

상용 스마트글래스-스마트폰 연동형 단안 RGB 기반 근접 위험 인지 시스템

김 정 현¹ · 이 제 민¹ · 강 형 준² · 이 동 회² · 이 주 영² · 김 지 형³ · 류 훈³ · 최 동 수³ · 김 영 원^{3*}

¹국립금오공과대학교 소프트웨어공학과 석사과정

²국립금오공과대학교 컴퓨터공학부 소프트웨어전공 학사과정

³국립금오공과대학교 컴퓨터공학부 교수

A Monocular RGB-Based Proximity Hazard Awareness System Using Commercial Smart Glasses and a Smartphone

Jeonghyeon Kim¹ · Jemin Lee¹ · Hyeongjun Kang² · Donghee Lee² · Juyoung Lee² · Jihyung Kim³ · Hoon Ryu³ · Dong-Soo Choi³ · Youngwon Kim^{3*}

¹Master's Student, Department of Software Engineering, Kumoh National Institute of Technology, Gyeongsangbuk-do 39177, Korea

²Undergraduate Student, Software Engineering Major, School of Computer Engineering, Kumoh National Institute of Technology, Gyeongsangbuk-do 39177, Korea

³Professor, School of Computer Engineering, Kumoh National Institute of Technology, Gyeongsangbuk-do 39177, Korea

[요 약]

상용 스마트글래스는 착용자의 전방 RGB 영상을 촬영할 수 있지만, 대부분 깊이 센서가 없고 자체 연산 자원이 제한적이다. 본 논문은 Meta Ray-Ban Display의 단안 영상을 Galaxy S24+ 에서 분석하여 근접 위험을 판단하는 모바일 시스템을 제안한다. Depth Anything V2(DAv2)로 상대깊이를 추정하고, SegFormer로 경고 대상에서 제외할 영역을 구분하며, MediaPipe로 사용자 손을 제거한다. 남은 영역의 근접도, 크기와 형태를 결합해 안전(CLEAR), 주의(CHECK), 장애물(OBSTACLE) 상태를 결정한다. 150회 시행 단위 시험에서 정확도 95.33%와 F1-score 96.52%를 얻었다. 조건별 30초 영상의 3회 반복 시험에서 제안 방식의 프레임 단위 정확도는 96.41%로 YOLOv8n의 54.38%보다 높았고, 벽·모퉁이 정확 판정률은 각각 99.52%와 1.28%였다. 제안 방식의 처리속도는 2.46 fps였으며 화면 경고와 진동 경고의 평균 전체 중단 지연시간은 516.25ms와 534.45ms였다. SegFormer 제거 시 빈 복도 오탐률이 9.14%에서 99.62%로 증가하여 의미 분할의 필요성을 확인하였다.

[Abstract]

Commercial smart glasses provide first-person RGB video; however, they generally lack depth sensors and sufficient on-device computing resources. This study presents a mobile system that analyzes monocular video from Meta Ray-Ban Display on a Galaxy S24+ to identify nearby hazards. Depth Anything V2 (DAv2) estimates relative depth, whereas SegFormer excludes low-risk regions, and MediaPipe removes the user's hands. The remaining regions are scored based on proximity, size, and shape to generate CLEAR, CHECK, or OBSTACLE states. In 150 trial-level tests, the system achieved 95.33% accuracy and a 96.52% F1-score. In repeated 30-second video tests, its frame-level accuracy was 96.41%, compared with 54.38% for YOLOv8n. The corresponding wall-and-corner rates were 99.52% and 1.28%. The proposed method processed 2.46 fps, with mean display and vibration latencies of 516.25 and 534.45 ms. Removing SegFormer increased the empty-corridor false-alarm rate from 9.14% to 99.62%, confirming the importance of semantic exclusion.

색인어 : 스마트글래스, 근접 위험 인지, 단안 깊이 추정, 의미 분할, 온디바이스 인공지능

Keyword : Smart Glasses, Proximity Hazard Awareness, Monocular Depth Estimation, Semantic Segmentation, On-Device AI

<http://dx.doi.org/10.9728/dcs.2026.27.9.2453>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 04 August 2026; **Revised** 21 August 2026

Accepted 08 September 2026

***Corresponding Author; Youngwon Kim**

Tel: +82-54-478-7542

E-mail: kim01@kumoh.ac.kr

I. 서 론

상용 스마트글래스는 착용자의 전방 RGB 영상을 연속적으로 촬영할 수 있어 보행 보조, 작업 안전 및 원격 지원에 활용될 수 있다. 다만 배터리와 연산 자원이 제한되므로, 글래스는 영상을 수집하고 스마트폰은 이를 분석해 결과를 제공하는 역할 분담이 효과적이다. Ko와 Cho[1], Gao 등[2]도 스마트글래스 영상과 모바일 연산을 결합한 보행 보조 시스템을 제안하여 이러한 구성의 가능성을 제시하였다.

기존 연구는 주로 객체검출 모델로 물체의 종류와 위치를 파악한 뒤 위험 여부를 판단한다[1],[3]. 이 방식은 학습 범주에 포함된 의자, 사람, 자동차 등을 안내하는 데 효과적이지만, 학습되지 않은 물체나 화면에 일부만 보이는 물체, 얇은 차단봉은 놓칠 수 있다. 단안 깊이를 함께 사용하더라도 검출 상자 내부에서만 거리를 계산하면 객체검출의 누락이 위험 판단의 누락으로 이어진다[4].

본 연구는 물체를 먼저 식별하지 않고 사용자와 가까운 영역을 우선 탐색한다. DAv2로 상대깊이를 추정해 근접 후보를 생성하고[5], SegFormer로 바닥·보도·하늘 등 보행 경고에서 제외할 영역을 구분한다[6]. MediaPipe Hand Landmarker로 검출한 사용자 손도 제외하여[7], 객체 범주로 정의하기 어려운 벽과 모퉁이까지 근접 위험 후보에 포함한다.

Meta Ray-Ban Display가 촬영한 영상은 Meta Wearables Device Access Toolkit(DAT)을 통해 Galaxy S24+로 전송된다[8]. 스마트폰은 영상의 깊이를 추정하고, 바닥·하늘·손 영역을 제외한 뒤, 남은 영역의 크기와 가까운 정도로 위험 점수를 계산한다. 결과는 안전(CLEAR), 주의(CHECK), 장애물(OBSTACLE)의 세 단계로 표시되며, 장애물 상태에서는 진동으로도 알린다.

본 연구의 기여는 다음과 같다. 첫째, 상대깊이 지도 전체에서 근접 후보를 생성하고 의미 분할과 손 검출을 제외 마스크로 결합하는 위험 영역 통합 방법을 제시하였다. 이 방식은 미리 정의한 객체 범주에 의존하지 않아 벽과 모퉁이 등 구조적 장애물을 후보로 유지할 수 있다. 둘째, 최신 프레임 우선 처리, 상태 안정화, 진동 경고와 로그 저장을 하나의 스마트폰 앱에 통합하여 상용 스마트글래스와 연동되는 모바일 시제품을 구현하였다. 셋째, 선행연구 [1]의 객체검출 기반 접근을 참고한 YOLOv8n 비교 기준선을 구성하였다. 휴대폰에서의 반복실험과 동일 MP4의 450개 공통 프레임 분석을 통해 판정 성능, 처리속도 및 오경고 특성을 비교하였다.

II. 관련 연구

2-1 스마트글래스 기반 보행·위험 인지

Ko와 Cho[1]는 XR 글래스 영상을 스마트폰의 YOLOv8-Nano로 분석하고 음성 안내와 원격 지원을 제공하는 보행 보조 시스템을 제안하였다. Younis 등[3]은 스마트글래스 영상에서 움직이는 물체의 속도와 방향을 분석해 위험 수준을 구분하였다. 두 연구는 글래스와 스마트폰을 결합한 보행 보조의 가능성을 보여주었으나, 미리 학습한 객체가 검출되어야 위험을 판단할 수 있다.

Gao 등[2]은 RGB 카메라, ToF 거리 센서 및 전용 FPGA를 탑재한 자체 제작 글래스로 장애물 회피 장치를 구현하였다. RGB 영상과 거리 정보를 함께 사용해 빠른 반응을 달성했지만 별도 센서와 전용 연산 장치가 필요하다. 본 연구는 추가 거리 센서나 전용 연산 보드 없이 소비자용 RGB 글래스와 스마트폰만으로 구성된 시스템의 성능을 평가한다.

2-2 모바일·착용형 보행 보조

Park 등[9]은 초음파 센서, 카메라 객체 인식 및 음성 안내를 결합한 보행 보조 시스템을 제안하였다. 초음파 센서는 장애물까지의 실제 거리를 측정할 수 있지만 별도 장치가 필요하다. 본 연구는 별도 거리 센서 없이 RGB 영상의 상대깊이를 사용하며, 평가는 절대거리 추정보다 근접 위험 판정에 초점을 둔다.

보행 보조 시스템은 장애물 판정 정확도뿐 아니라 반응시간, 오경고 빈도와 알림 방식도 함께 고려해야 한다[1],[2]. 오경고가 반복되면 사용자 피로가 증가하고 시스템에 대한 신뢰가 낮아질 수 있다. 또한 이동 중에는 오래된 프레임을 순서대로 처리하기보다 현재 장면에 가까운 프레임을 분석해야 한다. 이에 따라 본 연구는 처리 중 수신된 프레임을 대기열에 누적하지 않고, 분석을 시작할 때 가장 최신 프레임을 선택한다.

2-3 단안 상대깊이 기반 위험 후보

Urban과 Caplier[4]는 YOLOv3로 장애물을 검출하고 FastDepth로 깊이를 추정하여 충돌까지 남은 시간(TTC)을 예측하였다. 물체의 움직임을 반영할 수 있다는 장점이 있지만, 객체검출 단계에서 누락된 물체에는 적용하기 어렵다. 본 연구는 DAv2의 픽셀 단위 상대깊이 지도 전체에서 근접 후보를 생성하므로 객체 범주를 식별하지 못해도 위험 후보를 유지할 수 있다.

기존 연구가 검출된 객체를 중심으로 위치와 거리를 판단했다면, 본 연구는 상대깊이 지도 전체에서 근접 후보를 먼저 생성한 뒤 의미 분할과 손 검출로 제외 영역을 제거한다. 따라서 벽과 모퉁이처럼 객체 범주로 정의하기 어려운 구조물

도 위험 후보에 포함할 수 있다. 4-4절에서는 선행연구 [1]을 참고한 YOLOv8n 비교 기준선의 구성과 원연구와의 차이를 밝히고, 동일 영상 프레임에서 두 방식의 판정 결과를 비교한다.

DAv2는 한 장의 RGB 영상에서 장면의 상대적인 깊이를 추정하는 모델이다[5]. 본 연구에서 사용한 DAv2 출력은 미터 단위의 실제 거리가 아니다. 따라서 각 프레임 안에서 무엇이 더 가깝고 먼지만 비교하며, 결과를 1m 또는 2m와 같은 거리로 해석하지 않는다.

2-4 의미 분할과 손 영역 제거

상대깊이만 사용하면 카메라 가까이에 보이는 바닥이나 보도도 장애물 후보가 될 수 있다. SegFormer는 영상의 각 픽셀을 바닥, 벽, 하늘 등으로 구분하는 모델이다[6]. 본 연구는 SegFormer가 바닥·도로·보도·하늘·천장·수면 등으로 분류한 영역을 위험 후보에서 제외한다.

사용자 손이 카메라 앞에 나타나면 매우 가까운 영역으로 계산되어 오경고가 발생할 수 있다. MediaPipe Hands는 RGB 영상에서 손과 21개 관절점을 검출한다[7]. 본 연구는 모든 관절점을 포함하는 사각형을 손 전체가 포함되도록 확장 뒤, 해당 영역을 근접 후보에서 제외한다.

SegFormer와 손 검출 결과는 장애물을 직접 결정하는 값이 아니라, 상대깊이 후보에서 경고 대상이 아닌 영역을 줄이는 보조 정보로 사용한다. 따라서 최종 판단은 한 모델의 결과에만 의존하지 않고, 상대깊이와 제외 영역을 함께 반영하여 계산한다.

III. 제안 방법

3-1 시스템 개요

제안 시스템은 그림 1과 같이 영상 촬영, 휴대폰 분석 및 사용자 알림의 세 단계로 구성된다. 글래스의 RGB 영상을 스마트폰으로 전송하면 DAv2가 상대적으로 가까운 영역을 생성한다. SegFormer는 바닥·하늘 등의 제외 영역을 분할하고 Hand Landmarker는 사용자 손을 검출한다. 근접 후보에서 두 제외 영역을 제거한 뒤 남은 영역의 근접도와 크기로 위험 점수를 계산하고, 결과를 휴대폰 화면과 진동으로 전달한다.

3-2 상용 글래스-휴대폰 연동 구조

시스템은 Meta Ray-Ban Display와 국내 출시형 Galaxy S24+로 구성하였다. 글래스는 RGB 영상을 촬영하고, 스마트폰은 영상 형식 변환, 세 모델 실행, 위험 점수 계산, 화면 표시, 진동, 로그 저장을 담당한다. LiDAR, ToF, RGB-D와

같은 별도 거리 센서는 사용하지 않았다.

Android 앱은 Meta Wearables DAT를 시작하고 카메라 권한과 글래스 연결 상태를 확인한 뒤 영상을 요청한다[8]. 영상 전송 설정은 VideoQuality.MEDIUM, 24 fps이고, 수신 영상은 504×896 해상도의 I420 형식이다. 앱은 이 영상을 화면 표시용 Bitmap과 모델 입력용 RGB 데이터로 바꾼다. 여기서 24 fps는 글래스가 보내는 영상의 설정값이며, 휴대폰이 모델 분석을 끝내는 속도와는 다르다.

3-3 프레임 처리와 실시간성 유지

모바일 CPU에서 DAv2와 SegFormer를 모든 입력 프레임에 실행하면 처리 대기열이 계속 증가한다. 이를 방지하기 위해 앱은 한 번에 한 프레임만 분석한다. 분석 중 수신된 프레임은 대기열에 누적하지 않고, 현재 분석이 끝나는 즉시 가장 최신 프레임으로 교체한다.

이 방식은 24 fps 입력 전체를 분석하지 못하지만 오래된 장면을 뒤늦게 경고하는 문제를 줄인다. 연산량이 큰 SegFormer는 분석 프레임 10장마다 한 번 실행하고, 나머지 9장에는 직전 분할 결과를 재사용한다. 따라서 고개를 빠르게 돌리면 제외 영역이 현재 장면보다 늦게 갱신될 수 있다.

최신 프레임 우선 처리는 대기열 지연을 억제하지만 처리 중 나타났다 사라지는 짧은 위험 장면을 누락할 수 있다. 또한 SegFormer 결과를 재사용하는 동안 장면이 크게 바뀌면 분할 결과와 현재 영상이 어긋날 수 있다. 구체적인 개선 방안은 5-3절에서 제시한다.

3-4 DAv2를 이용한 가까운 영역 생성

상대깊이 추정에는 DAv2 Small ONNX 모델을 사용하였다. 모델은 ONNX Runtime CPU 모드에서 내부 스트레드 6개로 실행하였다. RGB 영상을 280×280으로 축소하고 ImageNet 통계로 정규화한 뒤 [1,3,280,280] 형태로 입력하였다. 출력은 절대거리로 변환하지 않고 상대깊이로 사용하였다.

깊이 출력 D에서 지나치게 크거나 작은 값의 영향을 줄이기 위해 각 프레임의 5번째 백분위수 p_5 와 95번째 백분위수 p_{95} 를 구한다. 그다음 식 (1)과 같이 값을 0에서 1 사이로 바꾼다.

$$N(i) = \text{clamp}\left(\frac{D(i) - p_5}{p_{95} - p_5}, 0, 1\right) \quad (1)$$

$N(i)$ 는 픽셀 i 가 현재 영상에서 얼마나 가까운지를 나타내며, 값이 클수록 더 가깝다. 식 (2)와 같이 $N(i)$ 가 기준값 T_{near} 이상인 픽셀을 모아 가까운 영역 M_{near} 를 만든다.

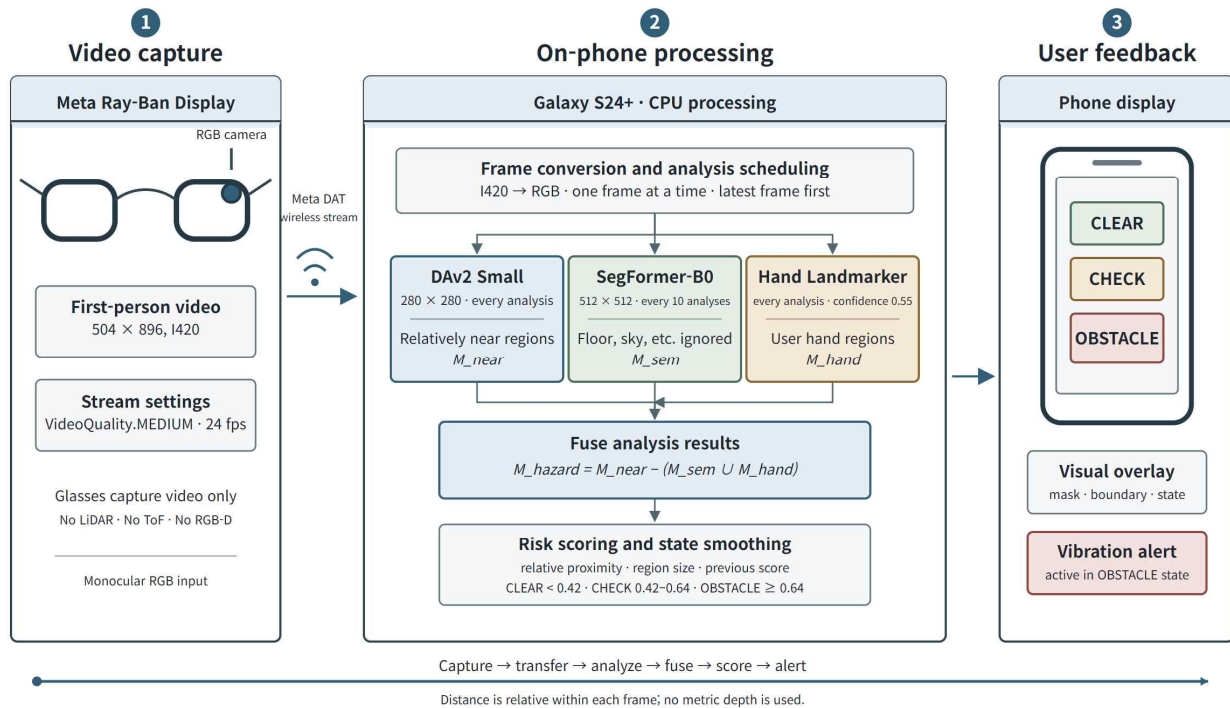


그림 1. 스마트글래스-스마트폰 기반 근접 위험 인지 시스템의 3단계 처리 흐름

Fig. 1. Three-stage workflow of the smart-glasses-smartphone proximity-hazard awareness system

$$M_{near}(i) = \begin{cases} 1, & N(i) \geq T_{near} \\ 0, & otherwise \end{cases} \quad (2)$$

근접 영역 기준은 $T_{near}=0.82$, 매우 가까운 영역 기준은 0.92로 설정하였다. 3×3 팽창·침식으로 작은 빈틈을 보정한 뒤 바닥·손·하늘을 제외하였다.

3-5 SegFormer와 Hand Landmarker를 이용한 제외 영역 생성

SegFormer-B0 ADE20K ONNX 모델은 512×512 영상을 입력받아 각 픽셀을 150개 범주 중 하나로 분류한다. 모델은 CPU 모드에서 내부 스레드 4개로 실행하였다. 본 연구에서는 floor, road, sidewalk, grass, earth, field, sand, path, sky, ceiling, water 등 보행 경고 대상에서 제외할 영역을 위험 후보에서 제거하였다.

손 영역은 MediaPipe Hand Landmarker를 CPU IMAGE 모드로 실행해 만든다. 최대 두 손을 찾고, 신뢰도가 0.55 이상인 21개 관절점을 모두 감싸는 사각형을 넓혀 손 전체를 포함한다. 넓히는 크기는 정규화 좌표 0.06과 손 크기의 35% 중 큰 값을 사용한다. SegFormer 결과는 분석 프레임 10장마다 갱신하지만 손 영역은 매 분석 프레임마다 새로 계산한다.

3-6 물체 종류에 의존하지 않는 위험 영역 통합

최종 위험 후보 영역 M_{hazard} 는 식 (3)과 같이 생성한다. DAV2가 생성한 근접 영역 M_{near} 에서 SegFormer의 제외 영역 M_{sem} 과 사용자 손 영역 M_{hand} 를 제거한다.

$$M_{hazard} = M_{near} - (M_{sem} \cup M_{hand}) \quad (3)$$

식 (3)의 결과는 장애물의 정확한 윤곽이 아니라 현재 영상에서 상대적으로 가깝고 제외 대상이 아닌 영역을 나타낸다. 각 마스크의 정의와 생성 방법은 표 1에 정리하였다.

표 1. 근접 위험 후보 마스크 구성

Table 1. Masks for proximity hazard generation

Symbol	Meaning	Generation Method
M_{near}	Relatively near regions	Apply T_{near} to the DAV2 output
M_{sem}	Regions excluding the floor and sky	SegFormer segmentation results
M_{hand}	User's hand region	Rectangular expansion enclosing the hand
M_{hazard}	Final hazard candidate region	$M_{near} - (M_{sem} \cup M_{hand})$

객체 범주를 사용하지 않으므로 객체검출기의 학습 범주에 없는 구조물도 위험 후보로 유지할 수 있다.

3-7 위험 점수와 상태 판단

위험 점수는 한 가지 특징에만 의존하지 않고, 상대깊이의 상위값과 후보 영역의 크기·형태를 함께 반영한다. 프레임 t에서 계산한 원시 위험 점수는 식 (4)와 같다.

$$S_t = 0.28P_{98} + 0.10P_{90} + 0.04P_{75} + 0.05A + 0.03A_v + 0.45C + 0.03K + 0.02E \quad (4)$$

여기서 P98, P90, P75는 깊이 지도에서 가까운 값의 분포를 나타내고, A와 A_v는 위험 후보와 매우 가까운 후보가 화면에서 차지하는 비율이다. K는 화면 전체의 대표 깊이와 비교한 가까움, E는 후보의 존재 여부를 뜻한다. C는 가장 큰 후보 영역의 면적, 폭, 높이를 식 (5)와 같이 결합한 값이다. 따라서 작은 점 형태의 잡음보다 벽이나 의자처럼 크고 이어진 영역에 높은 점수가 부여된다.

$$C = 0.50C_A + 0.20C_W + 0.30C_H \quad (5)$$

연속 프레임에서 경고 상태가 불필요하게 바뀌는 현상을 줄이기 위해 식 (6)의 지수이동평균을 적용하였다. 현재 점수가 이전 평균보다 높을 때는 $\alpha=0.70$, 낮을 때는 $\alpha=0.82$ 를 사용한다. 최종 점수는 원시 점수와 이동평균 중 큰 값으로 정하여 새로운 장애물에는 빠르게 반응하면서도 경고가 한 프레임 만에 사라지지 않도록 하였다.

$$EMA_t = (1 - \alpha)EMA_{t-1} + \alpha S_t \\ R_t = \max(S_t, EMA_t) \quad (6)$$

최종 점수에 따른 상태 구분은 식 (7)과 같다. 점수가 0.42 미만이면 안전, 0.42 이상 0.64 미만이면 주의, 0.64 이상이면 장애물 상태로 판단한다.

$$H_t = \begin{cases} CLEAR, & R_t < 0.42 \\ CHECK, & 0.42 \leq R_t < 0.64 \\ OBSTACLE, & R_t \geq 0.64 \end{cases} \quad (7)$$

앱 로그의 DANGER는 본문의 OBSTACLE과 같은 상태이다. 안전 상태에서는 경고를 제공하지 않고, 주의 상태에서는 후보 영역을 화면에 표시한다. 장애물 상태에서는 화면 표시와 진동을 함께 제공한다.

3-8 임계값 설정 절차

근접 영역 기준 0.82, 매우 가까운 영역 기준 0.92, 상태 기준 0.42와 0.64는 정량 평가와 분리한 개발 영상으로 설정하였다. 개발 단계에서는 카메라와 장애물 표면 사이의 거리를 줄자로 확인하면서, 벽·의자·책상이 1 m 이내에 들어오면 CHECK 또는 OBSTACLE로 전환되고 빈 복도에서는 불필

요한 경고가 지속되지 않도록 값을 조정하였다. 여기서 1 m는 임계값 조정에 사용한 기준 거리이며 단안 상대깊이 모델이 절대거리를 측정한다는 의미는 아니다. 최종 임계값은 150회 시행 시험과 3회 반복 비교 전에 고정하여 모든 평가에 동일하게 적용하였다. 이 절차는 개발 장면에서 경고 발생 여부를 관찰한 경험적 보정이며, 환경별 최적 임계값을 추정하는 것은 아니다.

본 연구의 적용 범위는 조명이 비교적 일정하고 배경이 단순한 실내 보행 환경이다. 식 (1)의 프레임별 정규화는 상대깊이 출력의 범위를 맞추는 과정이며, 판정 임계값을 환경에 따라 바꾸는 적응형 기법은 아니다. 현재 구현에서는 실내·실외 여부, 배경 복잡도 또는 물체 겹침에 따라 임계값을 자동 변경하지 않는다. 따라서 같은 값을 적용하더라도 장면 구성에 따라 경고 결과가 달라질 수 있으며, 실외·복잡 배경·겹침 조건으로 확장하기 위한 보정 및 검증 계획은 5-3절에 제시한다.

IV. 실험

4-1 실험 환경 및 지연시간 정의

휴대폰 내부 처리시간과 방식별 성능을 측정하기 위해 Galaxy S24+ 후면 카메라로 일반 객체, 벽·모퉁이 및 빈 경로를 각각 30초씩 촬영하였다. YOLOv8n과 제안 방식은 세 조건을 각각 3회 반복하여 방식별 9개 영상을 분석하였다. 제안 방식에서 SegFormer를 제거한 조건은 빈 경로를 3회 측정하였다. 휴대폰 내부 처리시간은 전처리 시작부터 상태 판단 완료까지로 정의하였다. 전체 종단 지연시간은 글래스 영상의 기준 LED 점등부터 휴대폰 화면 경고 또는 진동 시작까지로 정의하였다. 두 시점을 240 fps 고속카메라로 동시에 촬영해 30회 반복 측정하고 평균, 중앙값과 p90을 계산하였다.

그림 2는 방식별 평균 처리시간과 환산 처리속도를 보여준다. YOLOv8n은 4,005프레임에서 평균 66.64ms와 14.76 fps, 제안 방식은 668프레임에서 평균 407.96ms와 2.46 fps를 기록하였다. SegFormer 제거 조건은 빈 경로 269프레임에서 평균 333.88ms와 2.96 fps였다. YOLOv8n의 처리시간 중앙값과 p90은 67.22ms와 79.04ms, 제안 방식은 369.81ms와 641.01ms였으며 SegFormer 제거 조건의 p90은 370.25ms였다. 제안 방식에서 DAV2 추론은 평균 330.61ms로 전체 처리시간의 81.0%를 차지하였다. 따라서 처리속도를 개선하려면 DAV2 경량화가 가장 효과적일 것으로 판단된다.

표 2와 같이 30회 측정에서 화면 경고의 전체 종단 지연시간은 평균 516.25ms, 중앙값 514.59ms, p90 546.25ms였으며 진동 경고는 각각 534.45ms, 533.33ms, 562.92ms였

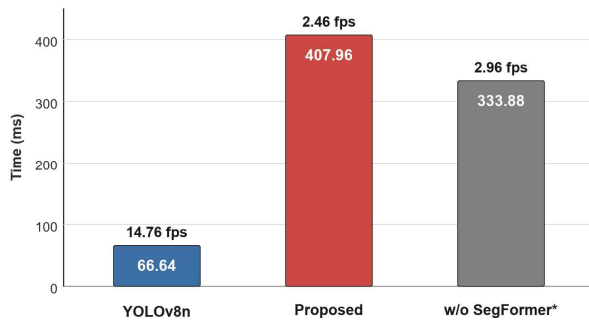


그림 2. 평균 휴대폰 내부 처리시간과 환산 처리속도
Fig. 2. Mean on-device processing time and throughput (*empty-corridor clips only)

다. 진동 경고는 화면 경고보다 평균 18.20 ms 늦게 시작하였다. 휴대폰 내부 평균 처리시간 407.96 ms는 화면 경고 지연의 약 79.0%, 진동 경고 지연의 약 76.3%를 차지하여 모델 추론이 전체 지연을 좌우하는 주요 요인임을 확인하였다.

4-2 대표 장면의 정성적 분석

그림 3은 벽, 의자·책상 및 손·바닥에서 생성된 대표 출력이다. 벽과 일반 장애물에는 위험 후보와 경계 상자가 표시되며, 손과 바닥은 비장애물로 제시하여 오경고 여부를 확인하였다. 정량 평가는 별도의 반복 시험으로 수행했으며 4-3절에 제시한다.

표 2. 전체 중단 지연시간 측정 결과

Table 2. End-to-end latency results (n=30)

Alert output	n	Mean (ms)	Med. (ms)	P90 (ms)
Display	30	516.25	514.59	546.25
Vibration	30	534.45	533.33	562.92

4-3 정량 정확도 평가 및 결과

정확도 평가는 벽, 일반 장애물 및 비장애물의 세 조건에서 각각 50회씩 총 150회 수행하였다. 일반 장애물은 의자와 책상을 각각 25회, 비장애물은 손과 바닥을 각각 25회 제시하였다. 벽과 일반 장애물 조건에서는 30cm 안전선에 도달하기 전에 장애물 경고가 나오면 성공(TP), 나오지 않으면 실패(FN)로 기록하였다. 손과 바닥 조건에서는 장애물 경고가 없으면 성공(TN), 오경고가 나오면 실패(FP)로 기록하였다. 실제 충돌은 하지 않았으며 각 시행 직후 결과를 CSV 파일에 기록하였다.

그림 4(a)와 같이 벽은 50회 중 49회를 정확히 판단해 98.0%, 일반 장애물은 50회 중 48회로 96.0%, 손과 바닥은 50회 중 46회에서 잘못된 경고가 발생하지 않아 92.0%였다. 전체 150회 중 143회를 정확히 판단해 정확도는 95.33%였다. 세부 결과는 TP=97, FN=3, FP=4, TN=46이며, 재현을



그림 3. 장애물 및 비장애물 장면의 정성적 출력 예
Fig. 3. Qualitative outputs for obstacle and non-obstacle scenes

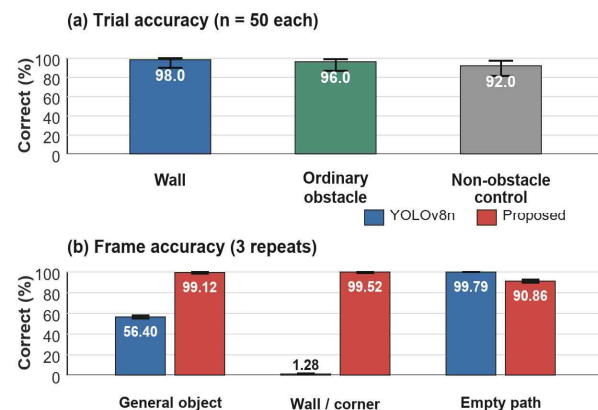


그림 4. 시행 단위 및 프레임 단위 정확도 비교
Fig. 4. Trial-level and frame-level accuracy comparison

97.00%, 정밀도 96.04%, 특이도 92.00%, 오탐률 8.00%, F1-score 96.52%였다.

4-4 객체검출 비교 기준선 및 제거실험

비교 기준선은 Ko와 Cho[1]의 객체검출 기반 접근을 참고하여 구성하였다. 원연구는 15종 장애물을 학습한 YOLOv8-Nano와 객체 종류·위치·근접도를 결합한 RSS 위

험 점수, 네 단계 경보를 사용한다. 본 비교에서는 COCO 사전학습 YOLOv8n과 검출 상자의 위치·면적을 이용한 세 단계 규칙을 사용하였다. 따라서 원연구 전체의 재현이 아니라, 객체검출 결과를 위험 후보로 사용하는 접근과 상대깊이 기반 접근을 비교하는 기준선이다. 음성 안내와 원격 지원 기능은 비교 대상에서 제외하였다.

YOLOv8n 검출 상자의 중심이 화면 가로 30~70%에 있고 하단이 화면 높이의 50% 아래에 있으면 보행 경로 후보로 정하였다. 신뢰도는 0.55, NMS IoU는 0.45이며, 상자 면적이 화면의 30% 이상이면 CHECK, 37% 이상이면 DANGER로 분류하였다. 이 기준은 평가 영상과 분리한 개발 영상에서 장애물이 1m 이내에 들어오면 경고하고 빈 경로에서는 CLEAR를 유지하도록 조정된 뒤 고정하였다. 휴대폰 반복실험은 같은 스마트폰과 카메라에서 조건별로 3회 수행하고 평균과 표준편차를 계산하였다.

표 3은 선행연구의 보고값과 본 연구의 결과를 정리한 것이다. Ko와 Cho[1]는 객체 탐지 정확도 95.4%와 음성 출력 단계를 포함한 약 160ms의 처리시간을 보고하였다. Gao 등[2]은 RGB·ToF 시스템에서 충돌 회피율 100%와 320ms 미만의 전체 종단 지연시간을 보고하였다. Urban과 Caplier[4]는 충돌까지 남은 시간 예측의 평균절대오차를 0.320s로 보고하였다.

표 3. 선행연구와 제안 시스템의 정량 비교

Table 3. Quantitative comparison with prior studies

Study	System and method	Reported quantitative result
Ko and Cho [1]	RGB XR glasses + smartphone; YOLOv8-Nano	95.4% detection; 160 ms including speech output
Gao et al. [2]	RGB + ToF; FPGA + smartphone; cross-modal model	100% collision avoidance; E2E < 320 ms
Urban and Caplier [4]	Monocular RGB; YOLOv3 + FastDepth + TTC model	TTC MAE 0.320 s (SD 0.375 s)
Proposed	Monocular RGB; Galaxy S24+ CPU; DAv2 + SegFormer + MediaPipe	96.41% frame accuracy; 407.96 ms processing; 516.25 ms display E2E

표 3은 센서, 연산 장치, 데이터 및 지표가 서로 다른 연구의 보고값을 정리한 것으로, 동일 조건의 성능 순위를 뜻하지 않는다. 특히 선행연구 [1]의 처리시간에는 음성 출력이 포함되지만 본 연구의 휴대폰 내부 처리시간은 상태 판단 완료까지이다. 객체검출 기반 접근과의 입력 조건을 맞춘 판정 비교는 별도로 표 4에 제시한다.

그림 4(b)에서 일반 객체의 정확 판정률은 YOLOv8n 56.40±1.40%, 제안 방식 99.12±0.76%였다. 벽·모퉁이에서는 각각 1.28±0.37%와 99.52±0.82%, 빈 경로에서는 99.79±0.21%와 90.86±1.37%였다. 전체 프레임을 합산한

YOLOv8n의 정확도·재현율·F1-score는 54.38%, 30.03%, 46.15%였고, 제안 방식은 96.41%, 99.32%, 97.32%였다.

일반 객체에서 YOLOv8n의 판정률이 낮게 나타난 장면에서는 근접한 물체가 화면 밖으로 잘리면서 검출 상자가 사라지거나 신뢰도가 기준 아래로 낮아지는 현상이 관찰되었다. 반면 제안 방식은 객체 전체가 보이지 않아도 근접 영역을 위험 후보로 유지하였다.

입력 영상과 분석 프레임 수의 차이를 줄이기 위해 저장된 동일 MP4를 분석하였다. 일반 객체, 벽·모퉁이 및 빈 경로에서 각각 두 개의 30초 영상을 선택하고, 영상당 0.4초 간격으로 75프레임을 추출하였다. 총 450개 공통 프레임을 동일하게 디코딩하여 YOLOv8n과 제안 방식에 입력하였다. 이 분석은 판정 결과의 비교이며, 4-1절의 휴대폰 처리시간 측정과 구분하였다. 장애물 영상은 CHECK 또는 OBSTACLE(DANGER), 빈 경로 영상은 CLEAR일 때 조건과 일치한 것으로 집계하였다.

표 4. 동일 MP4 프레임 기반 두 방식의 비교

Table 4. Same-frame comparison using identical MP4 videos

Scenario (frames)	YOLOv8n (%)	Proposed (%)
General object (150)	58.67	100.00
Wall/corner (150)	9.33	88.67
Empty path (150)	100.00	86.67
Overall (450)	56.00	91.78

표 4에서 일반 객체의 조건 일치율은 YOLOv8n 58.67%, 제안 방식 100.00%였고, 벽·모퉁이는 각각 9.33%와 88.67%였다. 빈 경로는 각각 100%와 86.67%, 전체는 56.00%와 91.78%였다. 제안 방식만 경고한 프레임은 일반 객체 62개, 벽·모퉁이 119개였으며 YOLOv8n만 경고한 경우는 없었다. 반면 빈 경로에서 제안 방식은 20개 프레임에 불필요한 경고를 표시하였다. 이 결과는 동일한 영상 입력에서도 두 방식의 구조물 경고와 빈 경로 오경고 특성이 다를 수 보여준다. 다만 영상 조건을 기준으로 집계한 값이므로 개별 프레임에서 실제 1m 이내 위험을 정확히 판정한 비율과는 구분해야 한다.

SegFormer의 효과를 확인하기 위해 빈 경로에서 SegFormer만 제거한 조건을 3회 측정하였다. SegFormer를 사용했을 때 오탐률과 평균 처리시간은 9.14±1.37%와 390.67±5.67 ms였으며, 제거했을 때는 99.62±0.66%와 333.80±7.35 ms였다. 의미 분할에는 평균 56.87 ms가 추가로 필요했지만 바닥 등 제외 영역에서 발생하는 오탐을 크게 줄였다. 손 제거와 시간 안정화의 개별 효과는 후속 실험에서 분석할 예정이다.

V. 논 의

5-1 실시간성 및 적용 범위

휴대폰 반복실험에서 제안 방식의 처리속도는 평균 2.46 fps로, 24 fps 입력의 모든 프레임을 분석하지 못하였다. 최신 프레임 우선 처리로 대기열 지연을 억제했지만 모바일 추론의 경량화가 필요하다. 같은 휴대폰 실험에서 YOLOv8n 비교 기준선은 14.76 fps로 약 6배 빨랐다. 한편 표 4의 동일 영상 분석은 처리속도가 아니라 공통 프레임에 대한 판정 특성을 비교한 것이다. 두 실험을 통해 연산 부담과 판정 특성을 각각 평가하였다.

Gao 등[2]은 RGB 카메라, ToF 거리 센서 및 전용 FPGA를 사용한 장치에서 촬영부터 알람까지 320 ms 미만의 처리 시간을 보고하였다. 본 연구는 별도 거리 센서나 전용 연산 보드 없이 소비자용 RGB 스마트글래스와 스마트폰 CPU로 시스템을 구성하였다. 화면 경고와 진동 경고의 평균 전체 중단 지연시간은 516.25ms와 534.45ms였다. 하드웨어와 입력 조건이 달라 절대적인 우열을 판단하기는 어렵지만, 휴대폰 내부 처리가 전체 지연의 76.3~79.0%를 차지하므로 DAv2를 포함한 모바일 추론 최적화가 우선적으로 필요하다.

150회 시행 단위 시험의 정확도는 95.33%였고, 3회 반복 프레임 시험의 빈 경로 정확 판정률은 90.86%였다. 별도의 동일 영상 분석에서 전체 조건 일치율은 제안 방식 91.78%, YOLOv8n 56.00%였다. 제안 방식은 벽·모퉁이 영상에서 경고를 유지한 비율이 높았으나, 빈 경로에서는 YOLOv8n보다 오경고가 많았다. 따라서 구조적 장애물을 후보에 포함하는 장점과 빈 경로 오경고를 함께 고려해야 한다.

5-2 한계

제안 시스템은 단안 RGB 영상의 상대깊이를 사용하므로 같은 물체라도 배경 구성에 따라 위험 점수가 달라질 수 있다. 어두운 환경, 반사면, 무늬가 적은 벽과 흔들린 영상에서도 깊이 추정 및 의미 분할 결과가 달라질 수 있다.

현재 구현은 SegFormer 결과를 10개 분석 프레임 동안 재사용하므로 급격한 시점 변화에서 분할 결과가 현재 영상과 어긋날 수 있다. 표 4의 장애물 영상에는 접근·후퇴가 포함되지만 프레임별 실제 거리는 기록되지 않았다. 영상 전체에 같은 정답을 적용하면 먼 장면의 지속 경고도 조건 일치로 집계될 수 있으므로, 표 4는 거리별 위험 판정 정확도가 아닌 장면 조건 일치율로 해석하였다. 또한 인접 프레임은 서로 연관되므로 450개 프레임을 독립된 보행 시행으로 보지 않았다.

이번 평가는 Galaxy S24+ 한 대, 참여자 한 명 및 제한된 실내 조명과 장애물 조건에서 수행하였다. 전체 중단 지연시간도 동일한 기기와 무선 환경에서 30회 측정했으므로 네트워크 상태, 촬영 해상도와 사용자 조건에 따라 달라질 수 있

다. 실제 보행 환경으로 확장하려면 다양한 사용자와 환경을 포함한 반복 검증과 모바일 추론 최적화가 필요하다.

5-3 향후 개선 방향

향후에는 FP16-INT8 경량화, NNAPI/GPU 가속, 소형 분할 모델과 장면 변화 기반 SegFormer 갱신을 검토하여 처리 시간을 줄일 예정이다. 프레임 수신·분석·표시 시각, 프레임 나이와 폐기 수를 기록해 최신 프레임 처리의 안정성을 평가한다. 판정 비교에는 프레임별 실제 거리와 경고 필요 여부를 독립적으로 표시하고, 영상 단위로 보정 자료와 평가 자료를 분리할 계획이다. DAv2, 의미 분할, 손 제거와 시간 안정화의 효과도 단계별 제거실험으로 확인한다.

환경 변화에 대해서는 실내·실외, 배경 복잡도와 물체 겹침 여부를 구분해 고정 임계값의 오경고와 미경고를 평가할 예정이다. 별도 보정 영상에서 근접 기준과 상태 기준의 후보값을 탐색하고, 1 m 이내 경고 누락을 줄이면서 빈 경로 오경고를 억제하는 기준을 선택한 뒤 평가 영상에는 고정 적용한다. 이후 밝기와 깊이 분포의 변화에 따라 기준을 조정하는 적응형 방식과 비교하되, 갱신 범위와 빈도를 제한해 상태가 불안정하게 변하지 않도록 할 계획이다. 이 과정의 후보값, 선택 기준 및 변경 이력을 기록하여 보정 절차를 재현할 수 있도록 한다.

VI. 결 론

본 논문은 Meta Ray-Ban Display의 단안 RGB 영상을 Galaxy S24+에서 분석하여 근접 위험을 알리는 모바일 시스템을 구현하였다. DAv2로 근접 영역을 생성하고 SegFormer와 MediaPipe로 바닥·하늘·사용자 손을 제외하여, 객체 범주로 정의하기 어려운 벽과 모퉁이도 위험 후보에 포함하였다. 또한 최신 프레임 우선 처리, 상태 안정화, 진동 경고와 로그 저장을 하나의 앱에 통합하였다.

150회 시행 단위 시험에서 정확도 95.33%와 F1-score 96.52%를 얻었다. 3회 반복 프레임 비교에서 제안 방식의 정확도와 F1-score는 96.41%와 97.32%로 YOLOv8n의 54.38%와 46.15%보다 높았다. 동일 MP4의 450개 공통 프레임을 분석한 전체 조건 일치율은 제안 방식 91.78%, YOLOv8n 56.00%였다. 제안 방식은 벽·모퉁이 영상에서 경고 비율이 높았지만 빈 경로에서는 오경고가 더 많았다. 휴대폰 실험에서 제안 방식의 처리속도는 2.46 fps였고, 화면·진동 경고의 평균 전체 중단 지연시간은 516.25ms와 534.45ms였다. SegFormer 제거 시 빈 경로 오답률은 9.14%에서 99.62%로 증가하였다. 향후에는 환경별 임계값 보정, 거리 라벨을 갖춘 동일 영상 비교와 모바일 추론 최적화를 통해 적용 범위를 넓힐 예정이다.

감사의 글

본 연구는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(RS-2026-25478706, RS-2026-25516934). 또한 본 연구는 국립금오공과대학교 슈퍼컴퓨터 지원 프로그램의 지원을 받아 수행되었음(2026).

참고문헌

- [1] M. Ko and J. Cho, "XR Glasses-Based Integrated Walking Assistance System with Obstacle Detection and Real-Time Remote Support for the Visually Impaired," *Journal of Korean Institute of Information Technology*, Vol. 24, No. 5, pp. 141-150, May 2026. <https://doi.org/10.14801/jkiit.2026.24.5.141>
- [2] Y. Gao, D. Wu, J. Song, X. Zhang, B. Hou, H. Liu, ... and L. Zhou, "A Wearable Obstacle Avoidance Device for Visually Impaired Individuals with Cross-Modal Learning," *Nature Communications*, Vol. 16, 2857, March 2025. <https://doi.org/10.1038/s41467-025-58085-x>
- [3] O. Younis, W. Al-Nuaimy, M. H. Alomari, and F. Rowe, "A Hazard Detection and Tracking System for People with Peripheral Vision Loss Using Smart Glasses and Augmented Reality," *International Journal of Advanced Computer Science and Applications*, Vol. 10, No. 2, pp. 1-9, 2019. <https://doi.org/10.14569/IJACSA.2019.0100201>
- [4] D. Urban and A. Caplier, "Time- and Resource-Efficient Time-to-Collision Forecasting for Indoor Pedestrian Obstacles Avoidance," *Journal of Imaging*, Vol. 7, No. 4, 61, 2021. <https://doi.org/10.3390/jimaging7040061>
- [5] L. Yang, B. Kang, Z. Huang, Z. Zhao, X. Xu, J. Feng, and H. Zhao, "Depth Anything V2," *Advances in Neural Information Processing Systems*, Vol. 37, pp. 21875-21911, 2024. <https://doi.org/10.52202/079017-0688>
- [6] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, "SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers," *Advances in Neural Information Processing Systems*, Vol. 34, pp. 12077-12090, 2021. <https://doi.org/10.48550/arXiv.2105.15203>
- [7] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, "MediaPipe Hands: On-Device Real-Time Hand Tracking," arXiv:2006.10214, 2020. <https://doi.org/10.48550/arXiv.2006.10214>
- [8] Meta. Meta Wearables Device Access Toolkit for Android [Internet]. Available: <https://github.com/facebook/meta-wearables-dat-android>.

- [9] S. Park, J. Park, H. Ryu, H. Lee, and K.-M. Lee, "Situation Awareness Auxiliary IoT Device and Pedestrian Navigation Service Application for Visually Impaired Persons," *Journal of Digital Contents Society*, Vol. 21, No. 2, pp. 269-275, February 2020. <https://doi.org/10.9728/dcs.2020.21.2.269>



김정현(Jeonghyeon Kim)

2025년 : 국립금오공과대학교
컴퓨터공학부 (학사)

2026년 : 국립금오공과대학교
소프트웨어공학과
(석사 과정)

2020년~2025년: 국립금오공과대학교 컴퓨터공학부 학사
2025년~현 재: 국립금오공과대학교 소프트웨어공학과
석사과정

※ 관심분야 : AX(AI Transformation), 확장현실(Extended Reality), 인간컴퓨터상호작용(Human Computer Interaction) 등



이제민(Jemin Lee)

2025년 : 국립금오공과대학교
컴퓨터공학부 (학사)

2026년 : 국립금오공과대학교
소프트웨어공학과
(석사 과정)

2020년~2025년: 국립금오공과대학교 컴퓨터공학부 학사
2025년~현 재: 국립금오공과대학교 소프트웨어공학과
석사과정

※ 관심분야 : AX(AI Transformation), 확장현실(Extended Reality), 인간컴퓨터상호작용(Human Computer Interaction) 등



강형준(Hyeongjun Kang)

2023년 : 국립금오공과대학교
컴퓨터공학부 재학중

2023년~현 재: 국립금오공과대학교 컴퓨터공학부 학사과정

※ 관심분야 : AX(AI Transformation), 확장현실(Extended Reality), 인간컴퓨터상호작용(Human Computer Interaction) 등

이동희(Donghee Lee)



2021년 : 국립금오공과대학교
컴퓨터공학부 재학중

2021년~현 재: 국립금오공과대학교 컴퓨터공학부 학사과정
※관심분야 : AX(AI Transformation), 확장현실(Extended Reality), 인간컴퓨터상호작용(Human Computer Interaction) 등

이주영(Juyoung Lee)



2023년 : 국립금오공과대학교
컴퓨터공학부 재학중

2023년~현 재: 국립금오공과대학교 컴퓨터공학부 학사과정
※관심분야 : AX(AI Transformation), 확장현실(Extended Reality), 인간컴퓨터상호작용(Human Computer Interaction) 등

김지형(Jihyung Kim)



2015년 : 국립한밭대학교(공학사)
2017년 : 국립한밭대학교(공학석사)
2022년 : 포항공과대학교(공학박사)

2022년~2024년: 포항공과대학교 미래IT융합연구원 박사 후 연구원

2024년~현 재: 국립금오공과대학교 컴퓨터공학부 조교수
※관심분야 : 몰입형 기술(Immersive Technology), 디지털 트윈(Digital Twin), 기술 기반 학습(Technology-enhanced Learning), 인간 컴퓨터 상호작용(Human-Computer Interaction)

류훈(Hoon Ryu)



2011년 : Purdue 대학교
전기컴퓨터공학부 (공학박사)
2005년 : Stanford 대학교 전기공학부
(공학석사)
2003년 : 서울대학교 전기공학부
(공학사)

2005년~2011년: 삼성전자 반도체총괄 System LSI 사업부
2011년~2024년: 한국과학기술정보연구원 국가슈퍼컴퓨팅본부
선임/책임연구원/양자정보응용연구단장

2024년~현 재: 국립금오공과대학교 컴퓨터공학부 교수
※관심분야 : 반도체 소재 전자구조 계산 (Electronic Structure Calculation for Semiconductor Materials), 거대 수치해석 병렬처리 (Parallel Processing of Numerical Analysis), 양자 알고리즘 활용기술 (Applications of Quantum Algorithm) 등

최동수(Dong-Soo Choi)



2013년 : 한국기술교육대학교(공학사)
2015년 : 한국기술교육대학교(공학석사)
2020년 : 한국기술교육대학교(공학박사)

2022년~2025년: 국립금오공과대학교 컴퓨터공학부 조교수

2025년~현 재: 국립금오공과대학교 컴퓨터공학부 부교수

※관심분야 : 몰입형 VR/AR, 햅틱(Haptics), 투명 압력 센서, 3D 터치 센서, 진동 촉각 액추에이터, 초점 가변 렌즈

김영원(Youngwon Kim)



2012년 : 고려대학교 (공학사)
2018년 : 고려대학교 대학원
(공학박사)

2018년~2019년: 고려대학교

2019년~2023년: 한국전자기술연구원(KETI)

2024년~현 재: 국립금오공과대학교 컴퓨터공학부 교수

※관심분야 : AX(AI Transformation), 확장현실(Extended Reality), 인간컴퓨터상호작용(Human Computer Interaction) 등