out, Pop의 평균 R은 0.690( $\sigma = 0.072$ ), 0.776( $\sigma = 0.096$ ), 0.802( $\sigma = 0.090$ )였다. Duration이 R에 미치는 영향은 통계적으로 유의했다( $F_{3,33} = 29.777$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.731$ ). 0.2, 0.4, 0.6, 0.8의 평균 R은 0.690( $\sigma = 0.071$ ), 0.715( $\sigma = 0.088$ ), 0.791( $\sigma = 0.082$ ), 0.829( $\sigma = 0.084$ )였다. R에 대한 완화 및 지속 시간의 상호 작용 효과는 통계적

으로 유의했다( $F_{6,66} = 3.373$ ,  $p < 0.01$ ,  $\eta_p^2 = 0.236$ ). Post-hoc 분석은 Duration이 0.4, 0.6, 0.8일 때 In과 Out 사이에 상당한 차이( $p < 0.01$ ), 모든 Duration 조건에서 In과 Pop 사이에 유의한 차이( $p < 0.05$ ), 기간은 0.6( $p < 0.05$ )이었다.

[0196] 다음으로 각 픽셀에 대해 얻은  $\theta_R$  매개변수를 분석했다. 특히 리액티브 버튼 입력에서 사용자의 성능을 설명하는 데 가장 중요한 값인  $\hat{\mu}_R$ 을 분석했다.  $\hat{\mu}_R$ 에 대한 Easing 효과는 통계적으로 유의했다( $F_{2,22} = 22.796$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.675$ ). In, 아웃, 팝의 평균  $\hat{\mu}_R$ 은 0.442( $\sigma = 0.099$ ), 0.5( $\sigma = 0.092$ ), 0.51( $\sigma = 0.105$ )이었다. Duration이  $\hat{\mu}_R$ 에 미치는 영향은 통계적으로 유의했다( $F_{1,585,17,433} = 14.569$ ,  $p < 0.001$ ,  $\eta_p^2 = 0.570$ ). 0.2, 0.4, 0.6, 0.8에 대한 평균  $\hat{\mu}_R$ 의 평균은 0.451( $\sigma = 0.117$ ), 0.450( $\sigma = 0.076$ ), 0.498( $\sigma = 0.095$ ), 0.0( $\sigma = 0.095$ ), 그리고 0.0  $\hat{\mu}_R$ 에 대한 완화 및 지속 시간의 상호 작용 효과는 통계적으로 유의했다( $F_{6,66} = 0.073$ ,  $p < 0.01$ ,  $\eta_p^2 = 0.236$ ). Post-hoc 분석은 Duration이 0.6과 0.8( $p < 0.001$ )일 때 In과 Pop 사이에 유의미한 차이가 있었고 Duration이 0.6과 0.8일 때( $p < 0.01$ ) In과 Out 사이에 유의한 차이가 있는 것으로 나타났다.

[0197] 픽셀에 대해 측정된 반응성 R과 적극성 P의 합은 항상 1이므로 독립변수가 적극성에 미치는 영향에 대한 추가 통계 분석을 수행할 필요가 없다(반응성 테스트로 충분함). 능동적인 히트맵도 1에서 반응성 값을 빼서 구하므로 따로 그럴 필요가 없다. 대신 각 픽셀에 대해 얻은  $\theta_P$  매개변수를 분석하였다. 특히, 사전 버튼 입력에서 사용자의 성능을 설명하는 데 가장 중요한 매개변수인  $\sigma_P$ 를 분석하였다.  $\sigma_P$ 에 대한 Easing의 효과는 통계적으로 유의하였다( $F_{2,22}$ ,  $p < 0.05$ ,  $\eta_p^2 = 0.25$ ). In, Out, Pop의 평균  $\sigma_P$ 는 0.043( $\sigma = 0.012$ ), 0.045( $\sigma = 0.014$ ), 0.038( $\sigma = 0.013$ )이었다. Duration이  $\sigma_P$ 에 미치는 영향은 통계적으로 유의하지 않았다( $F_{3,33} = 8.841$ ,  $p = 0.73$ ,  $\eta_p^2 = 0.5$ ). 0.2, 0.4, 0.6, 0.8의 평균  $\sigma_P$ 은 0.040( $\sigma = 0.009$ ), 0.043( $\sigma = 0.014$ ), 0.046( $\sigma = 0.013$ ), 0.013), 0.010이다.  $\sigma_P$ 에 대한 Easing 및 Duration의 상호 작용 효과는 통계적으로 유의하였다( $F_{6,66} = 3.348$ ,  $p < 0.05$ ,  $\eta_p^2 = 0.25$ ). Post-hoc 분석에서는 Duration이 0.8( $p < 0.05$ )일 때 In과 Pop 사이에 유의한 차이가 있었고 Duration이 0.6일 때( $p < 0.01$ ) In과 Out 사이에 유의한 차이가 있었고 Duration이 0.6일 때 Out과 Pop 사이에 유의한 차이가 있었다. 0.8( $p < 0.05$ ).

[0199] 본 발명에 따른 사용자 인터페이스 방법은 화면 영상과 입력 로그만을 분석하여 사용자 버튼 입력의 반응성과 선제성을 정량화할 수 있다. 여기서 반응성과 선제성은 컴퓨터의 주어진 출력에 따라 사용자가 버튼 입력을 얼마나 능동적으로 또는 능동적으로 계획하고 실행하는지를 의미할 수 있다. 본 발명은 사용자의 IOI(Input-to-Output) 분포가 다양한 버튼 입력 전략에 따라 고유하게 결정된다는 점을 고려하여 사용자의 반응성과 선제성 뿐만 아니라 각 전략에 대한 사용자의 실행 품질을 측정할 수 있게 한다.

[0201] 도 14는 본 발명에 따른 사용자 인터페이스 장치의 시스템 구성을 설명하는 도면이다.

[0202] 도 14를 참조하면, 사용자 인터페이스 장치(100)는 프로세서(1410), 메모리(1430), 사용자 입출력부(1450) 및 네트워크 입출력부(1470)를 포함할 수 있다.

[0203] 프로세서(1410)는 본 발명의 실시예에 따른 선제적 및 반응적 입력 수량화 기반의 사용자 인터페이스 프로시저를 실행할 수 있고, 이러한 과정에서 읽혀지거나 작성되는 메모리(1430)를 관리할 수 있으며, 메모리(1430)에 있는 휘발성 메모리와 비휘발성 메모리 간의 동기화 시간을 스케줄 할 수 있다. 프로세서(1410)는 사용자 인터페이스 장치(100)의 동작 전반을 제어할 수 있고, 메모리(1430), 사용자 입출력부(1450) 및 네트워크 입출력부(1470)와 전기적으로 연결되어 이들 간의 데이터 흐름을 제어할 수 있다. 프로세서(1410)는 사용자 인터페이스 장치(100)의 CPU(Central Processing Unit)로 구현될 수 있다.



- [0204] 메모리(1430)는 SSD(Solid State Disk) 또는 HDD(Hard Disk Drive)와 같은 비휘발성 메모리로 구현되어 사용자 인터페이스 장치(100)에 필요한 데이터 전반을 저장하는데 사용되는 보조기억장치를 포함할 수 있고, RAM(Random Access Memory)과 같은 휘발성 메모리로 구현된 주기억장치를 포함할 수 있다. 또한, 메모리(1430)는 전기적으로 연결된 프로세서(1410)에 의해 실행됨으로써 본 발명에 따른 사용자 인터페이스 방법을 실행하는 명령들의 집합을 저장할 수 있다.
- [0205] 사용자 입출력부(1450)은 사용자 입력을 수신하기 위한 환경 및 사용자에게 특정 정보를 출력하기 위한 환경을 포함하고, 예를 들어, 터치 패드, 터치 스크린, 화상 키보드 또는 포인팅 장치와 같은 어댑터를 포함하는 입력 장치 및 모니터 또는 터치 스크린과 같은 어댑터를 포함하는 출력장치를 포함할 수 있다. 일 실시예에서, 사용자 입출력부(1450)은 원격 접속을 통해 접속되는 컴퓨팅 장치에 해당할 수 있고, 그러한 경우, 사용자 인터페이스 장치(100)는 독립적인 서버로서 수행될 수 있다.
- [0206] 네트워크 입출력부(1470)은 네트워크를 통해 사용자 단말(1510)과 연결되기 위한 통신 환경을 제공하고, 예를 들어, LAN(Local Area Network), MAN(Metropolitan Area Network), WAN(Wide Area Network) 및 VAN(Value Added Network) 등의 통신을 위한 어댑터를 포함할 수 있다. 또한, 네트워크 입출력부(1470)은 데이터의 무선 전송을 위해 WiFi, 블루투스 등의 근거리 통신 기능이나 4G 이상의 무선 통신 기능을 제공하도록 구현될 수 있다.
- [0208] 도 15는 본 발명에 따른 사용자 인터페이스 시스템을 설명하는 도면이다.
- [0209] 도 15를 참조하면, 사용자 인터페이스 시스템(1500)은 사용자 단말(1510), 사용자 인터페이스 장치(100) 및 데이터베이스(1530)를 포함할 수 있다.
- [0210] 사용자 단말(1510)은 사용자에게 의해 운용되는 단말 장치에 해당할 수 있다. 본 발명의 실시예에서 사용자는 하나 이상의 사용자로 이해될 수 있으며, 복수의 사용자들은 하나 이상의 사용자 그룹으로 구분될 수 있다. 또한, 사용자 단말(1510)은 사용자 인터페이스 시스템(1500)을 구성하는 하나의 장치로서 사용자 인터페이스 장치(100)와 연동하여 동작하는 컴퓨팅 장치에 해당할 수 있다. 예를 들어, 사용자 단말(1510)은 사용자 인터페이스 장치(100)와 연결되어 동작 가능한 스마트폰, 노트북 또는 컴퓨터로 구현될 수 있으며, 반드시 이에 한정되지 않고, 태블릿 PC 등 포함하여 다양한 디바이스로도 구현될 수 있다. 또한, 사용자 단말(1510)은 사용자 인터페이스 장치(100)와 연동하기 위한 전용 프로그램 또는 어플리케이션(또는 앱, app)을 설치하여 실행할 수 있다.
- [0211] 사용자 인터페이스 장치(100)는 본 발명에 사용자 인터페이스 방법을 수행하는 컴퓨터 또는 프로그램에 해당하는 서버로 구현될 수 있다. 또한, 사용자 인터페이스 장치(100)는 사용자 단말(1510)과 유선 네트워크 또는 블루투스, WiFi, LTE 등과 같은 무선 네트워크로 연결될 수 있고, 네트워크를 통해 사용자 단말(1510)과 데이터를 송·수신할 수 있다.
- [0212] 또한, 사용자 인터페이스 장치(100)는 관련 동작을 수행하기 위하여 독립된 외부 시스템(도 1에 미도시함)과 연결되어 동작하도록 구현될 수 있다. 예를 들어, 사용자 인터페이스 장치(100)는 포털 시스템, SNS 시스템, 클라우드 시스템 등과 연동하여 다양한 서비스를 제공하도록 구현될 수 있다.
- [0213] 데이터베이스(1530)는 사용자 인터페이스 장치(100)의 동작 과정에서 필요한 다양한 정보들을 저장하는 저장장치에 해당할 수 있다. 예를 들어, 데이터베이스(1530)는 비디오에 관한 정보를 저장할 수 있고, 학습 데이터와 모델에 관한 정보를 저장할 수 있으며, 반드시 이에 한정되지 않고, 사용자 인터페이스 장치(100)가 본 발명에 따른 사용자 인터페이스 방법을 수행하는 과정에서 다양한 형태로 수집 또는 가공된 정보들을 저장할 수 있다.
- [0214] 또한, 도 15에서, 데이터베이스(1530)는 사용자 인터페이스 장치(100)와 독립적인 장치로서 도시되어 있으나, 반드시 이에 한정되지 않고, 논리적인 저장장치로서 사용자 인터페이스 장치(100)에 포함되어 구현될 수 있음은 물론이다.
- [0216] 상기에서는 본 발명의 바람직한 실시예를 참조하여 설명하였지만, 해당 기술 분야의 숙련된 당업자는 하기의 특허 청구의 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

## 부호의 설명

[0218]

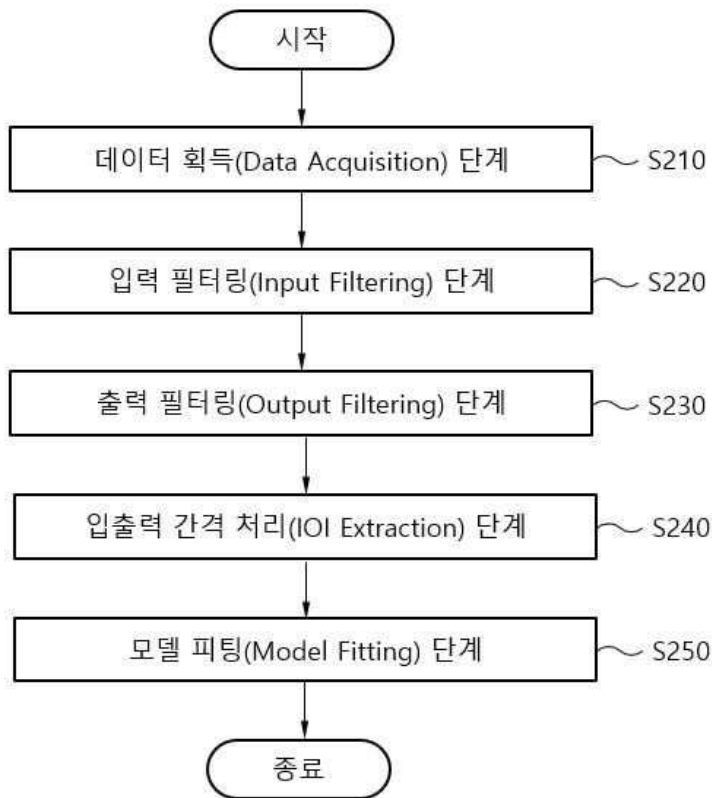
- 100: 사용자 인터페이스 장치  
 110: 데이터 획득부  
 120: 입력 필터링부  
 130: 출력 필터링부  
 140: 입출력 간격 처리부  
 150: 모델 피팅부  
 1500: 사용자 인터페이스 시스템

## 도면

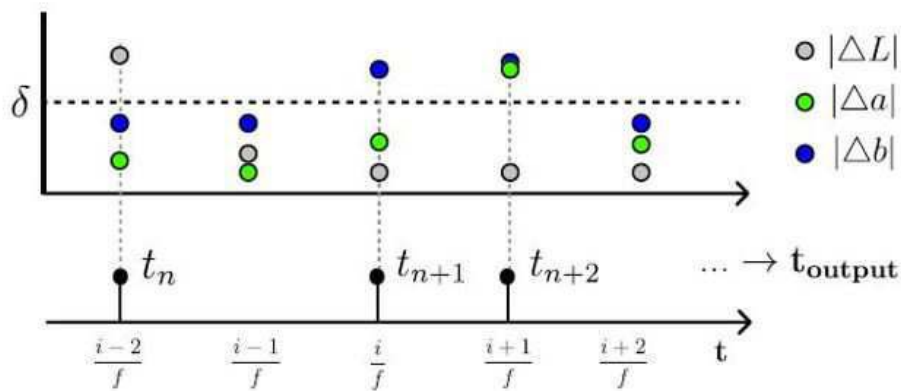
### 도면1



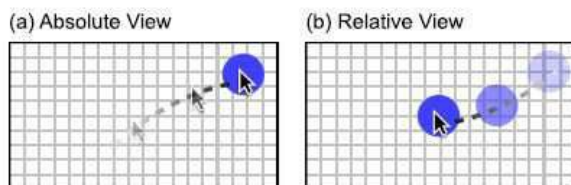
도면2



도면3

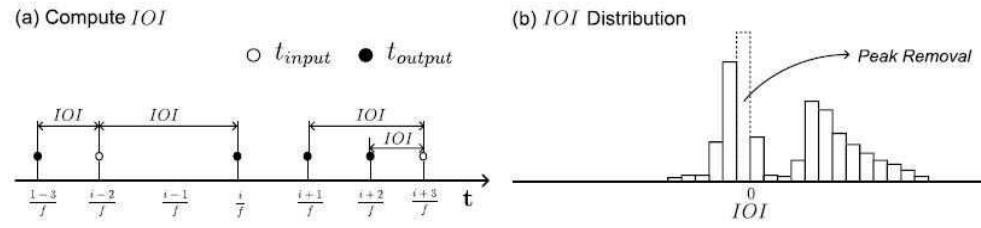


도면4

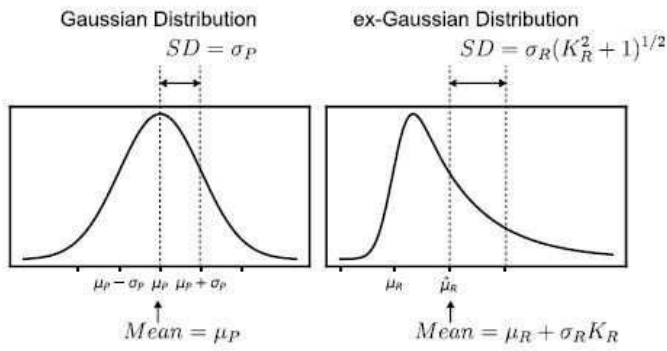




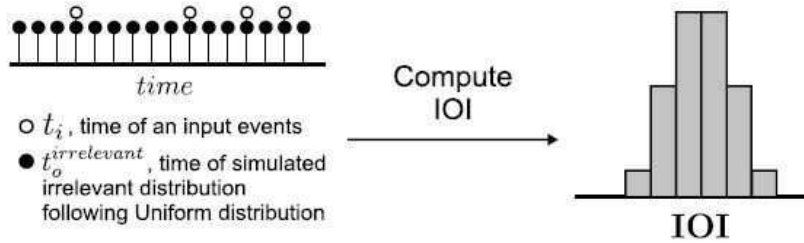
도면5



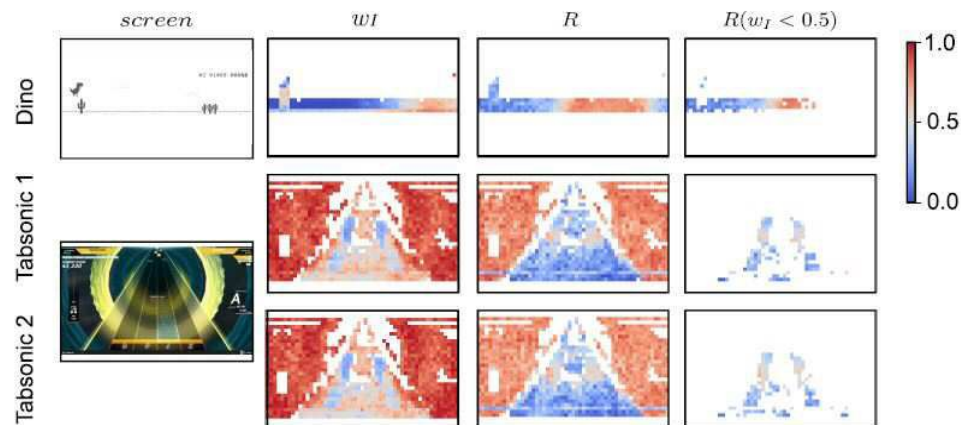
도면6



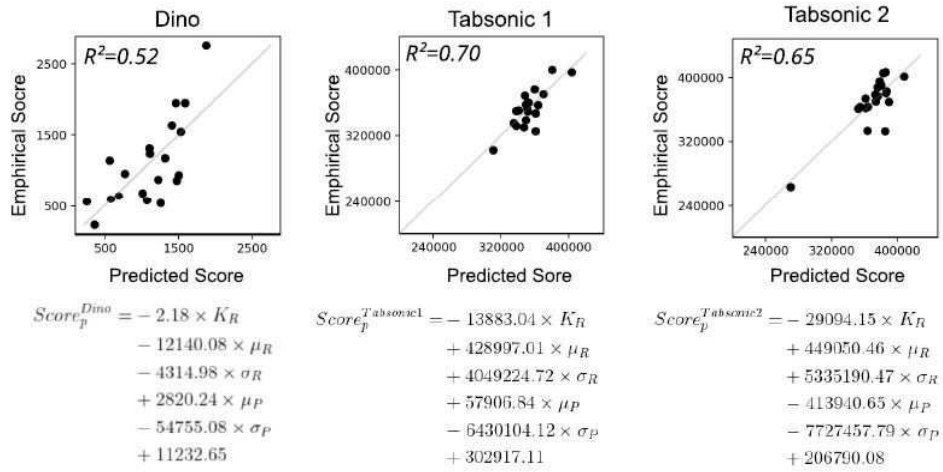
도면7



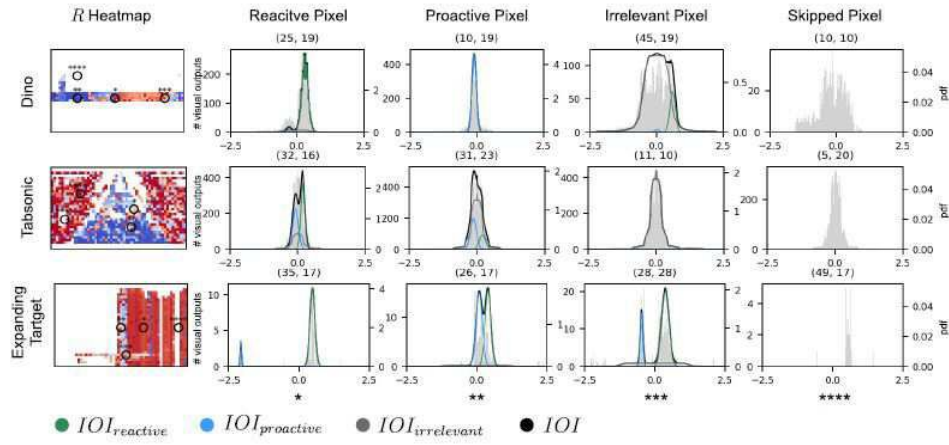
도면8



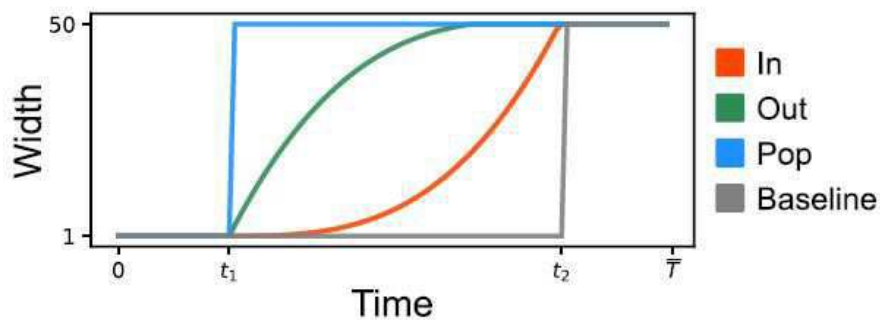
도면9



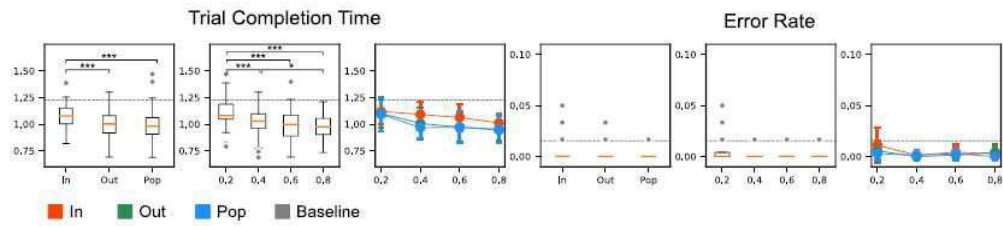
도면10



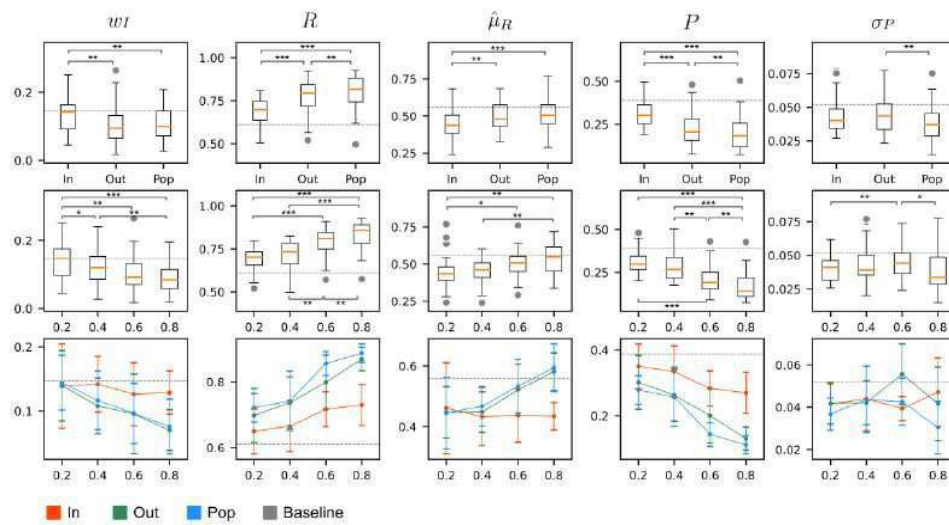
도면11



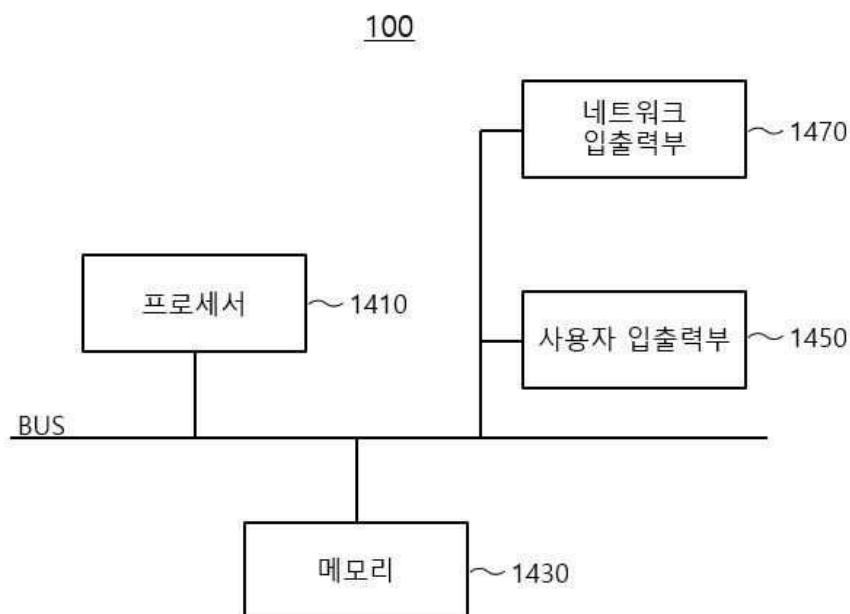
도면12



도면13



도면14



도면15

