얼굴 인증과 손-얼굴 혼성 제스처 인식을 이용한 사용자 인터페이스

김태희, 곽노윤
백석대학교 컴퓨터공학부
comiann113@naver.com, nykwak@bu.ac.kr

User Interface using Face Verification and Hand-face Hybrid Gesture Recognition

Taehi Kim, Noyoon Kwak
Division of Computer Engineering, Baekseok University

요 약

본 논문은 InsightFace 기반의 얼굴 인증과 MediaPipe 기반의 혼성 제스처 인식을 이용한 사용자 인터페이스에 관한 것이다. 우선, InsightFace를 활용하여 입력 프레임에 등장하는 사람의 얼굴을 인식한 후, 인식된 사람이 등록된 조작권자인지 인증하고, 혼성 제스처 기반의 사용자 인터페이스 제어 허용 여부를 판단한다. 등록된 조작권자일 경우, MediaPipe Face Mesh모델에서 선택한 7개의 랜드마크 좌표들과 Hands 모델의 21개의 랜드마크 좌표들을 이용해 얼굴 및 손 제스처를 인식한다. 그 후, 해당 사용자 이벤트를 발생시키고, 이에 대응되는 사용자 명령을 수행한다. 제안된 혼성 제스처 인터페이스는 얼굴 혹은 손 제스처 모드 전환 시, 별도의 제스처를 취할 필요가 없다는 것이 특징이다. 본 논문의 연구진이 실시한 사용성 평가 실험을 통해 얼굴 제스처와 손 제스처 각각 98.7% 및 96.4%의 평균 인식률을 얻을 수 있었다.

Ⅰ. 서 론

HCI(Human Computer Interaction) 기술은 센서 기술과 인공지능, 그리고 CPU, GPU, 메모리 등의 비약적 발전에 힘입어 사람의 오감뿐만 아니라 이를 초월한 수단을 통해 인간과 컴퓨터 사이의 자연스런 소통을 증진하고자 지속적 기술 혁신을 거듭해 오고 있다. 특히 사용자의 음성, 시선, 표정, 제스처, 터치 외에도 근전도, 심전도, 뇌파, 맥파, 눈전위도 등의 생체신호를 통해 디지털 기기를 조작하는 신개념의 NUI(Natural User Interface) 방식들도 속속 연구되고 있다. NUI 방식은 인간과 기계 사이의 접점을 없애고 직관적이고 자연스러운 사용자 경험을 제공할 뿐만 아니라 사용자의 인지적 혹은 신체적 능력이나 처한 상황이 반영된 상호작용을 통해 상황과 목적에 맞게 작동함으로써 원하는 작업을 사용자 친화적으로 처리함에 목적이 있다[1].

최근 들어 구글의 MediaPipe가 제스처 인식 분야에서 크로스 플랫폼 프레임워크의 중심으로 급부상하면서 본 논문의 연구진도 MediaPipe Hands 모델을 이용한 손 제스처 인터페이스[2]]와 MediaPipe Face Mesh 모델을 이용한 얼굴 제스처 인터페이스 기술[3]을 발표한 바 있었다. 또한 손 제스처 인터페이스와 얼굴 제스처 인터페이스의 각각의 장단점이 존재함에 따라 손 제스처와 얼굴 제스처를 혼용한 혼성 제스처 기반의 사용자 인터페이스[4]를 제안함으로써 사용자의 일상적 사용 경험을 크게 방해하지 않으면서도 자유롭고 편리하게 사용자 의도를 전달할 수 있었다. 하지만 동일한 장비 혹은 컴퓨터를 다수의 작업자들이 공동으로 관리하는 산업 환경이나 생활환경들이 많이 존재한다. 이러한 상황에서 보안성과 편의성을 제공하기 위해서는 다수의 사용자들 중 조작권자를 식별해 등록된 조작권자에게만 혼성 제스처 인터페이스의 제어 권한을 허용하는 선별적 수단이 필요하다. 이러한 필요성에 따라 본 논문은 InsightFace를 이용해 사용자의 얼굴을 인식하고, 인식한 얼굴이 적정 조작권자인지를 식별해 인증된 조작권자에게만 혼성 제스처 기반의 사용자 인터페이스를 제공하는 시스템을 제안하고자 한다. 본 논문에서는 InsightFace를 이용한 얼굴 인증 부문에 대해 선술하고 MediaPipe에 기반한 손-얼굴 혼성 제스처 인식을 이용한 사용자 인터페이스 부문을 후술할 것이다.

Ⅱ. 본론

2.1 얼굴 인증 및 혼성 제스처 인식 과정

본 논문은 입력 프레임에서 2D와 3D 심층 얼굴 분석을 위한 통합 라이브러리인 InsightFace[5]를 이용해 얼굴을 인식한 후, 등록된 조작권자인지 인증하고, 구글의 MediaPipe 프레임워크[6] 중 Face Mesh모델과 Hands 모델을 이용해 조작권자의 얼굴과 손 제스처를 혼성형으로 인식한다. 그림 1은 제안된 혼성 제스처 기반의 사용자 인터페이스의 전체적인 순서도를 나타낸 것이다. 제안된 방법은 크게 얼굴 인증 과정과 혼성 제스처 인식 과정으로 구성된다.

첫째, 얼굴 인증 과정에서는 일련의 입력 영상 프레임들에서 다수의 사용자들 중 조작 권한이 부여된 사용자의 얼굴 특징을 식별한 후, 이를 통해 얼굴 인증된 조작권자를 선별한다. 이를 위해, 먼저 정면, 측면, 기울어진 얼굴 구도를 포함하는 일련의 인물 영상 프레임들을 일정 프레임 간격으로 샘플링(sampling)해 얼굴 인식 데이터셋을 생성한다. 이후 얼굴 인식 데이터셋으로부터 탐지한 얼굴 영역에서 512차원의 임베딩 벡터를 추출해 정규화한 후, 코사인 유사도를 이용해 얼굴을 분류하는 지도학습 과정을 통해 얼굴 인식 모델을 생성한다. 얼굴 인식 데이터셋은 30fps(frame per second)의 8초 분량의 인물 동영상 2개를 5개 프레임 간격으로 프레임 데이터를 저장하여 한 사람당 100개의 데이터를 가지도록 생성한다. 얼굴 구도는 정면, 측면, 기울어진 얼굴 총 3가지 구도를 사용한다. 얼굴 인식 학습 시, 얼굴 인식 데이터셋을 바로 입력하지 않고, InsightFace의 RetinaFace[7]을 이용해 얼굴 영역을 탐지한 후, InsightFace의 Arcface[8]을 이용해 임베딩(embedding)을 수행한다. 임베딩한 결과로 512차원의 얼굴 특징 데이터가 만들어지는데, 이를 넘파이(Numpy)의 npy 파일로 저장한다. Arcface는 DCNN(Deep Convolutional Neural Network) 임베딩 방식을 이용해 임베딩 특징을 추출한다. 여기서 추출한 임베딩 특징과 가중치를 정규화한 후, 이 두 개의 값을 내적하여 코사인 유사도를 얻고, 이 값으로 얼굴 부류(face class)를 분류한다. 임베딩 데이터는 (n, 512) 차원으로 만들어지는데, 여기서 n은 데이터의 개수를 의미한다. 다음으로, 저장된 npy 파일을 입력받고 Keras를 통해 학습한 후, 얼굴을 인식하는 모델을 생성한다. 생성한 얼굴 인식 모델은 onnx 파일로 변환하여 저장한다.

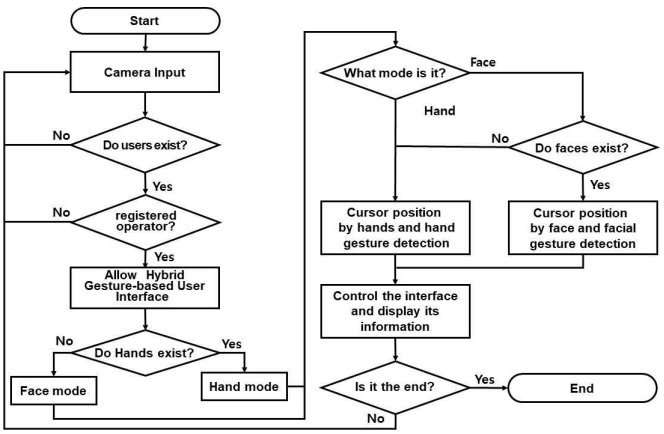


그림 1. 제안된 혼성 제스처 기반의 사용자 인터페이스의 전체적인 순서도

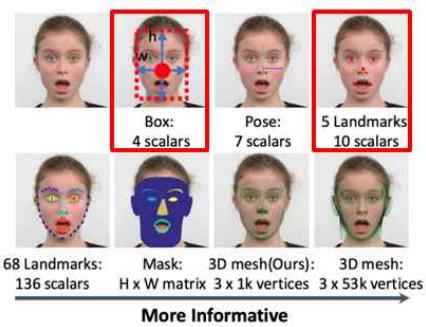


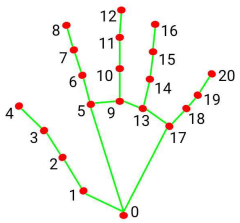
그림 2. RetinaFace 얼굴 인식 정보

이후, 얼굴 인식 과정에서는 일련의 입력 영상 프레임들에서 얼굴 영역을 탐지한 후, 얼굴 인식 모델에 입력하면 얼굴 인식을 수행해 해당 얼굴 부류와 인식 신뢰도를 출력한다.

제안된 방법의 얼굴 데이터 학습과 얼굴 인식 과정은 InsightFace의 사전 학습된 buffalo_l 모델을 사용하여 얼굴 인식을 진행한다. buffalo_l 모델에 5개의 onnx 파일이 있는데, 그 중 RetinaFace를 이용해 얼굴 검출(face detection)을 수행하고, ArcFace를 이용해 얼굴 인식(face recognition)은 수행한다. 이때 RetinaFace는 그림 2의 Box와 5 landmarks를 통해 사람의 얼굴 영역과 얼굴 랜드마크들의 위치를 찾는다. 이렇게 찾은 얼굴 영역을 Arcface를 통해 512차원의 임베딩 벡터로 변환한다. 그리고 얼굴 인식 및 조작권자 인증 과정에서는 인식된 얼굴이 누구인지 알기 위해 onnx 파일을 이용해 얼굴 인식을 수행한 후, 그렇게 인식된 얼굴이 등록된 조작권자이고 그 인식 신뢰도(혹은 정확도)가 기설정된 인증 임계값(적정 수준 0.95 혹은 엄격한 수준 0.97) 이상일 때 최종적으로, 해당 얼굴을 장비 조작 권한을 갖는 등록된 조작권자의 얼굴인 것으로 인증한다. 참고로 InsightFace 기반의 얼굴 인증은 선행 연구[3]에서는 초당 프레임 수(fps)의 1/5 간격 단위로 얼굴 인증을 수행하지만, 본 논문에서 초당 프레임 수(fps)의 1/6 간격 단위로 수행하여 연산량 부담을 줄인다. 제안된 얼굴 인증 과정에서는 얼굴 영역을 탐지한 후, 얼굴 인식을 수행해 기설정된 인식 신뢰도 이상인 사용자를 조작권자로 인증하되, 입력되는 영상 내에 복수의 사람이 존재하는 경우, 어느 한 사람이 인식된 직후엔 그 인식된 사람을 바로 화면 속에서 지우거나 무시한 상태에서 다른 사람들을 다시 인식하는 과정을 반복수행한다. 이후, 얼굴 인증 과정에서 선별된 조작권자의 얼굴 및 손의 혼성 제스처 인식을 허용하면, 혼성 제스처 인식 과정에서는 손 제스처 인식[2]과 얼굴 제스처 인식[4] 결과에 따른 사용자 이벤트를 발생시켜 그에 대응하는 사용자 명령을 실행하도록 인터페이스를 제어하고, 필요시 혼성 제스처 인식 결과를 조작권자와 상호작용하도록 화면에 표시한다.

2.2 MediaPipe Hands 기반의 손 제스처 인터페이스

제안된 MediaPipe Hands 기반의 손 제스처 인터페이스[2]의 손 제스처 인식 과정에서는 그림 3의 21개의 손 모델 랜드마크들(landmarks of hands model)을 이용해 손가락 펼침과 굽힘 여부를 탐지하는 손 제스처 인식 시스템에서, 카메라의 각 입력 프레임에서 검출한 상기 손 모델 랜드마크들을 이용해 손가락들의 맞닿음과 벌림, 그리고 각 손가락의 펼침과 굽힘의 조합으로 정의한 손 다이얼 자세를 탐지하면, 손 모델 랜드마크들 중 두 개의 좌표들 또는 손 모델 랜드마크들의 조합으로 구한 두 개의 좌표를 이용해 상기 손 다이얼 자세의 각도를 산출해 저장하고 필요시 표출한다. 그리고 손 다이얼 제스처 인식 단계에서는 기설정된 프레임 개수로 구성된 연속 프레임 리스트의 특정 구간에서 상기 손 다이얼 자세를 포함한 프레임의 수가 기설정된 손 다이얼 자세 점유비 이상인 조건을 만족하는 순간 그 프레임의 손 다이얼 자세의 각도를 시작 각도로 설정한 후, 손의 움직임에 따라 변하는 상기 손 다이얼 자세를 포함하는 각 프레임의 현재 각도와 시작 각도 간의 각도 격차를 기설정된 각도 대 눈금 단위로 변환해 다이얼 눈금의 증감량으로 인식한다.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

그림 3. 제안된 손 제스처 인식에서 사용하는 손 랜드마크들

2.3 MediaPipe Face Mesh 기반의 얼굴 제스처 인터페이스

얼굴 제스처 인식의 경우 앞서 소개한 MediaPipe Face Mesh의 3D 얼굴 랜드마크 좌표들 중 그림 4의 빨간 점으로 표시한 총 7개의 랜드마크 좌표들을 이용해 Pan 각도, Tilt 각도, Roll 각도의 3D 얼굴 각도, 양안 개폐 상태 및 유지시간 등의 조합으로 구성된 얼굴 제스처를 인식한다.

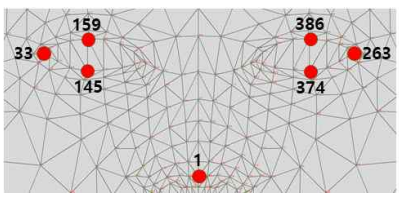


그림 4. 제안된 얼굴 제스처 인식에서 사용하는 7개의 랜드마크들

본 논문의 얼굴 제스처 인터페이스는 우선, 얼굴 제스처 추정 과정에서는 일련의 카메라 입력 프레임들에서 탐지한 얼굴 영역에서 얼굴의 Pan 각도와 Tilt 각도를 추정한 후, 각각의 얼굴 각도에 대응하는 화면상 커서 위치의 수평 좌표와 수직 좌표를 산출한다. 다음으로 얼굴 영역에서 추정한 얼굴의 Roll 각도와 양안 개폐 상태 및 그 유지 시간 등으로 구성된 얼굴 정보에서 얼굴 제스처를 인식해 이 커서 위치와 연관시켜 사용자 이벤트를 발생시키고 이 사용자 이벤트에 대응하는 명령을 실행하도록 인터페이스를 제어한다. 이때, 제안된 얼굴 인터페이스에서는 화면상 커서의 수평 좌표들과 수직 좌표들에 커서 이동속도에 따른 적응형 이동 평균 처리를 적용해 커서의 미세 떨림을 안정화시킨다. 또한 양안 동시 개폐 상태를 판별할 시, 양안 동시 개폐의 불연속성을 반복적으로 검사해 특정 조건에서 양안의 일시적 개폐 불일치 상태를 개방과 폐쇄 중 어느 한 상태로 통일되도록 수정한다. 예컨대, 양안 폐쇄 상태에서 양안 개방 상태로의 전환 시에 오작동이 더 자주 발생하기에 양안 개폐 인식 시, 양안이 모두 개방된 인식 시점을 기준으로, 최근 두 프레임을 검사해 양안이 모두 개방된 현재 프레임(t 프레임)과 양안이 모두 폐쇄된 전전 프레임($t-2$ 프레임) 사이에서 직전 프레임($t-1$ 프레임)이 한쪽 안구만 개방된 것으로 인식되면, 양안 모두가 폐쇄된 것으로, 강제로 수정해 인식하는 과정을 반복적으로 수행한다. 그 결과, 연산 부담을 상쇄할 만큼 양안 개폐 제스처의 인식률이 크게 상승하는 것을 확인할 수 있었다. 제안된 혼성 제스처 인터페이스의 얼굴 제스처와 그에 대응하는 사용자 이벤트 종류는 표 1과 같다.

표 1. 제안된 혼성 제스처 인터페이스의 얼굴 제스처와 그에 대응하는 사용자 이벤트 종류

User's facial gestures	Operating conditions and holding times	User Events
After keeping both eyes closed for a certain period of time, they open.	More than 0.2 seconds but less than 3 seconds	Left click
Perform left-click gesture twice within a certain period of time.	Successive trial gap of less than 2 seconds	Double click (left)
After keeping one eye closed for a certain period of time, it opens.	More than 0.2 seconds but less than 3 seconds	Right click
Move the cursor after maintaining the right-click gesture for a certain period of time.	Exceed 1 second	Drag
Both eyes open during drag event.	-	Drop
Roll head to the right	Critical roll angle: -35°	Scroll up
Roll head to the left	Critical roll angle: +35°	Scroll down
After keeping both eyes closed for a certain period of time, they open.	Exceed 3 seconds	Enable/Disable interface

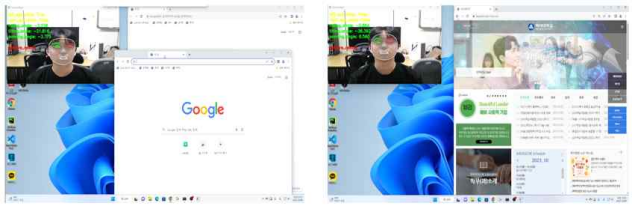
제안된 혼성 인터페이스의 손 제스처와 그에 대응하는 사용자 이벤트 종류는 표 2와 같다. 손 제스처 인식을 위해 사용되는 정보는 손의 모양 및 유지 시간, 위치, 각도이다.

표 2. 제안된 혼성 인터페이스의 손 제스처와 그에 대응하는 사용자 이벤트 종류[4]

User's hand gestures	Operating conditions /holding time	User Events
Current frame hand shape (e), Previous frame hand shape (d)	-	Left click
Perform left-click gesture twice within a certain period of time.	Successive trial gap of less than 2 seconds	Double click (left)
Current frame hand shape (e), Previous frame hand shape (c)	-	Right click
Move the cursor after maintaining the right-click gesture for a certain period of time.	Exceeds 0.5 seconds	Drag
During a darg event Current frame hand shape (e), Previous frame hand shape (d)	-	Drop
Move in one of the directions (up, down, left, right) with the hand shape in the form of (f)	-	Scrall up/down left/right
Hand shape in the form of (g) Decrease when rotating clockwise/ Increase when rotating counter-clockwise	-	Volume up/down
After keeping hand shape (a) for a certain period of time, hand shape (b)	Exceeds 0.1 seconds	Enable/Disable interface

Ⅲ. 시뮬레이션 결과 및 고찰

본 논문의 연구진은 사용자의 컴퓨터 환경에서 사용성을 검증하고 활용 사례를 보이기 위해 웹 서핑, 게임, 비디오 재생, 문서 열람 등 다양한 사용자 시나리오에 대한 시뮬레이션을 수행하였다. 그림 5(a)는 사용자가 웹 브라우저의 탭을 옮기는 상황으로, 마우스 드래그 중인 장면 예시이고, 그림 5(b)는 뒤로 가기 버튼을 누르는 상황으로 좌클릭 중인 장면 예시이다.



(a) (b)

그림 5. 제안된 혼성 제스처 인터페이스를 이용한 사용자 시나리오 시뮬레이션

표 3. 제안된 얼굴 제스처 인식의 사용성 평가 결과

User	Left Click	Double Click	Right Click	Drag	Drop	Scroll Up	Scroll Down	Enable/ Disable	Total
1	30	30	29	29	30	30	30	30	238
2	30	30	26	28	30	30	30	28	232
3	30	30	30	30	30	30	30	28	238
4	30	30	30	30	30	30	30	26	236
5	30	30	27	30	30	30	30	28	237
6	30	30	29	30	30	30	30	28	237
7	30	30	30	30	30	30	30	30	240
8	30	29	30	30	30	30	30	26	235
8	30	30	30	29	30	30	30	29	238
10	30	30	30	30	30	30	30	28	238
Avg (%)	**100**	**99.6**	**97.0**	**98.6**	**100**	**100**	**100**	**93.6**	**98.7**

표 4. 제안된 손 제스처 인식의 사용성 평가 결과

User	up	down	left	right	left click	right click	dial plus	dial minus	Total
A	30	30	27	29	30	29	29	30	234
B	30	30	29	29	29	30	30	30	237
C	28	30	30	29	30	30	30	29	236
D	29	29	25	30	24	24	30	28	219
E	30	28	26	27	28	30	30	30	229
F	30	29	24	30	30	29	30	28	230
G	28	26	25	28	30	30	30	30	227
H	30	30	30	26	30	29	30	30	235
I	30	30	30	26	30	28	30	30	234
J	30	30	29	29	28	28	30	30	234
Avg (%)	**98.3**	**97.3**	**91.6**	**94.3**	**96.3**	**95.6**	**99.6**	**98.3**	**96.4**

앞서 소개한 사용자 시나리오 시뮬레이션 외에도 얼굴 제스처와 손 제스처의 인식률을 측정하기 위해 10명의 실험자들을 대상으로 각 제스처를 취하였다. 표 3은 얼굴 제스처 인식의 사용성 평가를, 표 4는 손 제스처 인식의 사용성 평가를 나타낸 것이다. 사용성 평가 결과에 따르면, 얼굴 제스처와 손 제스처의 평균 인식률은 각각 98.7% 및 96.4%임을 확인할 수 있다.

Ⅳ. 결 론

본 논문에서는 InsightFace를 이용해 조작권자를 식별하고, MediaPipe의 Face Mesh 모델과 Hands 모델을 이용해 혼성 제스처 기반의 사용자 인터페이스를 구현하였으며, 사용자 이용 시나리오에 따른 시뮬레이션과 각 제스처 인터페이스의 사용성 평가 실험을 진행하였다.

제안된 얼굴 인증과 혼성 제스처 기반의 인터페이스는 얼굴 인증 모델로 입력 프레임에 등장한 얼굴들을 인식하고, 등록된 조작권자를 탐지한다. 만일 등록된 조작권자를 찾았을 경우, MdeiaPipe Face Mesh 모델과 Hands 모델을 이용해 입력 프레임으로부터 사용자의 얼굴과 손에 대한 3D 랜드마크 좌표를 추정한다. 그 후 Pan, Tilt, Roll의 3D 얼굴 각도와 양안의 개폐 여부를 이용해 얼굴 제스처를 판단한다. 또한 손가락의 접힘 여부와 중지와 검지가 이루는 각도, 손 모양, 위치, 각도의 변화 양상을 이용해 손 제스처를 판단한다. 판단한 각 제스처에 따라 대응되는 마우스 커서 이동 및 인터페이스 이벤트를 제어한다.

눈을 깜빡이거나 얼굴의 여러 각도를 조절하는 등 간결한 행동으로 마우스 커서 및 다양한 이벤트를 제어할 수 있다는 점에서 사용자가 사용법을 쉽게 익혀 편리하게 사용할 수 있을 것으로 본다. 또한, 손 제스처 기반의 인터페이스와 얼굴 제스처 기반의 인터페이스 간의 자유로운 전환이 가능하다는 점에서 각 제스처의 장점을 적극적으로 활용하고, 손이 불편할 경우 얼굴 제스처를 사용하는 등 다양한 상황에서 이를 유연하게 제어할 수 있을 것으로 기대한다. 그러나 입력 프레임에 등록된 조작권자를 포함한 다수의 사람이 등장할 때, 등록된 조작권자 외에 다른 사람도 혼성 제스처 기반의 인터페이스를 제어하는 등 보안성 측면에서 다소의 허점이 있다. 또한, 등록된 조작권자가 제안된 혼성 제스처 기반의 사용자 인터페이스를 제어하는 도중 다른 사람의 팔이 입력 프레임에 함께 등장할 경우, 해당 사람의 팔이 등록된 조작권자의 팔로 인식되는 오류 사항도 발견할 수 있었다. 따라서 본 논문의 연구진은 MediaPipe Holistic 모델을 활용하여 입력 프레임에 다수의 사람 인식 개체들을 저장하고, 등록된 조작권자만 혼성 제스처 기반의 사용자 인터페이스를 사용할 수 있도록 제한하는 방안을 연구할 계획이다.

ACKNOWLEDGMENT

본 논문은 2023년도 교육부의 재원으로 한국연구재단의 지원을 받아 수행된 지자체-대학 협력기반 지역혁신 사업의 결과이다.(2021RIS-004)

참 고 문 헌

[1] 김고운, “Natural User Interface 표준화 동향 연구”, 한국통신학회 하계종합학술발표회 논문집, pp. 750-752, 2020.

[2] 송락빈, 홍윤아, 곽노윤, “MediaPipe Hands 모델 기반의 손 제스처 인식을 이용한 사용자 인터페이스”, 멀티미디어학회논문지, 제26권, 제2호, pp. 101-113, 2023. 2.

[3] 목진왕, 곽노윤, “MediaPipe Face Mesh를 이용한 얼굴 제스처 기반의 사용자 인터페이스의 성능 개선”, 사물인터넷융복합논문지, 제9권, 제6호, pp. 125-134, 2023. 12.

[4] 박지홍, 목진왕, 곽노윤, “MediaPipe를 이용한 혼성 제스처 기반의 사용자 인터페이스”, 2023년도 한국디지털콘텐츠학회 추계종합학술대회 논문집, pp. 34-40, 2023. 11.

[5] InsightFace: 2D and 3D Face Analysis Project, Retrieved Jan. 9, 2024. from https://github.com/deepinsight/InsightFace

[6] Google MediaPipe, Retrieved Jan. 9, 2024. from https://mediapipe.dev.

[7] J. Deng, J. Guo, E. Ververas, I. Kotsia, and S.Zafeiriou, “RetinaFace: Single-Shot Multi-Level Face Localisation in the Wild,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5203-5212, 2020.

[8] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “ArcFace: Additive Angular Margin Loss for Deep Face Recognition,” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4690-4699, 2020.