Journal of Korea Robotics Society (2024) 19(4):335-346
https://doi.org/10.7746/jkros.2024.19.4.335 ISSN: 1975-6291 / eISSN: 2287-3961

표정 인지 및 모션 생성 모델을 활용하여 사람과 상호작용하는 로봇 시스템

A Robot System that Interacts with Human using Facial Expression Recognition and Motion Generation Model

송 호 준[1] · 한 지 형†
Ho-Joon Song[1], Ji-Hyeong Han†

Abstract: When people empathize, nonverbal communication such as facial expressions and gestures is more weighted than language. Therefore, in the human-robot interaction (HRI), it is important for robots to recognize and understand facial expression of humans. This is because it helps emotional communication and alleviates the psychological distance between robots and humans. In this paper, we introduce the robot system using facial expression recognition and motion generation models for interaction with human. It recognizes human emotions using POSTER++ based on Vision Transformer (ViT) and generates motions using FLAME that is developed to utilize diffusion model. In our scenario, we first perceive facial expression by POSTER++ and generate appropriate reaction texts using text generation model, Gemini AI. Then, we input the generated reaction text into FLAME to obtain 3D motion data, and the data is used to get arm joint angles that is calculated for Forward Kinematics (FK) and Inverse Kinematics (IK). Focusing on scenarios involving human interaction, we only use the head and arm motors of the humanoid robot, HUMIC. Furthermore, we simulate scenarios to verify that our system can recognize facial expression and show various reaction motions.

Keywords: Robot System, Human-Robot Interaction, Facial Expression Recognition, Motion Generation

1. 서 론

인간-로봇 상호작용 분야(HRI, Human-Robot Interaction)에서, 최근 로봇이 사람 가까이에서 맞춤형 역할 수행을 할 수 있도록 요구하게 되면서 관련 연구가 활발히 진행되고 있다. 특히 소셜 로봇, 정신 건강 모니터링 등 인공지능 기술을 활용하여 사람과 상호작용하는 로봇 기술이 개발되고 있으며, 이에 더하여 로봇과 사람이 사회적 교감을 하는 수준까지 인식이 확장되고 있다[1].

사람은 감정 교류를 할 때 언어로 표현하는 것보다 표정, 몸짓 등 비언어 의사소통의 비중이 높다. 그러므로 로봇이 인지적, 정서적 교류를 할 수 있도록 사람의 감정을 인지하고 이해하는 것은 중요하다. 감정적 교류는 로봇과 사람의 심리적 거리를 완화하며, 친밀한 관계를 형성할 수 있도록 하기 때문이다[2]. 이에 본 논문에서는 사람의 표정을 통해 감정을 인지하고 이에 알맞은 반응을 행동하는 것으로 사람과 상호작용하는 로봇 시스템을 제안한다. 표정 인지와 모션 생성을 위한 2가지 인공지능 기술을 사용하며, 이를 유기적으로 연결하여 감정에 따른 반응 모션을 생성하고, 휴머노이드 로봇 HUMIC을 통해 구현한다.

사람의 표정을 보고 감정을 인지하여 분류하기 위한 연구는 오래전부터 많은 관심을 가지고 꾸준히 진행되었다. 예전에는 이미지 분류 기술로 합성곱 신경망(CNN, Convolutional Neural

Received : Jul. 31. 2024; Accepted : Aug. 5. 2024
※ This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2024-RS-2022-00156295) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).
1. MS Student, Dept. of Computer Science and Engineering, Seoul National University of Science and Technology, Seoul, Korea (ghwns0703@seoultech.ac.kr)
† Associate Professor, Corresponding author: Dept. of Computer Science and Engineering, Seoul National University of Science and Technology, Seoul, Korea (jhhan@seoultech.ac.kr)

Network)을 많이 사용했으나 최근 Vision Transformer (ViT)[3]를 통한 연구가 급격히 증가하고 있다. TransFER[4], POSTER[5], POSTER++[6], S2D[7] 등 꾸준히 최첨단 성능 모델이 개발되고 있다. POSTER++은 작년에 최우수 성능을 보였으며, 8.4G FLOPS, 43.7 M 파라미터로 가벼운 모델을 지향한다. 이는 로봇 시스템에서 중요한 요소로 작용하므로 본 연구에서는 POSTER++을 사용하여 사람의 감정을 인지한다.

모션 생성 기술은 주어진 데이터를 통해 사람의 모션을 생성하는 분야로, 일반적으로 컴퓨터 그래픽스나 캐릭터 모델링 등에서 사용한다. 최근 FLAME[8] 등 Diffusion Model을 활용하는 연구가 진행되고 있다. 특히 FLAME은 모션 생성 모델 중 최초로 Denoising Diffusion Probabilistic Models (DDPM)[9] 방식을 활용하여 최우수 성능을 보였다. 하지만 모션 생성 기술이 컴퓨터 그래픽스에서만 사용되는 것은 아쉬운 부분이다. 이에 착안하여 본 논문에서는 FLAME을 채택하여 모션을 생성하고, 이를 구현한다.

제안하는 시스템에서는 사람의 감정을 인지, 이에 적절한 행동을 구현하기 위해 얼굴 탐색 프로세스, 감정-모션 프로세스, 기구학 해석 프로세스의 총 3가지로 이루어져 있다. 첫 번째로 얼굴 탐색 프로세스에서는 근접한 사람의 얼굴을 포착하고 해당 인물에게 초점을 맞춰 상호작용하는 대상을 인지시킨다. 다음으로 감정-모션 프로세스에서는 POSTER++로 감정을 인지하고 대규모 언어모델(LLM)인 Gemini AI를 사용하여 프롬프트 엔지니어링을 통해 각 감정에 적절한 반응 텍스트 출력한 다음, 이를 FLAME에 입력하는 것으로 모션을 생성한다. 마지막으로 기구학 해석 프로세스에서는 출력된 관절 데이터로 정방향 기구학(FK, Forward Kinematics) 및 역기구학(IK, Inverse Kinematics) 해석을 통해 양팔의 각 관절 위치 및 각도를 얻어 시뮬레이션한다. 실제 환경을 고려하기 위해 본 시스템은 휴머노이드 로봇 HUMIC에서 이루어지며, 해당 시나리오를 통해 사용자들의 만족도를 조사하여 평가한다.

본 논문은 총 6장으로 구성되어 있다. 2장에서는 본 연구에서 활용한 인공지능 기술 및 로봇공학에 관한 관련 연구를 소개하고, 3장에서는 로봇의 전체 사양과 제안하는 시스템의 구조를 기술하며, 4장에서는 시스템 프로세스가 어떻게 진행되는지 자세히 설명한다. 이후 5장에서는 실험 환경 및 시나리오와 이에 관한 결과를 분석하고, 마지막으로 6장에서 결론을 맺는다.

2. 관련 연구

이번 장에서는 우선 표정 인지 및 모션 생성 분야에서 최근에 활용한 기술과 동향을 소개한 후, 각 분야에서도 본 시스템에서 활용한 모델에 대하여 자세히 기술한다. 또한, 기구학 해석을 위해 필요한 수식들에 대하여 설명한다.

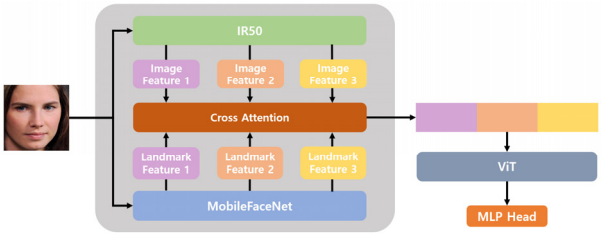
2.1 표정 인지

표정 인지 기술은 오래전부터 CNN을 활용하여 연구되었으나 최근에 ViT의 등장으로 이에 관한 연구가 활발히 진행되고 있다. [Table 1]은 최근 ViT 모델을 응용하여 개발한 모델들과 성능 지표를 보여준다. 2021년도부터 RAF-DB의 경우 정확도가 90%를 넘었으며, AffectNet에서는 66%를 가뿐히 넘겼다. 이는 Transformer의 Self-Attention이 얼굴의 특정 부위를 강조할 수 있도록 도와 표정을 학습하게 해주기 때문이다. 또한, 표정 인지 분야에서는 ViT 모델의 문제점 중 하나인 계산량과 파라미터 수를 줄이기 위해 1차로 얼굴 이미지의 특징 추출 단계를 거친다. 이를 통해 얼굴의 핵심 정보만을 ViT에서 학습한다는 장점도 존재하며, 실제 성능도 크게 향상된 것을 확인할 수 있다.

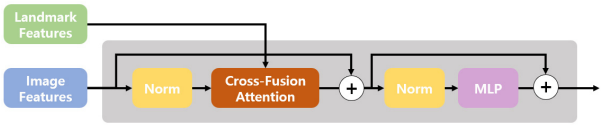
POSTER++에서도 마찬가지로 특징 추출과 ViT 학습의 두 단계를 가지며, 얼굴의 랜드마크 정보를 통해 클래스 간 불일치와 클래스 내 유사도를 극대화하는 것으로 학습한 모델이다. [Fig. 1]은 POSTER++의 전체 아키텍처를 보여준다. 우선 특징 추출 단계에서 일반 얼굴 이미지의 특징은 Improved ResNet50 (IR50)[11]으로, 얼굴의 랜드마크 이미지 특징은 MobileFaceNet[12]으로 추출한다. 이때 각각 크기가 다른 특징을 3개씩 가져온다. 여기서 크기마다 Cross-Attention을 통해 두 특징으로부터 하나의 새로운 특징을 얻는데, 이때 일반 얼굴 이미지 특징의 쿼리

[Table 1] The performance of ViT models on Facial Expression Recognition

Model	Date	RAF-DB (%)	AffectNet (%)
TransFER	May. 2021	90.01	66.23
APViT[10]	Dec. 2022	91.98	66.91
POSTER	Apr. 2022	92.05	67.31
POSTER++	Jun. 2023	92.21	67.49
S2D	Dec. 2023	92.57	67.62



[Fig. 1] Architecture of POSTER++. In this model, first it has each feature extraction. IR50 extracts normal facial image features and MobileFaceNet extracts facial landmark image features. Then, landmark features guide to focus on salient regions of facial expressions by cross-attention. After combining the 3 different sizes features, it train by ViT

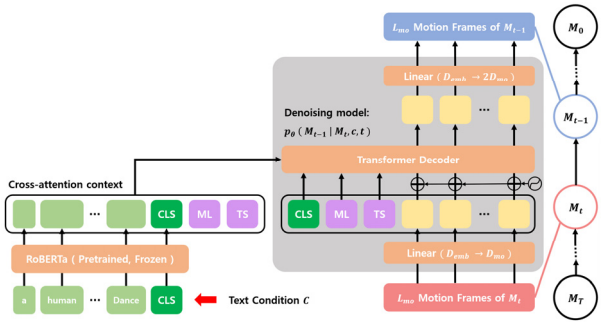


[Fig. 2] This is the framework of cross-attention mechanism. Facial landmark features interact with vanilla facial image features as queries of the landmark features. So, keys & values of the image features and queries of the landmark image features are combined

대신 랜드마크 이미지 특징의 쿼리를 입력하여 Attention을 계산한다. 이는 랜드마크 이미지 특징이 표정의 중요 얼굴 부위를 참고하기 위함이다. Cross-Attention에 대한 동작 구조는 [Fig. 2]에서 보여준다. 이후 서로 다른 크기의 특징을 하나의 시퀀스로 나열하여 ViT에 입력하는 것으로 Self-Attention을 통해 학습을 진행한다. POSTER++는 기존 POSTER 모델을 경량화하여 다른 ViT 모델 중에서도 8.4G FLOPS, 43.7 M 파라미터로 가벼운 모델에 해당하며, 최근 SOTA 모델인 S2D와 RAF-DB에서 0.36%, AffectNet에서 0.13%의 정확도 차이로 준수한 성능을 보여준다.

2.2 모션 생성

Diffusion Model은 본래 그림 생성 분야에서 사용하는 학습 모델이며, DDPM이 나오면서 더욱 큰 발전이 이루어졌다. DDPM은 입력 이미지에 고정된 가우시안 분포로 생성한 노이즈를 입힌 후, 이를 학습된 정규 분포를 통해 노이즈를 제거하는 것으로 원본 이미지와 생성 이미지가 같아지도록 학습한다. 최근 그림 생성 분야에서 큰 발전이 이루어지면서 모션 생성 분야에서도 Diffusion Model에 관심을 보이기 시작했다.



[Fig. 3] Architecture of FLAME that achieves SOTA performance. In the FLAME, 3D motion data is noised until time T, and then denoised to learn the 3D motion data. It has TS (Diffusion Time-Step) Token contained frame information and ML (Motion-Length) Token that is total length of motion. And the context used as a condition is extracted through RoBERTa by referring to the Classifier Guidance module

FLAME은 최초로 DDPM을 3D 모션 생성 분야에서 적용한 모델이다. [Fig. 3]은 이를 아키텍처로 표현한 것이다. 그림 대신 3D 모션 데이터에 노이즈를 추가한 뒤 다시 제거하면서 학습을 진행한다. 이때 그림 생성 모델과 달리 모션마다 프레임 수에 차이가 존재하기에 프레임 정보가 담긴 Diffusion Time-Step Token (TS)과 모션의 총 길이 정보를 알려주는 Motion-Length Token (ML)을 추가했다. 또한, Classifier Guidance[13]를 참고하여 RoBERTa를 통해 추출한 context를 condition으로 사용함으로써 학습에 도움을 준다. FLAME은 가장 최근에 최우수 성능을 보인 모델이며, 이때 FLAME에서 출력되는 모션 데이터는 기준 관절(Root Joint)의 위치 벡터 (x , y , z)와 각 관절의 회전 벡터인 사원수로 구성되어 있다.

2.3 로봇공학

2.3.1 사원수

사원수는 3차원 그래픽에서 회전을 표현할 때 사용하는 수학적 개념으로, 복소수 체계로 이루어진 수이다.

$$q = q_w + q_x i + q_y j + q_z k \tag{1}$$

사원수를 이용하는 이유는 행렬에 비해 연산 속도가 빠르고, 메모리 용량도 적다. 또한, 오일러각을 사용했을 때 Gimbal-lock 문제가 발생하여 모든 각도 변환을 표현하는 데에 한계가 있기 때문이다. Gimbal-lock이란 한 오브젝트의 두 회전축이 같은 방향으로 겹쳐 남은 하나의 회전축이 움직일 수 없게 되는 현상을 말한다. 이는 오일러각이 자체적으로 설정된 순서대로 축들을 개별적으로 계산하기 때문에 발생하며, 이로 인해 오일러각 대신 회전을 표현할 때 사원수를 사용한다. 오일러각과 사원수 간의 치환 방정식은 식 (2)처럼 표현한다.

$$\begin{aligned} q_w &= \cos(\alpha/2) \\ q_x &= \sin(\alpha/2)\cos(\beta_x) \\ q_y &= \sin(\alpha/2)\cos(\beta_y) \\ q_z &= \sin(\alpha/2)\cos(\beta_z) \end{aligned} \tag{2}$$

식 (1)에서 α는 오브젝트 자체의 회전 각도를 말하며, β는 각각 x, y, z 축을 기준으로 회전한 오일러각을 의미한다.

회전행렬을 표현할 때에도 문제가 발생한다. 오일러각은 각 축에 대하여 회전행렬이 별개로 존재하기 때문에 roll, pitch, yaw만큼 곱해야 하는데, 이때 행렬은 Post-multiplication 규칙을 사용하기 때문에 회전 순서에 따라 전혀 다른 좌표계가 된다. 반대로 사원수는 식 (3)처럼 단 하나의 회전행렬로 표현할 수 있다.

$$R(Q)=\begin{bmatrix}2(w^2+q_x^2)-1 & 2(q_xq_y-wq_z) & 2(q_xq_z+wq_y)\\ 2(q_xq_y+wq_z) & 2(w^2+q_y^2)-1 & 2(q_yq_z-wq_x)\\ 2(q_xq_z-wq_y) & 2(q_yq_z+wq_x) & 2(w^2+q_z^2)-1\end{bmatrix} \quad (3)$$

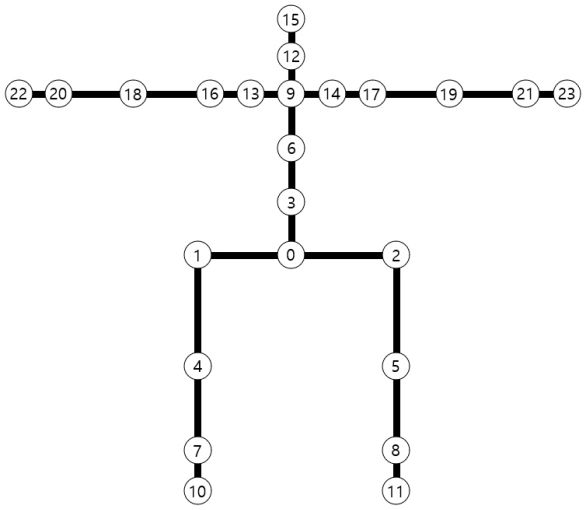
2.3.2 T 포즈 정의

T 포즈는 모션 생성과 포즈 추정에서 사람의 초기 포즈를 정의한다. [Fig. 4]는 관절이 24개인 모델의 T 포즈를 보여준다. 각 관절의 좌표계는 z 축이 정방향을 향하도록 배치하며, 이때 양손과 양발, 머리의 End-effector에는 축이 필요 없다. 또한, 기준 관절을 제외하고 모든 관절은 위치 벡터 없이 방향 벡터와 관절 간의 링크 길이만으로 구성되어 있다. 여기서 기준관절은 일반적으로 골반으로 설정한다. 여기서 각 관절의 위치를 계산하기 위해 Chain of Matrix Multiplication을 사용하며, 이는 식 (4)로 표현된다.

$$\vec{p}_n=\vec{p}_0+R_0\vec{J}_1+R_0R_1\vec{J}_2+\cdots+R_0R_1R_2\cdots R_{n-1}\vec{J}_n \quad (4)$$

각각 p는 목표 관절의 위치, R은 각 관절의 회전행렬, $\vec{J}$는 각 관절 사이의 거리를 의미한다. 회전행렬의 곱셈을 할 때, 해당 모델의 키네마틱 트리 정의가 필요하며, 이를 통해 연결된 키네마틱 트리 정보를 따라 계산해야 한다. 예를 들어, [Fig. 4]에서 12번 관절의 위치를 계산하고 싶다면 기준 관절의 위치 및 회전행렬과 3, 6, 9번 관절의 회전행렬을 알고 이를 순서대로 곱해야 한다. 아래 식 (5)는 이를 표현한 것이다.

$$\vec{p}_{12}=\vec{p}_0+R_0\vec{J}_3+R_0R_3\vec{J}_6+R_0R_3R_6\vec{J}_9+R_0R_3R_6R_9\vec{J}_{12} \quad (5)$$



[Fig. 4] T-pose Definition of 24 Joints Human Model. It is initial pose of human model, and has translation of root joint, rotation quaternion of each joint. We can obtain position vector of each joint by calculating chain of matrix multiplication

키네마틱 트리의 정의와 각 관절의 데이터를 안다면, 이를 토대로 모든 관절의 절대 좌표계 위치를 모두 얻을 수 있다.

3. 로봇 시스템

본 연구에서는 HRI 로봇 시스템을 구현하기 위하여 휴머노이드 로봇 HUMIC을 사용하여 진행한다. HUMIC은 성인 크기의 휴머노이드 로봇으로, 사람과 구조가 유사하게 제작되었다. [Table 2]는 본 연구에서 사용한 HUMIC의 전체 사양을 기술한 것이다. 전체 무게는 98 kg이며, Kinect V2를 통해 RGB 및 Depth 데이터를 얻을 수 있다. 머리에는 서보 모터인 Dynamixel Pro가 2 자유도(DoF, Degrees of Freedom), 팔과 손에 각각 6 자유도와 9 자유도로 구성되어 있다. 몸통에 임베딩되어 있는 모터 컨트롤러 OpenCR 1.0을 통해 서보 모터를 제어하며, 메인 CPU인 Nvidia Jetson AGX Xavier에 ROS (Robot Operating System)를 설치하여 전체적인 프로세스를 유기적으로 처리할 수 있게 한다. 그리고 TCP/IP 내부 통신이 가능하도록 돕는 라우터 IpTIME A7004M이 부착되어 있다. 하단 모바일 플랫폼에 4륜 매커넘 휠이 달려있어 쉽고 빠르게 전방향 이동이 가능하며, 이는 모터 드라이버를 통해 MCU Arduino Mega 2560으로 제어할 수 있다. 마지막으로 HUMIC의 키 높이를 조절할 수 있도록 해주는 리프팅 기능이 있어 1.1 m ~ 1.6 m 사이 조절이 가능하고, 전방 및 후방 장애물 탐지가 가능하도록 해주는 초음파 센서가 2개씩 부착되어 있다.

[Table 2] Total specification of humanoid robot HUMIC

Parts	Details	
Weight	98 kg	
Height	Maximum	1.60 m
	Minimum	1.10 m
Head	RGB-D Camera	Kinect V2
	Neck	H54-100-S500-R
	Head	H42-20-S300-R
Arm	Dinamixel Pro Manipulator-H (6 DoF) x2	
Hand	OpenBionics Brunel V2 (9 DoF) x2	
Body	Motor Controller	OpenCR 1.0
	CPU	Nvidia Jetson AGX Xavier
	Router	IpTIME A7004M
Mobile	Mobile Platform	Programmable Mecanum Wheel Vectoring Robot - IG52DB
	MCU	Arduino Mega 2560 x2
	Lift Column	STC500700
	Motor Driver	SZH-GNP521
	Proximity Sensor	HC-SR04P x4

[Table 3]에서는 본 연구에서 설정한 각 관절의 한계 각도를 보여준다. 이는 사람의 관절 움직임을 기준으로 하며, 여기서 팔꿈치(Elbow)의 경우에는 사람의 팔꿈치 관절 회전 범위가 좁으므로 이에 맞춰 좁게 설정했다. 또한, 아래쪽 팔(Lower Arm)과 손(Hand)은 End-effector의 위치에 영향을 주지 않아 제한을 두지 않았다. [Table 4]에서는 HUMIC 로봇팔 관절 사이의 링크 길이를 기술한다. 팔의 자유도는 각각 어깨에 2 자유도, 팔꿈치에 1 자유도, 손목에 3 자유도로 구성되어 있다. [Table 5]는 Kinect V2의 전체 사양을 기술한다. RGB-D 카메라인 Kinect V2는 3차원 이미지뿐 아니라 심도(Depth) 값도 알 수 있어서 이미지의 각 픽셀에 대응하여 심도 맵을 얻을 수 있다. 그러나 심도 측정 범위의 최소가 0.5 m이므로 해당 거리 이상인 상황에서만 심도 측정이 가능하다. 본 연구에서는 사람과 로봇이 근접한 상황을 전제로 실험하므로 대상과의 정확한 거리는 측정할 수 없다.

본 논문에서 제안하는 로봇 시스템에서는 HUMIC에서 상단의 Kinect V2와 머리에 해당하는 Dynamixel Pro 서보 모터, 이를

[Table 3] Limit Angle of Each Arm Joint of HUMIC

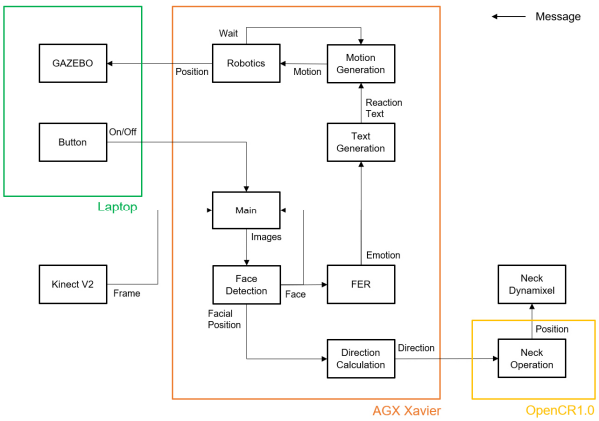
Joints	Limit Angle
Shoulder	-180° ~ 180°
Upper Arm	-90° ~ 90°
Elbow	0 ~ 90°
Lower Arm	-180° ~ 180°
Wrist	-90° ~ 90°
Hand	-180° ~ 180°

[Table 4] Each Arm Link of HUMIC

Parts	Link
Shoulder ~ Upper Arm	29.7 mm
Upper Arm ~ Elbow	285.05 mm
Elbow ~ Lower Arm	244.81 mm
Lower Arm ~ Wrist	32.4 mm
Wrist ~ Hand	129.6 mm

[Table 5] The specification of Kinect V2

Color	Resolution	1920 x 1080
	fps	30
	Field of View	84.1° × 53.8°
Depth	Resolution	512 × 424
	fps	30
	Field of View	70° × 60°
Depth Mesuring Range		0.5 m ~ 4.5 m



[Fig. 5] Block Diagram of our proposed HRI robot system. There are three processors: laptop, Nvidia Jetson AGX Xavier, and OpenCR 1.0. In the Laptop, it can turn on/off our system and simulate the HUMIC using both arm manipulators. In the OpenCR 1.0, it just controls neck and head motors. And the other nodes (Face Detection, FER, Text Generation, Motion Generation, Robotics) operate on the main processor

[Table 6] ROS node descriptions of our HRI robot system

	Node	Description
Laptop	Button	Robot system on/off
AGX Xavier	Main	System and image process
	Face Detection	Face detection
	Direction Calculation	Human direction calculation
	FER	Facial expression recognition
	Text Generation	Interaction text generation
	Motion Generation	Interaction motion generation
	Robotics	Forward and inverse kinematics calculation
OpenCR 1.0	Neck Operation	Moving robot head direction

제어하기 위한 OpenCR 1.0을 사용한다. 메인 프로세서 Nvidia Jetson AGX Xavier를 통해 POSTER++와 FLAME 모델뿐 아니라 대부분의 프로세스를 제어하고, 이를 통해 얻은 모션을 외부 컴퓨터에서 Gazebo 시뮬레이터로 구현한다. 또한, ROS를 사용하여 시스템 구조를 구축했다. ROS를 통해 로봇의 센서, 모터 등 여러 부품을 유기적으로 구동할 수 있으며, TCP/IP 통신으로 여러 PC 간 메시지 교환도 가능하다.

[Fig. 5]는 제안하는 HRI 시스템의 전체 구조를 보여준다. 본 시스템에서는 외부 컴퓨터와 메인 프로세서, 모터 컨트롤러의 세 보드 간에 데이터를 유기적으로 처리한다. 외부 컴퓨터에서는 로봇 시스템을 Button 노드를 통해 On/Off하며, Gazebo 시뮬레이터를 통해 로봇의 양팔 모션을 시연한다. OpenCR 1.0에서는 얼굴을 포착하여 추적하기 위해 목과 머리 관절 모터를 제

어한다. 그리고 메인 프로세서에서 이외의 모든 주 프로세스인 이미지 처리, 얼굴 탐색, 표정 인지, 텍스트 생성, 모션 생성, 기구학 해석이 실행된다. [Table 6]에서는 본 시스템의 메인 노드들을 간략히 기술한다.

4. HRI 프로세스

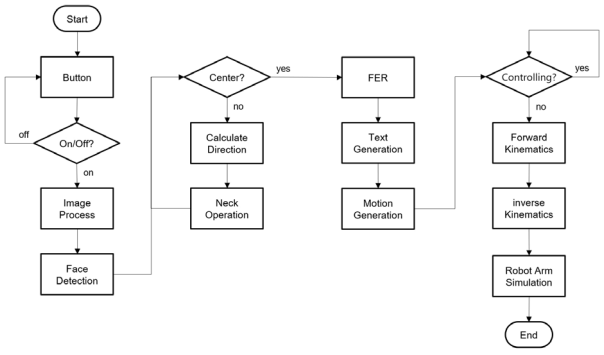
이번 장에서는 전체 프로세스와 각 메인 노드에서 이루어지는 프로세스들을 서술한다. 전체적으로 3가지 프로세스가 있으며, 이는 각각 사람의 얼굴을 탐색하여 포착하는 얼굴 탐색 프로세스, 대상의 감정을 인지하여 모션을 생성하는 감정-모션 프로세스, 모션 데이터로 각 관절의 위치 및 각도를 계산하는 기구학 해석 프로세스로 구분한다.

4.1 전체 프로세스

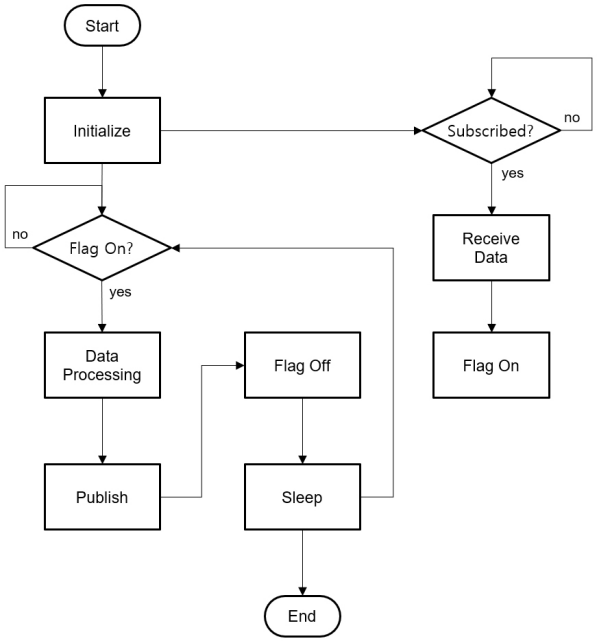
[Fig. 6]은 본 HRI 시스템의 전체 프로세스를 순서도로 간략히 보여준다. 노트북에서 On/Off 명령을 내릴 수 있으며, Off 일 경우 메인 노드를 제외한 모든 노드가 종료된다. 이때 메인 노드에서는 On/Off 상관없이 Kinect V2로부터 얻은 이미지를 실시간으로 출력한다. 그리고 On 커맨드 입력 시 HRI 시스템 프로세스가 구동되고, 얼굴 탐색 프로세스, 감정-모션 프로세스, 기구학 해석 프로세스가 차례로 동작하면서도 동시에 진행된다.

[Fig. 4]의 (a) 얼굴 탐색 프로세스에서는 Kinect V2으로 RGB 및 심도 이미지를 얻어 사람의 얼굴을 포착하고, 이때 근처 사람이 있다면 그 사람의 얼굴이 중심에 위치하도록 머리 관절을 움직여 추적한다. 다음으로 [Fig. 4]의 (b) 감정-모션 프로세스에서는 POSTER++를 통해 사람의 표정 인지, Gemini AI로부터 해당 감정에 적절한 반응을 텍스트로 얻으며, FLAME에 이를 입력하여 모션 데이터를 획득한다. 마지막으로 [Fig. 4]의 (c) 기구학 해석 프로세스에서는 모션 데이터로부터 각 관절의 회전 벡터를 얻어 FK를 통해 각 관절의 위치를 계산한 후, 그대로 IK를 풀이하여 얻은 각 관절의 각도를 Gazebo 시뮬레이터에서 모델링한 HUMIC으로 구현한다. 이때 모션 생성 노드는 하나의 모션을 생성한 후 시뮬레이션 상에서 해당 모션의 동작이 끝날 때까지 대기 상태가 된다.

[Fig. 7]은 본 시스템에서의 기본적인 노드 프로세스를 설명한다. 처음에 노드를 초기화한 후 프로세스가 시작되는데, 이때 메시지를 전달받을 때까지 대기하며, 메시지가 오면 Flag를 On 상태로 전환한다. 여기서 Flag는 하나의 노드가 동작할 때 다른 노드로부터 데이터를 받았는지 확인하는 용도로 쓰이며, Flag가 On 상태가 되면 데이터를 각 노드의 역할에 알맞게 처리한다. 이후 다음 노드에게 메시지를 전달하면 Flag는 Off 상태가 되어 다시 메시지를 받을 때까지 대기한다. 여기서 Sleep은 일정 시간동안 대기하도록 하며, 모든 노드가 10 ms로 일정한 프로세스 처리 시간을 보내도록 하는 역할을 갖는다.



[Fig. 6] Flow chart of our proposed HRI robot system process. There are 3 main processes. (a) Face Detection Process : Detect a human face nearby and focus on that person's face (b) Emotion-Motion Process : Recognize human emotion on POSTER++, Generate and receive appropriate reaction text to emotion on Gemini AI, Generate and get from reaction text to motion data on FLAME (c) Robotics Process : Obtain each joint's angle to calculate FK and IK and control on Gazebo

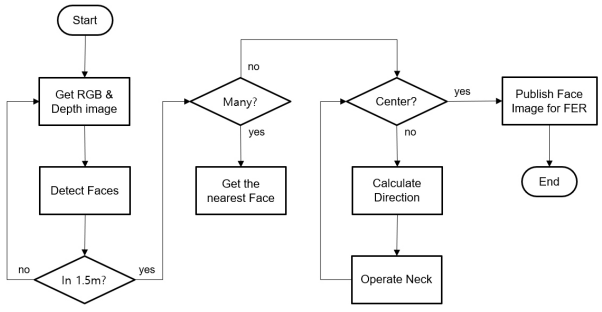


[Fig. 7] Flow chart of the basic node process. This process first initialize variables, and then check whether a message is subscribed. If it has been received, the flag on. After this, the node process received data and publish some messages. After all the processing, the flag off and the node wait for data to be subscribed

4.2 얼굴 탐색 프로세스

사람의 얼굴을 탐색하기 위해 사람이 많은지, 대상과의 거리는 가까운지, 그 사람이 로봇을 보고 있는지 등 많은 조건과 상황에 맞춰 대응해야 한다. 이 때문에 본 연구에서는 Kinect V2로부터 RGB 이미지와 심도 데이터를 모두 사용한다. 여기서 대상과의 거리를 파악하기 위해 RGB 카메라와 심도 카메라의 내·외부 파라미터 및 왜곡 정도를 구하여 카메라 캘리브레이션을 진행하며, 이를 통해 RGB-D 이미지를 생성한다.

[Fig. 8]은 얼굴 탐색 세부 프로세스를 설명한다. 해당 프로세스에서 본 시스템은 카메라에 보이는 얼굴을 탐색한 후, 로봇이 누구와 상호작용하고 있는지 대상이 알 수 있도록 얼굴을 추적한다. 첫 번째로 Kinect V2로부터 RGB 및 심도 이미지를 얻고, RGB 이미지로부터 얼굴을 탐색하여 추출한다. 다음으로 카메라 캘리브레이션으로 얻은 RGB-D 이미지를 통해 추출된 얼굴의 중심 좌표에 있는 심도 값을 얻는다. 이때 심도 값이 1.5 m를 기준으로 안쪽인 경우, 로봇은 대상과 상호작용하는 상태로 간주한다. 이후 얼굴의 수를 파악하며, 둘 이상이면 더 가까운 사람을 포착한다. 여기서 심도 카메라의 최소 측정 범위가 0.5 m이므로 그 이내에서 둘 이상 잡힐 경우는 없는 것으로 간주했다. 마지막으로 얼굴이 이미지에서 중심에 있는지를 판단하여 중심이 아닌 경우 대상을 바라보지 않는 상태로 판단하여 얼굴을 추적하기 시작한다. 얼굴의 중심 좌표가 이미지의 중심 좌표로부터 폭의 1/6, 높이의 1/4 이상 벗어났을 경우 얼굴이 중심에 없다는 것으로 간주하며, 이때 목 동작(Neck Operation) 노드에 움직일 방향을 전달한다. 그러면 이 노드를 통해 Open CR 1.0에서 목 관절을 움직여 얼굴이 중심에 올 때까지 머리를 회전시킨다. 마지막으로 이미지의 중심 좌표와 얼굴 중심 좌표가 오차 범위 내로 안정되어 정지하면 표정 인지 노드에 얼굴 이미지를 전송한다.



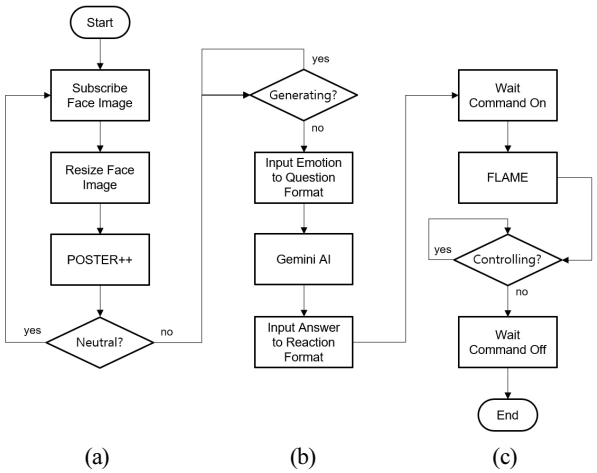
[Fig. 8] Flow chart of the Face Detection & Tracking process. This process first calibration RGB and depth cameras from Kinect V2, then get RGB and depth images. Second, it detect faces and focus on a person's face inside 1.5 m. If the face is not in center, it caculate which direction to move and then operate neck joints. Finally, when the face is in center, publish the face image for FER node

4.3 감정-모션 프로세스

감정-모션 프로세스에서는 3개의 개별 노드가 각각 동작한다. 각각 사람의 표정 이미지로부터 감정을 인지하는 표정 인지 노드, 인지한 감정을 토대로 반응 텍스트를 생성하는 문장 생성 노드, 이를 통해 모션 데이터를 생성하는 모션 생성 노드이다. [Fig. 9]는 감정-모션 프로세스를 자세히 기술한다.

표정 인지 노드에서는 AffectNet 데이터셋으로 학습된 POSTER++ 모델을 통해 사람의 표정을 인지한다. POSTER++에서는 얼굴 이미지를 112 x 112 크기로 입력받기 때문에 수신받은 얼굴 이미지의 크기를 재조정한 후 POSTER++에 입력한다. 그러면 모델 내부에서 IR50, MobileFaceNet, ViT를 거쳐 감정 상태를 도출한다. 이때 감정이 무표정이 아닌 상태가 될 때까지 대기하며, 대상이 다른 감정 상태가 되면 이를 문장 생성 노드에 전달한다.

문장 생성 노드에서는 구글의 생성 모델인 Gemini AI를 통해 프롬프트 엔지니어링을 진행한다. 본 시스템에서는 여러 프롬프트 엔지니어링 중에서 제로샷 프롬프트 방식을 사용한다. 이를 위해 하나의 완성된 문장 포맷을 작성하였으며, 이는 [Table 7]에 기술되어 있다. 질문(Question), 대답(Answer) 포맷에는 각각 '감정(Emotion)', '반응(Reaction)'을 입력할 수 있게 한다. 이때 감정과 반응 안에 들어가는 것은 단어 하나로 제한한다. 예를 들어, '슬픔'이라는 감정 상태를 입력한다면 Gemini



[Fig. 9] Flow chart of the Emotion-to-Motion Generation process. This process obtain from emotion to 3D motion data. (a) FER : This node recognize the human emotion from the face on POSTER++. (b) Text Generation : If the emotion is not neutral, this node received reaction text from Gemini AI and input the reaction text into our text format. (c) Motion Generation : This node generate 3D motion data to input the text format on FLAME. If FLAME is operating, it does not receive the reaction text, and if the motion is simulating, FLAME does not publish new motion data to Robotics node

[Table 7] Each Arm Link of HUMIC

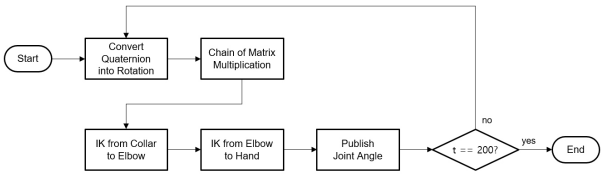
Question	If my friend is {Emotion}, what kind of movement would I do with my both arms? Pick one and tell me.
Answer	A person {Reaction} with his both arms.

AI에서 'hug'라는 동사를 출력하며, 이를 '반응'에 입력하여 문장을 완성한다. 마지막으로 대답 포맷을 전달하기 전, 만약 모션 생성 노드에서 대기 커맨드(Wait Command)가 On이라면 해당 노드에서 Emotion 데이터를 처리하지 않고 모션 생성이 끝날 때까지 대기하고, Off 상태가 되면 마지막으로 전달받은 감정 상태 메시지를 질문 포맷에 입력한다.

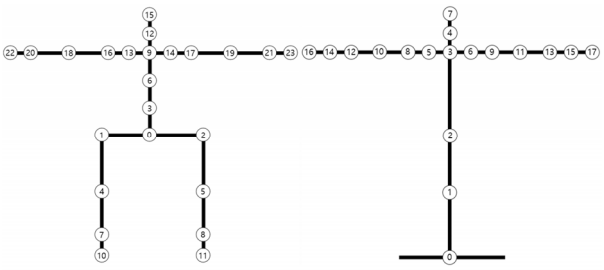
모션 생성 노드에서는 HumanML3D 데이터셋으로 학습된 FLAME 모델을 통해 행동 문장을 모션 데이터로 출력한다. 모션을 생성하기 전, 대기 커맨드를 On 상태로 바꾼 후, FLAME에 반응 텍스트를 입력하여 모션 데이터를 얻는다. 마지막으로 Gazebo 시뮬레이터에서 HUMIC의 양팔이 제어되고 있다면 완전히 끝날 때까지 대기하고, 이후 생성된 모션 데이터를 기구학 해석 프로세스에 전달한다.

4.4 기구학 해석 프로세스

기구학 해석 프로세스에서는 FLAME에서 얻은 모션 데이터로부터 각 관절의 위치를 계산하는 FK와 각 관절의 위치 정보를 토대로 관절의 각도를 계산하는 IK, 마지막으로 모션을 시뮬레이터의 HUMIC 모델에 적용하는 과정을 진행한다. 이때, 모션 데이터는 시간 t에 대하여 200 frame을 갖는다. [Fig. 10]은 해당 프로세스를 간략히 설명한다.



[Fig. 10] Flow chart of the Robotics Process. This node is first convert quaternion into rotation matrix, then calculate FK to obtain each joint's position vector. After calculating FK, IK from Collar to Elbow and from Elbow to Hand. Finally, each joint angle is published into Gazebo to simulate the motion



(a) (b)

[Fig. 11] T-pose Definition of FLAME and HUMIC. (a) T-pose of FLAME : It has 24 joints and its arms have each 5 joints. (b) T-pose of HUMIC : It has 17 joints and its arms have each 6 joints

첫 번째로, FK를 계산하기 위해서는 FLAME에서 주는 모션 데이터를 토대로 T 포즈를 정의한 후 Chain of Matrix Multiplication을 진행해야 한다. 그러나 FLAME에서 정의하는 T 포즈와 HUMIC의 T 포즈는 명백히 다르다. 이는 [Fig. 11]에서 보여주는 각각의 T 포즈와 각각의 키네마틱 트리 정보를 기술한 [Table 8]을 통해 직관적으로 확인할 수 있다. 따라서 본 프로세

[Table 8] Kinematic Tree Definition of each FLAME and HUMIC model

Model	Kinematic Tree Definition
FLAME	00_Pelvis └01_L_Hip └04_L_Knee └07_L_Ankle └10_L_Foot └02_R_Hip └05_R_Knee └08_R_Ankle └11_L_Foot └03_Spine1 └06_L_Spine2 └09_L_Spine3 └12_Neck └15_Head └13_L_Collar └16_L_Shoulder └18_L_Elbow └20_L_Wrist └22_L_Hand └14_R_Collar └17_R_Shoulder └19_R_Elbow └21_R_Wrist └23_R_Hand
HUMIC	00_Pelvis └01_Spine1 └02_L_Spine2 └03_L_Spine3 └04_Neck └07_Head └05_L_Shoulder └08_L_Upper_Arm └10_L_Elbow └12_L_Lower_Arm └14_L_Wrist └16_L_Hand └06_R_Shoulder └09_R_Upper_Arm └11_R_Elbow └13_R_Lower_Arm └15_R_Wrist └17_R_Hand

스에서는 우선 FLAME과 HUMIC의 키네마틱 트리 정보를 비교해 변환하는 작업을 진행한다.

FLAME과 HUMIC의 T 포즈 정의는 다르지만 본 시스템에서는 HUMIC 모델의 양팔만을 사용하기 때문에 우선 FLAME의 하단을 제외한다. 더불어 HUMIC은 FLAME과 달리 척추 관절 대신 하나의 기둥으로 구성하므로 이에 대하여 HUMIC에 임의로 척추 관절의 위치를 지정한다. 가장 중요한 양팔의 경우에 FLAME은 관절이 5 자유도이지만 HUMIC은 관절이 6 자유도이다. 그러나 여기서 HUMIC의 아래쪽 팔(Lower arm) 관절은 End-effector의 위치 벡터에 어떤 영향도 주지 않는다. 즉, HUMIC에서 아래쪽 팔 관절을 제외하면 두 모델 다 5 자유도로 일치하게 된다. 그러므로 팔 한정으로 FLAME과 HUMIC은 같은 키네마틱 트리 정보를 갖는다.

HUMIC의 모든 관절 링크 길이 및 FLAME으로부터 얻은 기준 관절의 위치 벡터와 각 관절의 사원수로 이전 식 (4)을 통해 FK를 계산할 수 있다. 다음으로 모션을 그대로 재현하기 위해 각 관절의 위치에 대하여 End-effector로 설정한 후 IK 계산을 진행해야 하지만, 이는 비효율적이다. 따라서 본 연구에서는 IK를 어깨 관절에서부터 팔꿈치 관절까지와 팔꿈치 관절에서 손 관절까지의 2단계에 걸쳐 계산을 진행한다. 우선 HUMIC은 어깨 관절이 총 2 자유도로, 각 관절이 회전하더라도 위치는 이동하지 않으므로 팔꿈치 관절을 End-effector로 잡아 IK 계산을 하면 시간별 각 모션의 팔꿈치 관절의 위치를 FLAME에서의 관절 위치에 맞추는 것이 가능하다. 또한, 아래쪽 팔 관절(Lower Arm)은 End-effector의 위치에 영향을 주지 않으므로 팔꿈치 관절을 기준으로 손 관절을 End-effector로 잡아 IK를 계산하면 모든 관절의 위치를 FLAME에서의 모션과 유사하게 일치시키는 것이 가능하다. 이때 IK는 ikpy 오픈소스를 활용하였으며, 이후 계산된 각 관절의 각도를 Gazebo 내의 HUMIC 모델에 입력하여 모션을 구현한다.

5. 실험 및 결과

본 논문에서 제안하는 시스템의 성능 평가를 위하여 실험을 진행했다. 이번 장에서는 실험 환경에 대하여 설명하고, 우선 각 시스템 프로세스가 유기적으로 동작하는지 검증한다. 이에 따라 얼굴 탐색 프로세스, 감정-모션 프로세스 및 기구학 해석 프로세스까지를 차례로 성공적으로 수행하는지 실험한다. 마지막으로 시나리오에 따른 실험 결과를 보여준 후 연구대상자에 대하여 몇 가지 설문조사를 했다.

5.1 실험 환경

제안하는 시스템에서는 ROS, POSTER++, Gemini AI, FLAME까지 다양한 모델과 개발 패키지가 있으며, 이를 하나의 환경에서 개발할 수 있도록 세팅하는 것은 중요하다. 이에 따라 노트북 PC 및 Nvidia Jetson AGX Xavier에서 동작할 수 있는 개발환경을 설계할 필요가 있다. [Table 9]는 두 PC에서의 환경을 직관적으로 보여준다.

[Table 9] Our system development environment for laptop and Nvidia Jetson AGX Xavier

	Laptop PC	Nvidia Jetson AGX Xavier
OS	Ubuntu 20.24 LTS	JetPack 5.1.3
Python	v3.10	v3.10
ROS	Noetic	Noetic
CUDA	v12.1	v11.4
PyTorch	v2.3.1	v2.1.0a
PyTorch3D	v0.77	v0.75

우선 노트북에서 ROS Noetic 버전은 Ubuntu 20.04 LTS보다 높은 버전에서는 지원하지 않으며, 파이썬은 Gemini AI가 3.10 버전보다 낮으면 사용할 수 없다. Pytorch3D의 최신 버전인 0.77 버전은 Pytorch 2.3.1버전까지 사용 가능하며, 이에 따라 CUDA는 12.1 버전을 채택한다. 이와 다르게 AGX Xavier는 JetPack 5.1.3버전까지만 지원하고 있으며, OS 설치 시 자동으로 CUDA 11.4버전과 Pytorch 2.1.0a버전이 설치되기에 Pytorch3D는 0.75버전을 채택했다.

모델 중에서 POSTER++는 RAF-DB와 AffectNet 데이터셋 중에서 AffectNet으로 학습된 모델을 사용하여 표정을 인지하였고, FLAME에서는 HumanML3D와 BABEL 데이터셋 중에서 HumanML3D로 학습된 모델을 선택했다.

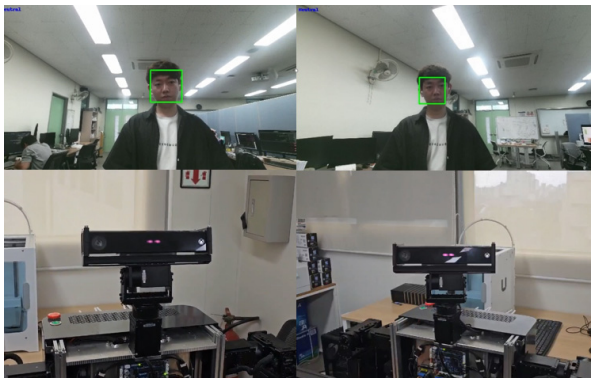
실험 환경은 사람과 로봇이 함께 있는 공간을 전제로 가정하며, 사람의 수에는 제한이 없다. 사람의 얼굴을 감지하는 범위는 1.5 m로 제한했으며, 1.5 m보다 먼 거리에서는 로봇과 상호작용하는 상황이 아닌 것으로 가정한다. 이때 실험 범위는 2.963 m × 5.187 m × 3.077 m이며, 이는 로봇팔의 사정거리에 추가로 Kinect V2의 회전 범위까지 고려하여 1.5 m를 추가한 것이다. 시나리오는 HUMIC이 사람의 표정을 보고 이에 적절한 반응 모션을 시뮬레이션하면 해당 시뮬레이션에 대하여 사용자가 평가한다. 본 시스템 서비스에 대해 만족도를 조사하며, 총 7가지 감정에 대하여 5점 척도로 평가한다.

5.2 실험 결과

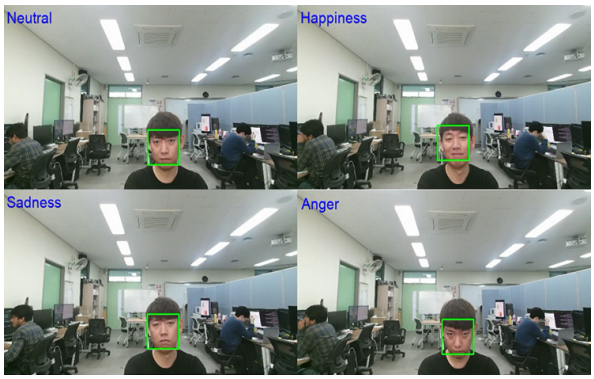
제안하는 시스템이 유기적으로 동작 가능한지 확인했으며, 얼굴 탐색, 감정-모션 프로세스, 기구학 해석과 시뮬레이션까지 진행한 후, 실제로 감정에 적절한 반응을 보이는지 실험하였다.

우선 메인 프로세스에서 모니터를 통해 얼굴 탐색 및 추적, 표정 인지까지 원활하게 돌아가는지 확인했으며, 그 결과 약

100 ms로 매우 안정적으로 데이터를 처리하는 것을 볼 수 있었다. 이때 PC를 통해 전체 시스템의 On/Off 또한 잘 동작하였다. 제한범위 1.5 m 이내에 사람이 둘 이상 있는 경우에 Kinect V2로부터 심도 값을 읽어 근접한 사람을 대상으로 포착했으며, 약 4 m에서 사람의 얼굴이 발견되어도 얼굴을 추적하거나 표정 인지를 하지 않는 것 또한 확인하였다. POSTER++가 ViT 모델임에도 HUMIC에서 큰 지연 없이 잘 동작했으나 실제 사람의 표정을 통한 감정 인지는 약 62.3% 정도의 정확도를 보였다. 또한, 무표정과 기쁨, 슬픔은 비교적 잘 인지했으나 화남과 역겨움, 놀람과 두려움을 혼동하는 모습이 확인되었다. [Fig. 12]는 얼굴 탐색 프로세스의 결과를, [Fig. 13]은 실제 환경에서 표정을 인지한 결과를 보여준다.



[Fig. 12] Results of face detection process



[Fig. 13] Results of FER using POSTER++

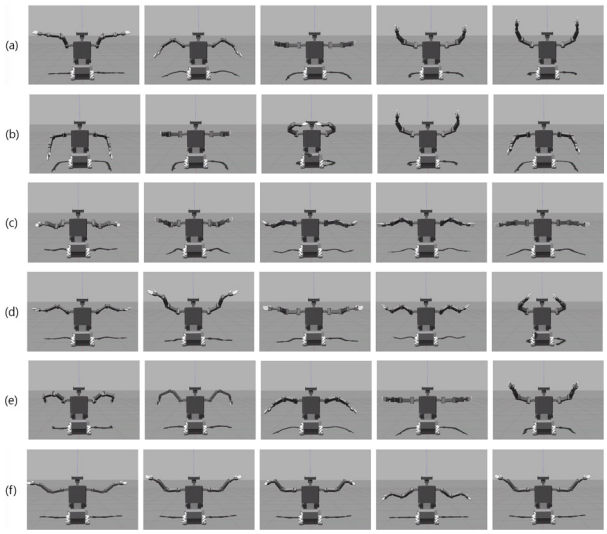
[Table 10] Reaction text of each emotion from Gemini AI

Emotion	Reaction		
Happy	Hug (36)	Shrug (17)	Clap (14)
	High-five (9)	Thumbs up (8)	Wave (7)
	Raise (5)	Double high-five (2)	
Sad	Hug (100)		
Surprise	Shrug (53)	Clap (2)	Spread (2)
Fear	Wave (63)	Flail (15)	Shrug (7)
	Hug (4)	Flutter (2)	Fee (2)
Disgust	Shrug (50)	Wave (7)	Push away (2)
Angry	Hug (91)	Shrug (6)	

[Table 10]은 문장 생성을 통해 얻을 수 있는 각 감정의 반응 텍스트이며, 감정마다 Gemini AI로부터 100회씩 텍스트를 얻어 정리한 것이다. 이때 단 한 번만 나오거나 동사를 다수 보유, 긴 문장 텍스트인 경우는 제외했다. 가장 다양한 반응 텍스트를 얻은 감정은 기쁨이며, 그 다음으로 두려움도 다양한 반응 텍스트를 얻었다. 반대로 슬픔은 'hug' 단 하나만을 출력했다.

[Fig. 14]는 Gazebo 시뮬레이터에서 HUMIC 모델을 통해 모션을 구현한 것이다. HUMIC의 팔 링크 길이는 실제와 같으며, 중력과 무게 등 실제 환경과 같도록 설정하였다. 실제로 FK, IK 과정을 거쳐 양팔 관절 각도를 입력하니 반응 텍스트에 따라 모션을 수행하는 모습을 확인할 수 있었다. 그러나 FMAME의 문제점은 두 가지 명백하게 존재한다. 우선 시뮬레이션에서 'flail' 같은 조금 복잡하거나 'wave', 'fee' 등 애매한 단어의 경우 아직 모션을 제대로 생성하지 못하는 것을 확인했다. 그리고 FLAME의 모션 생성 시간이 매우 길어 실시간으로 사람과 상호작용하는 것에 큰 어려움이 있었다. 이 때문에 모션이 실행되는 동안 미리 모션 데이터를 생성하도록 대기 명령을 하는 프로세스를 추가했음에도 전체 프로세스 시간이 약 36초로 오래 걸리는 것을 확인하였다. 이때의 지연 시간은 정확히 FLAME의 모션 생성 시간과 일치한다.

마지막으로 본 로봇 시스템에 대하여 각 감정에 따른 반응 모션 및 로봇 서비스의 만족도 조사를 했다. 총 10명의 연구대상자를 모집하여 실시했으며, 각 문항을 5점 척도(매우 만족, 만족, 보통, 불만족, 매우 불만족)로 선택하도록 했다. 첫 번째로 무표정을 제외한 총 6가지 감정에 대하여 랜덤으로 생성된 모션을



[Fig. 14] Motion simulation results of each Emotion. (a) Happy – Raise (b) Sad – Hug (c) Surprise – Spread (d) Fear – Flutter (e) Disgust – Push away (f) Anger – Shrug

[Table 11] Survey of satisfaction about reaction motion for each emotion on our proposed HRI robot system

Emotion	5	4	3	2	1	Average score
Happy	2	4	2	2	0	3.6
Sad	3	6	1	0	0	4.2
Surprise	0	1	7	1	1	2.8
Fear	0	2	3	5	0	2.7
Disgust	0	1	6	3	0	2.8
Angry	3	5	0	1	1	3.8

[Table 12] Survey of satisfaction with robot service

Category	Question	Average Score	
Reliability	The emotion recognition is accurate.	3.8	3.45
	The robot response is appropriate.	3.1	
Response	The robot service is satisfactory.	4.4	3.6
	The robot response is fast.	2.8	
Certainty	The robot system is reliable.	3.5	3.8
	The robot system is helpful.	4.1	
Empathy	The robot sympathizes with me.	3.8	4.15
	The robot is focusing on me.	4.5	

보여주었으며, 다음으로 전체 로봇 시스템 프로세스를 시연하였다. 감정에 따른 만족도 점수는 다음 [Table 11]과 같다. 기쁨의 경우, 고득점이 많으나 반응 모션의 종류가 많아 제대로 동작하지 못하는 모션을 출력하는 경우가 있어 저점도 꽤 받았으며, 화남도 'hug'가 대부분이지만 'shrug' 모션에 대해서는 낮은 점수를 받아 평균점수를 많이 깎았다. 반대로 슬픔의 경우 'hug' 모션 하나만 나와 높은 점수를 받았다. 그리고 놀람, 두려움, 역겨움 모두 어려운 반응 모션을 출력하기 때문에 낮은 점수를 받게 되었다.

다음으로 HRI 로봇 서비스에 대해 만족도 조사를 진행했다. 신뢰성, 응답성, 확실성, 감정이입의 4가지 구성에 대하여 각 2문항씩 총 8가지 문항을 준비했으며 이는 [Table 12]에서 해당 문항 및 이에 따른 평균점수를 확인할 수 있다. 만족도 조사 결과 신뢰성이 3.45점으로 가장 낮고, 반대로 감정이입은 4.15점으로 가장 높았다. 또한, 로봇이 자신에게 집중한다고 느꼈다는 말이 가장 많았으며, 개인적으로 만족스럽다는 의견도 있었다. 그러나 반응 행동과 반응 속도에서 좋은 점수를 받지 못했다. 이는 반응 모션의 단순화 및 복잡한 모션을 제대로 구현하지 못한 결과라고 본다.

6. 결 론

본 연구에서는 감정을 인지하고 이에 적절한 반응 모션을 생성하여 사람과 상호작용하는 로봇 시스템을 제안하였다. 얼굴 탐색, 감정-모션 프로세스, 기구학 해석과 시뮬레이션까지 다양한 프로세스를 유기적으로 연결하여 동작하도록 할 수 있었고, 또한 사람의 얼굴을 마주 보며 상호작용하는 모습을 볼 수 있었다. 그러나 여기에 몇 가지 수정할 부분이 있음은 분명하다.

POSTER++는 FLOPS 및 파라미터가 적어 충분히 로봇에서 실시간으로 표정 인지가 가능했으나 현재 표정 인지 분야에서 최우수 성능의 모델들과도 거의 차이 나지 않는 성능임에도 역겨움과 화남 같이 특정 감정에 대해서는 잘 구분하지 못하는 것을 확인하였고, 여전히 감정 인지 연구가 더 활발히 이루어져야 한다는 의견을 갖게 되었다. 모션 생성 부분에서는 FLAME의 경우 최대 프레임 수를 T = 200으로 설정하는 것이 가장 자연스러운 모션을 생성했으나 이때 가장 치명적인 점은 바로 모션을 생성하는 시간이 평균 36초로 매우 오래 걸린다는 것이다. 또한, 특정 모션의 경우 200 프레임 전부가 아닌 일부 짧은 시간대에만 동작하였다. 이를 해결하기 위해서는 FLAME보다 더 가볍고 준수한 모델을 사용하는 방법도 있으나 실시간 활용을 위하여 Gemini AI로부터 받은 모션 데이터를 저장하여 같은 텍스트에 대하여 저장된 모션 데이터를 불러오는 방법으로, 알맞은 모션을 곧바로 동작시킬 수 있다는 점에서 현재로서는 가장 낫다고 판단된다. 만족도 조사에서 신뢰성에서는 낮은 점수를 보였으나 반대로 감정이입, 사회적으로 도움이 될 수 있다는 문항에서는 높은 점수를 보였기 때문에 사람들의 정신적인 측면에서 만족을 줄 수 있는 HRI 로봇 시스템이 개발되었다고 볼 수 있으며, 이는 본 논문에서 목표하는 바와 같다.

인공지능을 위한 딥러닝 모델 개발은 여전히 계속되고 있으며, 꾸준히 성능이 향상되고 있다. GPT와 그림 및 음성 생성 AI 등 인공지능은 이미 우리 사회에 녹아있다. 기술의 발전으로 사람의 감정 인지 및 휴머노이드 로봇의 적절한 모션 생성 등 꾸준히 기술이 개발된다면 사람의 감정과 정신적인 교감까지를 기대하는 바이다.

References

[1] W. H. Lee and M. J. Jung, "IROS 2013 Social HRI research trends," *Korea robotics society review*, vol. 11, no. 1, pp. 14-24, Jan., 2014, [Online], https://www.dbpia.co.kr/pdf/pdf View.do?nodeId=NODE02361684.

[2] B. Kang and S. Jun, "User Perception of Social Robot's Emotional Expression and Forms of Anthropomorphism," *Archives of Design Research*, vol. 35, no. 2, pp. 137-153, May, 2022, DOI: 10.15187/adr.2022.05.35.2.137.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Housby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *arXiv: 2010.11929*, 2021, DOI: 10.48550/arXiv.2010 .11929.

[4] F. Xue, Q. Wang, and G. Guo, "TransFER: Learning Relation-aware Facial Expression Representations with Transformers, *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada, pp. 3581-3590, 2021, DOI: 10.1109/ICCV48922.2021.00358.

[5] C. Zheng, M. Mendieta, and C. Chen, "POSTER: A Pyramid Cross-Fusion Transformer Network for Facial Expression Recognition," *arXiv:2204.04083,* 2023, DOI: 10.48550/arXiv.2204.04083.

[6] J. Mao, R. Xu, X. Yin, Y. Chang, B. Nie, and A. Huang, "POSTER++: A simpler and stronger facial expression network," *arXiv:2301.12149*, 2023, DOI: 10.48550/arXiv.2 301.12149.

[7] Y. Chen, J. Li, S. Shan, M. Wang, and R. Hong, "From Static to Dynamic: Adapting Landmark-Aware Image Models for Facial Expression Recognition in Videos," *IEEE Transactions on Affective Computing*, 2024, DOI: 10.1109/TAFFC.2024.3453443.

[8] J. Kim, J. Kim, and S. Choi, "FLAME: Free-form Language-based Motion Synthesis & Editing," *AAAI Conference on Artificial Intelligence*, vol. 37, no. 7, pp. 8255-8263, 2023, DOI: 10.1609/aaai.v37i7.25996.

[9] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," *arXiv.2006.11239*, 2021, DOI: 10.48 550/arXiv.2006.11239.

[10] F. Xue, Q. Wang, Z Tan, Z. Ma, and G. Guo, "Vision Transformer with Attentive Pooling for Robust Facial Expression Recognition," *IEEE Transactions on Affective Computing*, vol. 14, no. 4, pp. 3244-3256, Oct.-Dec., 2023, DOI: 10.1109/TAFFC.2022.3226473.

[11] J. Deng, J. Guo, J. Yang, N. Xue, I. Kotsia, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 5962-5979, Oct., 2022, DOI: 10.1109/TPAMI.2021.3087709.

[12] S. Chen, Y. Liu, X. Gao, and Z. Han, "MobileFaceNets: Efficient CNNs for Accurate Real-Time Face Verification on Mobile Devices," *Biometric Recognition*, vol. 10996, pp. 428-438, Aug., 2018, DOI: 10.1007/978-3-319-97909-0_46.

[13] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, M. Chen, "GLIDE : Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models," *arXiv:2112.1074*, 2022, DOI: 10.48550/arXiv.2112.10741.

송 호 준

2015 서울과학기술대학교 기계시스템디자인공학과(학사)

2024 서울과학기술대학교 컴퓨터공학과(석사)

관심분야: 인공지능, 로보틱스, 인간 로봇 상호작용

한 지 형

2008 KAIST 전기 및 전자공학과(학사)

2015 KAIST 전기 및 전자공학과(박사)

2015~2017 한국전자통신연구원 선임연구원

2017~현재 서울과학기술대학교 컴퓨터공학과 부교수

관심분야: 인간-로봇 상호작용, 인간 의도 파악, 지능형 로봇, 기계학습, 딥러닝