JDCS
디지털콘텐츠학회논문지
Journal of Digital Contents Society
Vol. 27, No. 1, pp. 215-224, Jan. 2026

Check for updates

# 자기중심 RGB-D 데이터를 사용한 3D 인체 포즈 추정

백 성 민[1] · 김 예 진[2*]
[1]한국전자통신연구원 콘텐츠연구본부 책임연구원
[2]홍익대학교 게임학부 부교수

# 3D Human Pose Estimation Using Egocentric RGB-D Data

**Seongmin Baek[1] · Yejin Kim[2*]**

[1]Senior Researcher, Contents Research Division, ETRI, Daejeon 34129, Korea
[2]Associate Professor, School of Games, Hongik University, Sejong 30016, Korea

**[요 약]**

웨어러블(wearable) 기술의 발전으로 사용자의 이동이 자유로워지면서 디바이스(device)의 시점(viewpoint)에서 인체 포즈(human pose)를 추정하는 방식이 큰 관심을 받고 있다. 본 논문에서는 이러한 자기중심 시점(egocentric viewpoint)에서 입력되는 깊이 데이터(depth data)를 이용하여 실시간 3D 인체 포즈(pose)를 추정하는 방법을 소개한다. 제안하는 방법에서는 head-mounted display에 장착된 다중 깊이 카메라로부터 자기중심 깊이 데이터(egocentric depth data)를 스트림(stream) 방식으로 획득하고, ResNet 기반 네트워크 (ResNet-based network)를 통해 사용자의 3D 스켈리톤 관절 위치(skeletal joint position)를 추정한 후 그 정확도를 참조(reference) 스켈리톤 관절 위치와 비교하여 보정한다. 이를 통해 자기중심 포즈 추정을 위해 필수적으로 사용되는 방대한 크기의 데이터셋(dataset)의 크기를 크게 줄이면서 인체의 다양한 포즈를 추적할 수 있다.

**[Abstract]**

Advancements in wearable technology have enabled greater user mobility, leading to significant interest in methods to estimate human poses from the device viewpoint. This paper introduces a novel method for estimating real-time 3D human poses using the depth data input from the egocentric viewpoint. The proposed method acquired egocentric depth data in a streaming manner from multiple depth cameras mounted on a head-mounted display. It then estimated the 3D skeletal joint positions of the user using a ResNet-based network. The accuracy was compared with reference skeletal joint positions for calibration. The proposed approach enables the tracking of diverse human poses while reducing the size of the large datasets traditionally required for egocentric pose estimation.

**색인어 :** 3D 포즈 추정, 깊이 데이터, 자기중심 시점, ResNet 기반 네트워크, 가상 환경
**Keyword :** 3D Pose Estimation, Depth Data, Egocentric Viewpoint, Resnet-Based Network, Virtual Environment

**http://dx.doi.org/10.9728/dcs.2026.27.1.215**



**Received** 01 December 2025; **Revised** 23 December 2025
**Accepted** 16 January 2026

***Corresponding Author; Yejin Kim**

**Tel:** +82-44-860-2122
**E-mail:** yejkim@hongik.ac.kr

## Ⅰ. 서 론

인체 포즈 추정(human pose estimation)은 애니메이션과 게임부터 감시, 의료, 인간과 컴퓨터의 상호작용에 이르기까지 다양한 애플리케이션과의 관련성으로 인해 주목받고 있으며, 최근에는 메타버스(metaverse)에서 원격 회의 등에 사용자 간 상호작용에 대한 응용이 가능해지고 있다. 이러한 기술들은 HMD(head-mounted display)를 사용하여 사용자가 시청각적으로 가상공간에 몰입하는 데 중점을 두고 있으나, 사용자의 신체 제어에 관한 연구는 부족한 편이다.

현재 메타버스에서 사용자에 대한 제어는 HMD와 콘트롤러(controller)를 이용하여 손과 머리의 위치 및 회전 정보를 획득하고, 신체 나머지 부분은 역 운동학(inverse kinematics)으로 추론하고 있다[1]. 하지만, 이는 전신에 대한 부정확한 추정이 발생하고, 사용자의 하반신 데이터를 획득하기 어렵기 때문에 외부 트랙커(external tracker)를 별도로 부착하거나[2], 상반신 동작에 맞춰 하반신을 시뮬레이션하는 방법이 사용된다[3]. 이는 사용자 입장에서 SoO(sense of ownership)이나 SoA(sense of agency)에 대한 경험 부족으로 가상 공간에서의 몰입도(immersion)가 떨어질 수 있다는 단점이 존재한다[4].

최근에는 딥 러닝(deep learning)에 대한 기술 발달로 3D 포즈 추정에 대한 정확도가 높아지고 있다. 초기에는 카메라 센서를 외부에 두고 사용자 영상을 획득하는 아웃사이드-인(outside-in) 방식을 통해 포즈를 추정했다면, 최근에는 스마트 안경, 착용형 카메라, HMD를 포함한 웨어러블(wearable) 기술의 발전으로 디바이스(device)의 시점(viewpoint)에서 인체 포즈를 추정하는 인사이드-인(inside-in) 방식의 자기중심(egocentric) 포즈 추정에 대한 관심이 크게 높아지고 있다. 자기중심 시점은 사용자에 부착된 센서로부터 획득한 데이터를 사용하기 때문에 사용자의 이동이 자유롭다는 장점이 있으나, 외부 카메라나 센서에 의존하는 기존 포즈 추정과 달리, 사용자의 머리 부근에서 영상을 획득하기 때문에 신체 부위가 가려진다는 어려움이 있고, 하반신의 경우 이러한 현상은 두드러진다. 또한, 다양한 카메라 각도, 높이, 방향 등 자기중심적 설정에서 가능한 시점의 범위가 넓어서 다양한 동작 추적 환경에서 정확한 포즈 추정을 보장하는 모델이 필요하다.

자기중심 시점의 영상을 얻는 것은 카메라 센서의 제한적인 화각으로 인해 쉽지 않은 문제이므로, 대부분의 연구는 어안 카메라(fisheye camera)를 이용하여 영상을 획득하고 있다. 이러한 방식은 일반적인 환경에서 사용자의 일상 활동을 캡처할 수 있다는 장점은 있으나, 왜곡된 단안 비디오(monocular video)의 모호성으로 인해 깊이(depth) 정보의 부정확이 발생할 수 있고, 이는 카메라로부터 신체 부위의 거리를 정확하게 파악하는 것을 어렵게 만든다. 또한, 카메라로부터 획득한 RGB 영상의 경우 사용자가 착용한 옷의 색상, 조명, 배경 등 실제 세계의 복잡성이 포함되어 있어 딥 러닝(deep learning)을 기반한 신경망 네트워크 방법들은 방대한 양의 데이터셋(dataset)을 요구한다. 전통적인 인체 포즈 추정을 위한 MPII[5] 및 Human3.6M[6]과 같은 벤치마크(benchmark)용 데이터셋이 존재하지만, 시점의 차이 때문에 자기중심 포즈 추정에 직접적으로 적용하기에는 어렵다. 따라서, 자기중심 시점에서 RGB 이미지 및 3D 실측 데이터(ground truth data)를 동시에 수집하는 것은 많은 시간과 노력을 요구하고, 관련 연구는 부족한 편이다.

본 논문에서는 자기중심 시점에서 입력되는 깊이 데이터(depth data)를 이용하여 실시간 3D 인체 포즈를 추정하는 방법을 소개한다. 제안하는 방법에서는 HMD에 장착된 다중 깊이 카메라로부터 자기중심 깊이 데이터를 스트림(stream) 방식으로 획득한다. 깊이 데이터의 경우 색상이나 조명의 영향을 받지 않는다는 장점이 있다. 사용자의 인체 포즈는 ResNet 기반 네트워크 (ResNet-based network)를 사용하여 3D 스켈리톤 관절 위치(skeletal joint position)를 추정하고, 그 정확도는 참조(reference) 스켈리톤 관절 위치와 비교하여 보정한다. 참조 스켈리톤 관절 데이터는 외부에 장착된 저가형 RGB-D 카메라를 사용하여 획득한다. 이를 통해 자기중심 포즈 추정을 위해 필수적으로 사용되는 방대한 크기의 데이터셋(dataset)의 크기를 크게 줄이면서 인체의 다양한 포즈를 추적할 수 있다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 가상현실 환경에서의 사용자 동작 추적 및 포즈 추정에 관한 관련 연구를 분석한다. 3장에서는 제안하는 자기중심 시점에서의 3D 인체 포즈 추정을 위한 방법의 개요를 소개한다. 4장과 5장에서는 자기중심 데이터 획득 방법 및 ResNet 기반 네트워크를 사용한 인체 포즈 추정에 대해서 각각 설명한다. 6장에서는 사용자 실험 결과를 보여주고, 마지막으로 7장에서는 본 논문의 결론을 내린 후 향후 계획을 소개하고 마친다.

## Ⅱ. 관련 연구

가상현실용 HMD 장치가 점점 널리 보급됨에 따라 HMD와 콘트롤러를 이용하여 아바타(avatar)를 제어하는 방법도 활발히 연구되고 있다. Dittadi 등은 VAE(variational autoencoder)를 사용하여 노이즈(noise)가 많은 머리와 손 데이터에서 포즈 추정을 하는 방법을 제안하였다[7]. 이를 위해 special inference model을 학습시켜 제한된 정보로 전신 포즈를 예측하는 문제를 해결하였다. Aliakbarian 등은 FLAG(flow-based avatar generative model)을 사용하여 전신 포즈를 예측하였고, 관절의 불확실성 추정치를 제공하였다[8]. Jiang 등은 transformer architecture를 사용하여 HMD와 콘트롤러 포즈를 입력으로 글로벌(global) 스켈리톤 상태를 예측하였다[9]. 그러나, 이러한 방식들은 물리적 제약을 적용하지 않은 운동학적(kinematics)에 의존한 아바타 움직임 제어에 초점이 맞추어져 있어, 본 논문에서 제안하는 신

경망 네트워크를 사용한 사용자 자세 추적과는 다르게 발 미끄러짐(foot-skating)이나 지터링(jittering)과 같은 동작 노이즈 문제가 발생할 수 있다. Meta에서는 HMD와 두 개의 콘트롤러에서 신호를 받아와 물리적으로 유효한 전신 동작을 시뮬레이션하는 reinforcement learning framework를 제시하였다[3]. 하지만, 사용자가 양손에 콘트롤러를 항상 들고 있어야 하므로 움직임의 제약이 따르는 문제가 있어 가상훈련과 같은 분야에는 적용이 어렵다.

최근에는 자기중심 시점에서 전신 동작을 캡쳐(capture)하는 연구가 주목받고 있지만, 카메라 시점에서 전신 바디를 추적하기는 어려운 문제이다. 초기에는 가슴이나 머리 부분에 부착된 카메라 센서로부터 볼 수 있는 손, 팔 등 일부 관절만 복원하여 객체와 상호작용할 수 있는 방법을 제시하였다[10]-[12]. EgoCap은 헬멧에 부착된 스테레오 어안 카메라(stereo fisheye camera)를 이용한 전신 촬영 방법을 제안하였다[13]. 이 연구에서는 어안 뷰(fisheye view)를 위한 새로운 포즈 추정 프레임워크(framework)에 대규모 데이터셋에서 훈련된 ConvNet 기반 신체 부위 감지기를 결합하여 전신 포즈를 추정하였고, 어안 이미지의 배경, 의상 색상, 조명 변화로 자기중심 전용 데이터셋을 생성하였다. Xu 등은 머리에 장착한 단일 어안 카메라를 사용하여 자기중심 시점에서 3D 포즈를 추정하는 방법을 제안하였다[14]. 그들은 3D 포즈 데이터를 얻기 위해 surreal 데이터셋[15]을 기반으로 자기중심 어안 시점에서 합성된 데이터를 생성하였고, 저해상도의 하체 관절을 예측하기 위한 two-stream CNN 기법을 도입하였으며, location-sensitive distance module을 사용하여 실제 관절 위치을 복원하였다. Tome 등은 HMD에 부착된 카메라를 통해 얻은 단안의 이미지에서 가려짐(occlusion)으로 인한 높은 불확실성을 모델링하기 위한 전신 및 하체의 two-stream architecture를 제안하였다[16],[17]. 그들의 연구는 상체와 하체 사이의 해상도 차이를 해결하기 위해 dual branch decoder를 통해 2D 관절 위치의 불확실성을 처리하였다. 학습 데이터셋으로는 동작 캡쳐 데이터(motion capture data)와 물리 기반 렌더링(physics-based rendering) 설정으로 다양한 배경과 조명에서 피부색, 체형, 의상 스타일, 체형 등을 무작위로 합성하하였다. Zhao 등은 스케리오 카메라가 부착된 안경 프레임을 이용한 자기중심 동작 캡쳐 및 포즈 추정 방법을 제안하였다[18]. 여기서, 자체 가려짐(self-occlusion) 문제를 해결하기 위해 기존 2개의 branch로 구성된 인체 포즈 감지 방법에 바디 파트 정보를 추가하여 스켈리톤 구성(configuration)을 개선하였다. Wang 등은 광시야각 자기중심 어안 카메라를 사용하여 인체 뒤의 장면 깊이 맵(map)을 예측하기 위한 자기중심 깊이 추정 네트워크를 제안하였다[19],[20]. 이를 통해 자기중심 비디오에서 정확하고 안정적인 글로벌 인체 포즈를 추정하는 방법을 제안하였고, CNN으로 감지한 2D 및 3D 키포인트(key point), VAE-based motion priors, SLAM-based camera pose 추정을 활용하여 시간적 흔들림이나 추적 실패와 같은 문제를 해결하였다. Hwang 등은 VQ-VAE(vector quantized-variational autoencoder)를 사용하여 자기중심 이미지에서 인체 포즈를 예측하고, 인식한 포즈를 최적화하는 추정 파이프라인(estimation pipeline)을 통해 하체가 작게 나타나고 상체에 의해 가려지는 문제를 해결하였다[21]. Jiang 등은 dynamic motion signatures와 static scene structures를 사용하여 자기중심 비디오에서 보이지 않는 인체 포즈를 추론하는 방법을 제안하였다[22]. 여기서, 그들은 단기(short-term) 및 장기(long-term) 포즈 운동학(dynamics)을 결합하여 자세 추정 확률을 추정하고, 긴 스퀀스(sequence)에 대해 관절 추론을 수행하기 위한 분류기(classifier)를 제안하였다. 나중에, 그들은 기하학적인 일관성을 보장하는 자세 추정을 위해 카메라 SLAM의 다이나믹 동작과 보이는 인체의 부분 정보를 사용하였다[23].

자기중심적 설정에서 신경망 모델의 훈련을 위해 실사 데이터를 얻는 것은 많은 시간과 노력을 요구하는 작업이므로 학습을 위해선 대부분 합성 데이터를 사용한다. 최근에는 합성 데이터셋을 통합하고, domain adaptation 기법을 적용하여 합성과 실사 데이터 사이의 차이를 완화하는 방법이 제안되었다[24]. Zhang 등은 어안 렌즈로 인한 이미지 왜곡을 자동 카메라 캘리브레이션(automatic camera calibration)으로 완화시키는 방법을 제안하였다[25]. 하지만, 기존 대부분의 RGB 이미지에 기반한 신경망 모델들은 방대한 양(수십만에서 수백만 개의 이미지)의 데이터셋을 요구하고 있으나, 본 논문에서는 사용자의 깊이 정보를 활용하여 그 크기를 크게 줄이는 방법을 제안한다.

## Ⅲ. 개 요

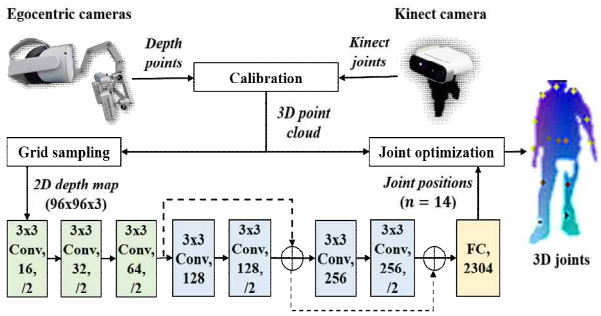


**그림 1.** 제안하는 시스템의 전체 구성도
**Fig. 1.** Overview of the proposed system

그림 1은 본 논문에서 제안하는 시스템의 전체 구조를 나타낸다. 우선, HMD에 부착한 다중 RGB-D 센서에서 획득한 다양한 시점의 깊이 정보 데이터를 단일 좌표계 시스템(coordinate system)으로 통합하여 포인트 클라우드(point cloud) 형태로 생성한다. 다음은 3D 포인트 클라우드 데이터를 RGB-D 카메라의 2D 좌표계 시스템으로 투영(project)시킨 후, 사용자의 스켈리톤 조인트(skeletal joint)를 추정하기

위해 깊이 기반 신경망 네트워크(depth-based neural network)를 학습시킨 후 입력된 사용자의 깊이 영상에서 3D 관절 위치를 추정한다.

## Ⅳ. 데이터 획득

### 4-1 다중 RGB-D 센서 구성

제안하는 시스템은 다중 RGB-D 센서[26]들에서 스트림으로 받은 깊이 정보 데이터를 사용하여 사용자의 3D 자세를 실시간 추정하는 것을 목표로 한다. 상업용 RGB-D 센서는 아웃사이드-인 방식으로 영상을 촬영하는 것을 목적으로 제작되었기 때문에 시야각(FOV; field of view)이 크지 않으며, 사용자 머리 앞에 센서를 위치하면 전체 바디(full body)의 데이터를 획득하기 어렵다는 문제가 있다. 따라서, 제안하는 시스템에서는 그림 2와 같이 다중 RGB-D 센서들을 배치하여 사용자 머리 위치에서도 전체 바디의 깊이 데이터를 획득할 수 있도록 구성하였다. 부착된 3개의 센서 중 중간(middle)은 메인으로 센서 좌표계(sensor coordinate)의 기준이 되고, 가로 방향으로 설치하여 목부터 배 높이까지 촬영할 수 있다. 다른 2개의 서브 센서는 메인 센서의 좌우로 세로 방향으로 설치하여 각각 가슴 아래 왼쪽과 오른쪽을 촬영할 수 있다.


**그림 2.** HMD에 부착한 다중 RGB-D 센서 구성도
**Fig. 2.** Configuration of multi RGB-D sensors attached to HMD

### 4-2 다중 센서 캘리브레이션

제안하는 시스템은 3개의 다른 시점에서 촬영된 깊이 데이터를 사용하기 때문에 동일한 좌표계로 데이터를 통합(unification)할 필요가 있다. 이를 위해 메인과 나머지 서브 센서들에서 촬영된 영상이 겹치는 부분에 체커 보드(4×8 checkers)를 사용해 서로 다른 위치에서의 영상을 반복 촬영하면서 각 센서로부터 깊이 데이터를 획득한다. 영상에서 체커 보드에 대해 corner detection을 진행하면, 2D corner 위치에 대해 깊이 데이터의 3D 위치를 가져올 수 있다. 이때, 2D 픽셀(pixel) 하나에 대응하는 3D 깊이 값은 노이즈(noise)가 있을 수 있기 때문에 주변 깊이 데이터의 평균점을 계산하여 사용한다. 각 영상에서 추출한 $3N(N=32$, 3회 촬영)개의 3D 위치는 ICP(iterative closest point) 알고리즘[27]을 이용하여 회전(rotation) 및 이동(translation) 행렬(matrix)을 계산하여 서브 센서에서 획득한 깊이 데이터의 좌표계를 메인 센서에서 획득한 데이터가 사용하는 좌표계로 정렬(align)한다. 그림 3은 서브 영상에서 얻은 좌우 서브 센서에서 획득한 깊이 데이터를 메인의 좌표계에 통합한 결과를 보여준다.

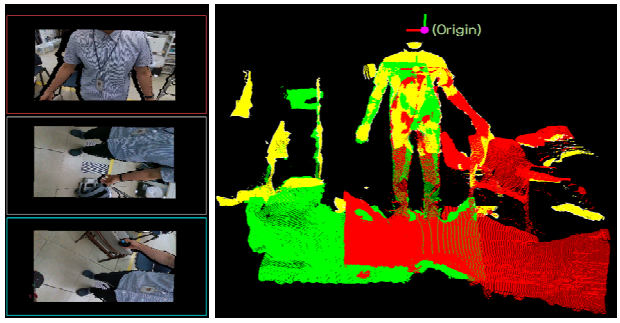


**그림 3.** 다중 센서 캘리브레이션: sub1(빨간색)과 sub2(녹색)에서 획득한 깊이 데이터가 메인(노란색)의 좌표계 시스템에 통합된 결과
**Fig. 3.** Calibration of multi-sensors: result of alignment of depth data captured at sub1 (red) and sub2 (green) sensors into the main (yellow) coordinate system

### 4-3 트레이닝 데이터 생성

본 논문에서 3D 자세 추정 방법은 신경망 네트워크를 사용하기 때문에 학습과 테스트용 ground truth 데이터가 필요하다. Human3.6M[6]과 같이 대규모 자세 데이터셋이 공개적으로 존재하지만, 자기중심 시점에서 획득한 데이터가 아니기 때문에 제안하고 있는 시스템에서 사용하기는 어렵다. 또한, 새로운 데이터셋을 생성하는 작업은 방대한 크기의 이미지 레이블링(image labeling)과 같이 오랜 시간과 노력을 요구하기 때문에 제안하는 시스템에서는 Microsoft사의 Azure Kinect[28]를 외부 동작 센서(motion sensor)로 추가하여 사용한다. 이를 통해 추적하는 데 있어 더욱 효율적으로 데이터 획득(data acquisition)이 가능하다.

Kinect 센서는 깊이와 RGB 영상에서 배경과 신체를 분할하고, RGB 영상에서 CNN 모델을 통해 2D 관절 위치를 감지한 후 깊이 데이터를 사용하여 3D 관절 위치를 획득하는 방식이며, 상용 동작 센서로는 비교적 좋은 성능을 가지고 있다. 그러나, Kinect 관절 데이터는 관절의 위치, 관절 간 길이가 일정하지 않기 때문에 ground truth로 활용하기에는 문제가 있다. 따라서, 제안하는 시스템에서는 학습을 위한 참조 모델(reference model)로 사용하고, 최종 관절 위치는 보정을 통해 결정한다. 즉, HMD에 설치된 다중 RealSense 센서에서 사용자의 깊이 데이트를 획득하면서, 동시에 외부에 설치된 Kinect 센서에서는 사용자의 관절 위치를 추적한다.

Kinect 센서 위치는 고정되어 있지만, RealSense 센서는 사용자와 함께 움직이고 있기 때문에 사용자 깊이 데이터와 Kinect 스켈리톤 사이의 좌표를 맞추는 과정이 매번 필요하다. 이를 위해 ICP 알고리즘[27]을 적용하는데, 포인트 데이터가 많으면 계산 시간이 길어질 뿐 아니라 RealSense 센서의 깊이 데이터에는 노이즈가 많기 때문에 데이터를 샘플링(sampling)하여 저장하는 방식을 사용한다. 그림 4와 같이 사용자 깊이 데이터는 $dx \times dy$ 크기를 갖는 박스(box)로 이루어진 2D 그리드(grid)에 투영하여 박스 내에 입력된 포인트들의 평균 값을 사용한다. 그리드 사이즈는 사용자의 높이(height)에 비례하며, $dx \times dy$ 값을 조절하여 생성한다. 이때, 사용자 머리를 중심으로 사용자 포인트만 맵에 저장한다. Kinect 센서에서 입력된 스켈리톤에서는 학습에 필요한 14개의 관절 위치(naval, neck, shoulders, elbows, wrists, hip, knees, ankles)만 저장하며, 데이터 획득 및 저장 시간을 줄이기 위해 Kinect 깊이 데이터는 사용하지 않는다. 대신, 부모-자식 관절을 로컬 $y$축으로 위치시키고, 로컬 $x$축을 계산하여 로컬 $x-y$ 평면을 기준으로 주변에 16개 이웃(neighbor) 포인트를 사용한다. 포인트 간 거리는 그리드 내 박스 사이즈$(dx, dy)$를 기준으로 1~2배 거리로 설정한다. 각 RealSense 센서에서 나오는 포인트 간 매칭을 통해 ICP 알고리즘으로 계산하여 좌표계를 맞춘 데이터를 ground truth로 사용한다. 제안하는 방법은 깊이 데이터 기반으로 학습을 위한 데이터를 생성 하기 때문에 색상이나 조명 등에 영향을 받지 않고 데이터를 획득할 수 있다는 장점이 있다.

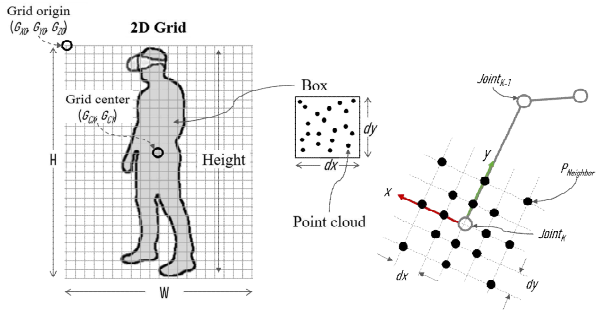


**그림 4.** $t$ 프레임에서 트레이닝 데이터 생성
**Fig. 4.** Generation of training data at $t$ frame

### 4-4 데이터 필터링

2가지 RGB-D 센서로부터 매 프레임 캘리브레이션을 수행하면 적은 수의 샘플 포인트로 매칭 관계를 빠르게 계산할 수 있으나, 매칭 에러로 인해 잘못된 결과가 나올 수 있다. 이는 신경망 모델 훈련에 큰 문제를 발생시킨다. 따라서, 훈련에 의해 생성된 학습 모델에 데이터를 다시 입력하고 결과를 출력하여 RMSE(ground truth 값과 root mean square error)를 계산한다. 거리 오차가 정해진 값 이상이면, 캘리브레이션에 의한 ground truth 계산이 잘못된 것으로 판단하고, 데이터를 버린다. 새로운 데이터를 추가로 입력하면서 오차가 큰 데이터를 삭제하는 과정을 반복하면 최종적으로는 오차 범위 이내로 들어오는 훈련 데이터만 남게 되며, 이에 따라 그림 5와 같이 학습 결과를 좋게 하는 효과를 가져올 수 있다.

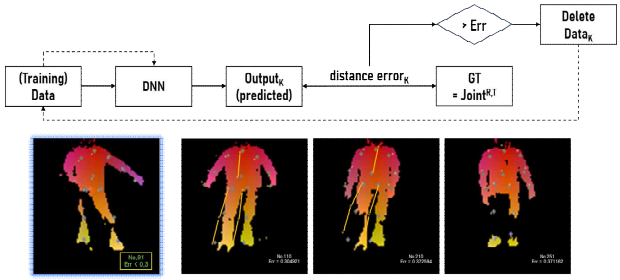


**그림 5.** 트레이닝 데이터를 위한 필터링 프로세스 (위), 오차 범위 내의 트레이닝 데이터 (아래 왼쪽), 오차 범위 밖의 트레이닝 데이터 (아래 오른쪽)
**Fig. 5.** Filtering process for training data (top), training data within an error (bottom left), and training data outside an error (bottom right)

## V. 자세 추정

본 논문에서 사용자의 자세는 신경망 네트워크를 사용하여 3D 관절 위치로 추정된다. 기존 RGB 영상을 사용하는 네트워크 모델과 다르게 제안하는 시스템에서는 인체 포즈의 포인트 클라우드를 특징 맵(feature map)으로 변환하여 신경망 네트워크의 학습 데이터로 사용한다. 이를 위해 지면 탐지(ground detection) 및 포인트 클라우드를 샘플링하는 단계를 거친 후 앞 장에서 설명한 2D 그리드로 투영하여 특징 맵을 생성한다.

### 5-1 지면 탐지

사용자 자세 추정에서 깊이 데이터를 사용하는 경우 지면 데이터는 노이즈로 분류되기 때문에 이를 제거해 주는 것이 필요하다. 그러나, 사용자가 부착된 센서는 이동하며 움직이기 때문에 고정형 센서와 달리 지면에 대한 정보를 미리 알 수 없다. 따라서, 매 프레임 입력되는 포인트 클라우드로부터 지면 영역을 탐색하는 과정이 필요한다.

그림 6와 같이 지면 탐색 방법은 사용자 센서 $y$축을 기준으로 포인트 클라우드 데이터를 하단 그리드 영역으로 투영한 후 포인트 평균값이 가장 낮은 지점(lowest value)의 영역을 선정하여 포인트 queue에 넣는다. Queue에서 포인트를 하나씩 꺼내면서 영역 주변 평균 포인트들을 탐색하여 범위 내 높이($R_h$)에 있는 영역은 다시 queue에 넣고, queue가 비워질 때까지 반복하여 계산한다. Queue에 남은 영역이 없으면, 큐에 있었던 영역들의 평균 포인트들을 모아 주성분 분석(PCA; principal component analysis) 알고리즘을 이용하여 지면의 위치와 좌표축을 계산한다. 영역의 낮은 값과 주변 값들을 참조하여 계산하므로 빠르게 지면 위치와 로컬 축

을 찾을 수 있다. 지면을 찾으면 지면 로컬 $y$축 범위($r_h$) 내에 있는 포인트들은 제거하고 사용자 영역에 있는 데이터만 사용한다.

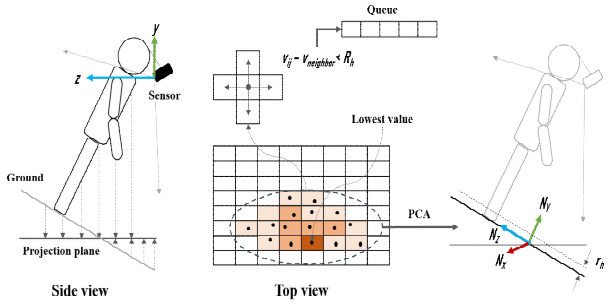


**그림 6.** 특징 맵 생성을 위한 지면 탐지
**Fig. 6.** Ground detection for generating a feature map

### 5-2 특징 맵 생성

본 논문에서 신경망 네트워크 학습에 필요한 특징 맵은 포인트 클라우드를 기반으로 생성된다. 메인 센서 좌표계를 기준으로 포인트 클라우드는 2D 그리드에 투영되어 샘플링(sampling)되는데, 이는 노이즈를 제거하고, 포인트 데이터 크기를 줄이는 효과가 있다. 그리드 크기는 사용자 키(height)에 비례하여 세로 길이 $H$와 가로 길이 $W$가 결정되며, 그리드의 원점은 왼쪽 상단에서 시작된다(그림 4 참조).

신경망 네트워크에 입력되는 특징 맵 크기는 $96\times96\times3$이므로 이를 맞추기 위해 그리드 내 박스 사이즈를 조절하여 샘플링 한다. 이는 사용자 데이터를 정규화(normalization)하는 효과를 줄 수 있어 다양한 신체 사이즈를 갖는 사람에 대한 데이터가 적어도 자세 추정이 가능하다. 그리드 데이터는 포인트 클라우드를 샘플링한 데이터로 사용자의 자세를 나타내고, 자세에서 손과 발의 움직임이 중요하기 때문에 다음과 같은 특징을 반영한 특징 맵으로 변환한다.

첫째, 그리드의 중심점을 기준으로 좌, 우로 멀어질수록 큰 값이 된다. 둘째, 하반신은 가려짐이 많이 발생할 수 있으므로, 위에서 아래로 내려갈수록 큰 값이 된다. 셋째, 사용자 머리 부분에서 데이터를 획득하므로 앞으로 나올수록 큰 값이 된다. 단, 센서 좌표계를 기준으로 계산하면 머리 위치가 가장 가까이 나타나고, 발이 멀리 있는 것으로 나타나므로 여기서는 지면 로컬 $z$축을 기준으로 계산한다. 넷째, 학습 모델에 입력값으로 사용하기 위해 그리드 데이터, $(f_x, f_y, f_z)$를 다음과 같이 정규화한다.

$$f_x = \frac{2|c_x - b_x|}{W}, f_y = \frac{b_y}{H}, f_z = \frac{D-(b_z - N_z))}{D}. \quad (1)$$

여기서, $c_x$는 그리드의 $x$축 중심점, $(b_x, b_y, b_z)$는 그리드 원점을 기준으로 박스 내 3D 포인트 대표 값, $W, H, D$는 그리드의 볼륨(volumne) 크기, $N_z$는 지면 로컬 $z$축의 노멀 벡터(normal vector)를 의미한다(그림 6 참조).

### 5-3 3D 자세 위치 추정

제아하는 시스템에서 특징 맵을 입력으로 받는 신경망 네트워크 구조는 ResNet architecture[29]를 기반으로 구성하였으며, 3계층의 convolution layer와 2계층 residual block, 1개의 fully connected 계층으로 설계하였다. RGB 영상과 달리 포인트 클라우드에 의해 생성된 특징 맵은 사용자 데이터만 추출하였고, 크기가 크지 않기 때문에 깊은 층의 사용이 필수적이지는 않다. 따라서, 학습에 필요한 계산량 및 모델 크기도 줄일 수 있다는 장점이 있다. 하나의 신경망 모델은 하나의 3D 관절 위치와 연결되어 최종 14개 신경망 모델이 학습되며, 학습 결과는 14개 관절의 3차원 위치가 그림 7과 같이 생성된다.

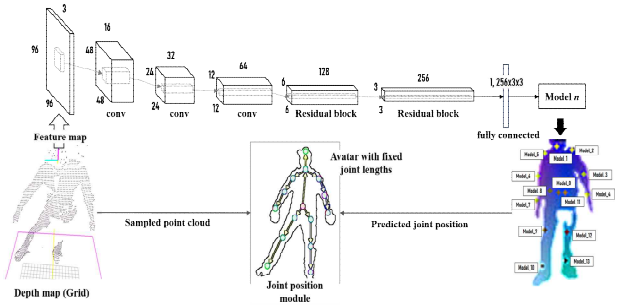


**그림 7.** 특징 맵을 사용한 ResNet 기반 3D 관절 위치 추정
**Fig. 7.** ResNet-based joint estimation using a feature map

학습 모델에 의한 추정 위치는 참조 데이터로 사용하며, 이후 깊이 데이터와 아바타 스켈리톤(avatar skeleton)을 이용하여 다음과 같이 정확한 위치를 보정한다. 추정된 관절 위치의 상하좌우로 그리드 내 박스 크기만큼 떨어진 위치에 가상의 포인트를 두고, 5개 포인트(추정된 위치 및 주변 4개 가상 포인트)로부터 가장 가까운 포인트 클라우드 위치를 찾아간다. 이 방법은 간략화된 ICP 방식으로, 1~3회 이내의 반복으로 빠르게 포인트 클라우드 위치로 이동되므로 예측된 관절 위치, $P_{pd}^k$에서 더 정확하게 실제 위치, $P_{real}^k$로 이동시킬 수 있다. 이때, 가려짐으로 인해 근처 포인트 클라우드가 없는 경우에는 $P_{pd}^k$를 그대로 사용한다. 앞에서 설명한 바와 같이 Kinect 센서의 관절 데이터는 위치와 길이가 변할 수 있기 때문에 예측된 관절 위치는 참조 위치로 간주하고, 고정 길이를 가진 아바타 스켈리톤과 클라우드 데이터의 샘플 포인트를 이용하여 최종 위치를 다음과 같이 결정한다.

$$\min \sum_{k=1}^{14} \left| R^{k-1} \cdot P_{av}^k - P_{real}^k \right|^2 \quad (2)$$

여기서, $k$와 $k-1$는 각각 현재와 부모 관절이고, 목표는 깊이 데이터에 의해 수정된 $P_{real}^k$와 아바타 관절 위치인 $P_{av}^k$ 사이의 차이를 최소화하는 회전 값, $R^{k-1}$을 찾는 것이다. 제안하는 방법에서는 관절 각도 값을 조절하여 최종 위치를 결정하기 때문에 관절의 길이는 변하지 않으며, 실제 데이터인 포인터

클라우드를 기반으로 변환된 결과에 맞추기 때문에 보다 정확한 관절 위치를 얻을 수 있다.

## Ⅵ. 실험 결과

### 6-1 실험 환경

본 논문의 실험은 Windows 11 환경에서 Intel i7 3.0GHz CPU, 32GB DDR4 메모리, NVIDIA GeForce GTX 4090 GPU로 수행되었다. RGB-D 센서로는 최소 인식 거리(30cm), 깊이 시야각(H: 87°, V: 58°), 무게(180g), 크기(L: 90cm, D: 25cm, H: 25cm), 가격 등을 고려해 Intel RealSense D435i[26]를 사용하였으며, 그림 2에서 나타내는 것과 같이 다중 센서들을 HMD에 장착한 후 센서가 바라보는 각도 및 HMD와의 거리를 조절할 수 있도록 알류미늄 프레임으로 구성(총 무게는 880g)하였다.

훈련 데이터셋으로 사용자의 다양한 동작(걷기, 뛰기, 앉기, 서기, 발차기 등)에서 6,000프레임(200s)을 생성하였고, 성능 평가를 위해 서로 다른 키, 의상, 배경을 가진 사용자들 10명의 스켈리톤 자세를 30fps로 검출하였다. 이때, Kinect 센서는 30fps(33.3ms)로 작동하고, RealSense 센서는 60fps(16.66ms)로 작동하기 때문에 여기서는 더 낮은 속도인 Kinect에 맞춰서 데이터를 획득하였다. Kinect 관절이 획득되는 시점에서 RealSense의 깊이 데이터를 저장하면 동기화 오차가 최대 약 8.33ms 수준이며, 일반적인 동작에서는 큰 무리가 없는 정도의 오차라고 할 수 있다.

### 6-2 사용자 자세 추정

그림 8은 제안하고 있는 시스템으로 걷는 동작(3,120프레임)에서 검출된 3D 관절 위치를 보여준다. 보행 동작에서 검출된 관절과 Kinect 관절 간의 유클리드(Euclean) 거리 기준 관절 위치 평균 오차(MPJPE; mean per joint position error)는 약 30mm로 측정되었다. 특히, 발목 관절은 평균 45mm로 가장 큰 오차를 보였는데, 이는 가림 현상과 자기중심 시점으로부터의 먼 거리로 인해 검출이 가장 어려운 부위임을 시사한다. 그에 반해 손목 관절은 상대적으로 가까운 위치로 인해 평균 약 38mm의 오차를 보여주었다. 그림 8에서 보여주듯이 인체의 주요 관절들(head, shoulders, elbows, wrists, root, hips, knees, ankles)은 모두 성공적으로 검출되었고, 자기중심 시점에서 가까울수록 더 작은 오차(12~45mm)를 보여주었다.

다양한 동작에 대한 관절 위치 평균 오차 (MPJPE) 및 다중 센서들이 부착된 HMD의 착용감에 관한 결과는 표 1에 나타내었다. 관절 위치 평균 오차 측정을 위해서 다중 아웃사이드-인 방식의 RGB-D 센서들로 구성된 시스템[30]을 사용하였고, 착용감 평가를 위해서 참여한 사용자들에게 기존 HMD 대비 추가된 장치의 무거움을 설문 조사(4점 리커트 척도 사용: 0-차이 없음, 1-약간 무거워짐, 2: 다소 무거워짐, 3: 매우 무거워짐)하였다. 표 1에서 나타내는 것과 같이 자기중심 시점의 특성상 신체적 가림이 많이 일어나는 역동적인 동작일수록 평균 오차가 증가하는 것을 관찰할 수 있었다. 착용감은 동작의 범위가 넓을수록 상대적으로 무겁게 느껴진다는 점도 관찰할 수 있었다. 그림 9는 표 1의 동작들에 대해 제안하는 시스템과 다중 Kinect 기반 시스템[30]에서 추정한 스켈리톤의 위치를 비교한 결과이다.

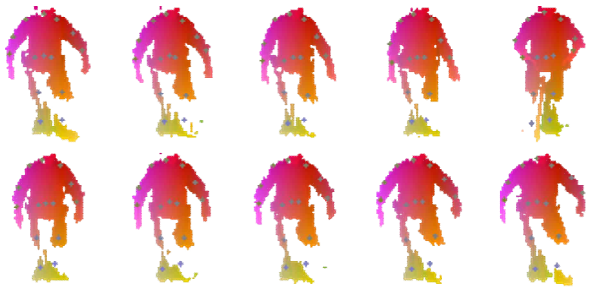

**그림 8.** 걷는 동작에서의 3D 관절 (청색) 검출: 검출 중요도에 따라 색상화된 깊이 맵 (빨강에서 노랑: 낮음에서 높음)

**Fig. 8.** Detection of 3D joints (cyan) on walking motion: A depth map colored based on the detection importance (red to yellow: less to more)

**표 1.** 다양한 동작에 대한 관절 위치 평균 오차 및 무거워짐 평가 (0: 차이 없음, 1: 약간, 2: 다소, 3: 매우)

**Table 1.** Mean per joint pisition errors and evaluation of heaviness (0: no difference, 1: little, 2: somewhat, 3: much) for various motions

| Motion Style | Motion Length (fps) | Average Errors (mm) | Heaviness (0~3) |
|---|---|---|---|
| Walking | 3,120 | 30 | 1.6 |
| Jogging | 1,815 | 32 | 1.9 |
| Running | 1,432 | 38 | 2.1 |
| Sitting | 916 | 34 | 1.1 |
| Standing | 884 | 33 | 1.3 |
| Punching | 1,106 | 41 | 1.9 |
| Kicking | 798 | 43 | 2.3 |

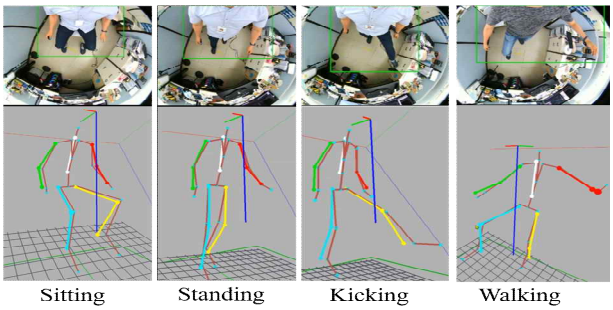


**그림 9.** 입력 동작으로부터 추정된 스켈리톤과 Kinect (빨간색) 스켈리톤의 비교

**Fig. 9.** Comparison between the estimated and the Kinect (red) skeleton from input motions

## Ⅶ. 결 론

본 논문에서는 기존에 널리 보급된 저가형 상용 RGB-D 카메라들을 HMD에 부착하여 자기중심 시점에서 3D 사용자 자세 추정을 하는 방법을 제안하였다. 실험 결과 사용자 동작 추적에 공간적 제약이 있는 다중 아웃사이드-인 센서들을 사용한 시스템[30]과 비교할 만한 자세 추정이 가능하였다. 특히, 자기중심 시점에서 획득된 불규칙한 3D 포인트 클라우드에서 직접적으로 사용자의 관절 위치를 추정할 수 있는 3D 자세 리그레션(regression) 네트워크 구조를 통해 자기 몸에 의해 가려짐 혹은 노이즈가 있는 환경에서도 안정적인 자세 추정이 가능하였다. 따라서, 제안하고 있는 자기중심 시점 동작 추적은 기존 HMD를 사용하는 가상현실 시스템의 다중 트랙커 센서들의 위치 최적화 및 시야각의 음영 문제[2],[30]에서 비교적 자유로워 사용자와 주위 환경과의 상호작용이 중요한 가상훈련과 같은 몰입형 콘텐츠에서 그 활용도가 높을 것으로 예상한다. 또한, 옷 색상, 조명, 배경 등에 영향을 받는 RGB 영상 기반 자기중심 시점 추적 방식들 대비 깊이 데이터를 활용하는 것은 신경망 네트워크에 필요한 학습 데이터의 제작 과정의 노력과 시간을 크게 단축할 가능성을 제시하였다.

현재 HMD에 부착한 다중 센서들 및 프레임의 무거움은 실험에 사용한 전체 동작들에서 1.1~2.3(전체 사용자들의 평균, 표 1 참조)으로 평가되었다. 이는 대부분의 사용자가 역동적인 동작일수록 HMD에 추가된 무게감을 느끼고 있는 것으로 판단된다. 이를 개선하기 위해서 사용한 다중 RGB-D 센서들을 초소형 스테레오 어안형 센서로 대처하고, 깊이 맵 정보를 RGB 이미지에서 복원하는 방법을 사용하여 HMD에 추가되는 무게를 최소화할 예정이다. 또한, 자기중심 시점에서 멀리 떨어진 발목(ankle)과 같은 관절 위치의 정확도를 개선하기 위해 MLP(multi-layer perceptron) 기반 신경망 네트워크 모델을 후처리(post-processing)로 활용할 계획이다.

## 감사의 글

This work was supported by Electronics and Telecommunications Research Institute (ETRI) grant funded by the Korean government [24ZC1200, Research on hyper-realistic interaction technology for five senses and emotional experience].

## 참고문헌

[1] DeepMotion. How to Make 3 Point Tracked Full-Body Avatars in VR [Internet]. Available: https://deepmotion.medium.com/how-to-make-3-point-tracked-full-body-avatars-in-vr-34b3f6709782.

[2] S. Merker, S. Pastel, D. Bürger, A. Schwadtke, and K. Witte, “Measurement Accuracy of the HTC VIVE Tracker 3.0 Compared to Vicon System for Generating Valid Positional Feedback in Virtual Reality,” *Sensors*, Vol. 23, No. 17, 7371, August 2023. https://doi.org/10.3390/s23177371

[3] A. Winkler, J. Won, and Y. Ye, “QuestSim: Human Motion Tracking from Sparse Sensors with Simulated Avatars,” in *Proceedings of SA 22: SIGGRAPH ASia 2022 Conference Papers*, Daegu, pp. 1-8, 2022. https://doi.org/10.1145/3550469.3555411

[4] N. Braun, S. Debener, N. Spychala, E. Bongartz, P. Sörös, H. H. O. Müller, and A. Philipsen, “The Senses of Agency and Ownership: A Review,” *Front in Psychol*, Vol. 9, April 2018. https://doi.org/10.3389/fpsyg.2018.00535

[5] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, “2D Human Pose Estimation: New Benchmark and State of the Art Analysis,” in *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, Columbus: OH, pp. 3686-3693, 2014. https://doi.org/10.1109/CVPR.2014.471

[6] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, “Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 36, No. 7, pp. 1325-1339, July 2014. https://doi.org/10.1109/TPAMI.2013.248

[7] A. Dittadi, S. Dziadzio, D. Cosker, B. Lundell, T. Cashman, and J. Shotton, “Full-Body Motion from a Single Head-Mounted Device: Generating SMPL Poses from Partial Observations,” in *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal: Canada, pp. 11667-11677, 2021. https://doi.org/10.1109/ICCV48922.2021.01148

[8] S. Aliakbarian, P. Cameron, F. Bogo, A. Fitzgibbon, and T. J. Cashman, “FLAG: Flow-based 3D Avatar Generation from Sparse Observations,” in *Proceedings of 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans: LA, pp. 13243-13252, 2022. https://doi.org/10.1109/CVPR52688.2022.01290

[9] J. Jiang, P. Streli, H. Qiu, A. Fender, L. Laich, P. Snape, and C. Holz, “AvatarPoser: Articulated Full-Body Pose Tracking from Sparse Motion Sensing, ” in *Proceedings of the 17th European Conference on Computer Vision (ECCV)*, Tel Aviv: Israel, pp. 443-460, 2022. https://doi.org/10.1007/978-3-031-20065-6_26

[10] G. Rogez, J. S. Supančič, and D. Ramanan, “First-Person Pose Recognition Using Egocentric Workspaces,” in *Proceedings of 2015 IEEE Conference on Computer Vision*

*and Pattern Recognition (CVPR),* Boston: MA, pp. 4325-4333, 2015. https://doi.org/10.1109/CVPR.2015.7299061

[11] H. Yonemoto, K. Murasaki, T. Osawa, K. Sudo, J. Shimamura, and Y. Taniguchi, "Egocentric articulated pose tracking for action recognition," in *Proceedings of 2015 14th IAPR International Conference on Machine Vision Applications (MVA)*, Tokyo: Japan, pp. 98-101, 2015. https://doi.org/10.1109/MVA.2015.7153142

[12] X. Li, Y. Hou, P. Wang, Z. Gao, M. Xu, and W. Li, "Trear: Transformer-Based RGB-D Egocentric Action Recognition," *IEEE Transactions on Cognitive and Developmental Systems*, Vol. 14, No. 1, pp. 246-252, March 2022. https://doi.org/10.1109/TCDS.2020.3048883

[13] H. Rhodin, C. Richardt, D. Casas, E. Insafutdinov, M. Shafiei, H.-P. Seidel, ... and C. Theobalt, "Egocap: Egocentric Marker-Less Motion Capture with Two Fisheye Cameras," *ACM Transactions on Graphics*, Vol. 35, No. 6, 162, December 2016. https://doi.org/10.1145/2980179.2980235

[14] W. Xu, A. Chatterjee, M. Zollhofer, H. Rhodin, P. Fua, H.-P. Seidel, and C. Theobalt, "Mo2Cap2: Real-time Mobile 3D Motion Capture with a Cap-mounted Fisheye Camera," *IEEE Transactions on Visualization and Computer Graphics*, Vol. 25, No. 5, pp. 2093-2101, May 2019, https://doi.org/10.1109/TVCG.2019.2898650

[15] G. Varol, J. Romero, X. Martin, N. Mahmood, M. J. Black, I. Laptev, and C. Schmid, "Learning from Synthetic Humans," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu: HI, pp. 4627-4635, 2017. https://doi.org/10.1109/CVPR.2017.492

[16] D. Tome, P. Peluse, L. Agapito, and H. Badino, "xR-EgoPose: Egocentric 3D Human Pose From an HMD Camera," in *Proceedings of 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, pp. 7727-7737, 2019. https://doi.org/10.1109/ICCV.2019.00782

[17] D. Tome, T. Alldieck, P. Peluse, G. Pons-Moll, L. Agapito, H. Badino, and F. de la Torre, "SelfPose: 3D Egocentric Pose Estimation From a Headset Mounted Camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 45, No. 6, pp. 6794-6806, June 2023. https://doi.org/10.1109/TPAMI.2020.3029700

[18] D. Zhao, Z. Wei, J. Mahmud, and J.-M. Frahm, "EgoGlass: Egocentric-View Human Pose Estimation From an Eyeglass Frame," in *Proceedings of 2021 International Conference on 3D Vision (3DV)*, London: UK, pp. 32-41, 2021. https://doi.org/10.1109/3DV53792.2021.00014

[19] J. Wang, L. Liu, W. Xu, K. Sarkar, and C. Theobalt, "Estimating Egocentric 3D Human Pose in Global Space," in *Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal: Canada, pp. 11480-11489, 2021. https://doi.org/10.1109/ICCV48922.2021.01130

[20] J. Wang, D. Luvizon, W. Xu, L. Liu, K. Sarkar, and C. Theobalt, "Scene-Aware Egocentric 3D Human Pose Estimation," in *Proceedings of 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver: Canada, pp. 13031-13040, 2023. https://doi.org/10.1109/CVPR52729.2023.01252

[21] J. Hwang and J. Kang, "Double Discrete Representation for 3D Human Pose Estimation from Head-mounted Camera," in *Proceedings of 2024 IEEE International Conference on Consumer Electronics (ICCE)*, Las Vegas, NV, USA, pp. 1-4, 2024. doi: 10.1109/ICCE59016.2024.10444241.

[22] H. Jiang and K. Grauman, "Seeing Invisible Poses: Estimating 3D Body Pose from Egocentric Video," in *Proceedings of 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu: HI, pp. 3501-3509, 2017. https://doi.org/10.1109/CVPR.2017.373

[23] H. Jiang and V. K. Ithapu, "Egocentric Pose Estimation from Human Vision Span," in *Proceedings of 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal: Canada, pp. 10986-10994, 2021. https://doi.org/10.1109/ICCV48922.2021.01082

[24] J. Wang, L. Liu, W. Xu, K. Sarkar, D. Luvizon, and C. Theobalt, "Estimating Egocentric 3D Human Pose in the Wild with External Weak Supervision," in *Proceedings of 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans: LA, pp. 13147-13156, 2022. https://doi.org/10.1109/CVPR52688.2022.01281

[25] Y. Zhang, S. You, and T. Gevers, "Automatic Calibration of the Fisheye Camera for Egocentric 3D Human Pose Estimation from a Single Image," in *Proceedings of 2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa: HI, pp. 1771-1780, 2021. https://doi.org/10.1109/WACV48630.2021.00181

[26] Intel. Intel® RealSense™ Depth Camera D435i [Internet]. Available: https://www.intelrealsense.com/depth-camera-d435/.

[27] P. J. Besl and N. D. McKay, "A Method for Registration of 3-D Shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 2, pp. 239-256, February 1992. https://doi.org/10.1109/34.121791

[28] Microsoft. Azure Kinect DK [Internet]. Available: https://a

zure.microsoft.com/en-us/.

[29] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas: NV, pp. 770-778, 2016. https://doi.org/10.1109/CVPR.2016.90

[30] S. Baek, Y.-H. Gil, and Y. Kim, “VR-Based Job Training System Using Tangible Interactions,” *Sensors*, Vol. 21, No. 20, 6794, October 2021. https://doi.org/10.3390/s21206794

**백성민(Seongmin Baek)**

2001년 : 포항공과대학교 대학원 (공학석사)

2001년～현 재: 한국전자통신연구원 책임연구원

※관심분야 : 자세 추정, 동작 인식, 사용자 인터랙션 등

**김예진(Yejin Kim)**

2003년 : 한국과학기술원 대학원 (전산학석사)

2013년 : University of California, Davis (전산학박사)

2003년～2016년: 한국전자통신연구원 선임연구원

2016년～현 재: 홍익대학교 게임학부 부교수

※관심분야 : 컴퓨터 그래픽스, 동작 생성, 게임 엔진 등