

mmWave 기반 낙상 감지

팀 도플러

수행 배경 및 목표

사회 취약층의 부상 및 사망의 주요 원인
중 하나인 낙상을 감지

보호 대상이나 환자의 낙상을 감지 및
예측하여 응급상황에 대한 신속 조치

다른 환자 모니터링 시스템과 결합한 케어
솔루션



mmWave Radar 센서의 신호를 사용
낙상을 감지하는 Ai 기반 시스템으로
신속 대응 시스템의 구축



기존 진행물의 개선이 주 목표

문제 정의 및 요구사항

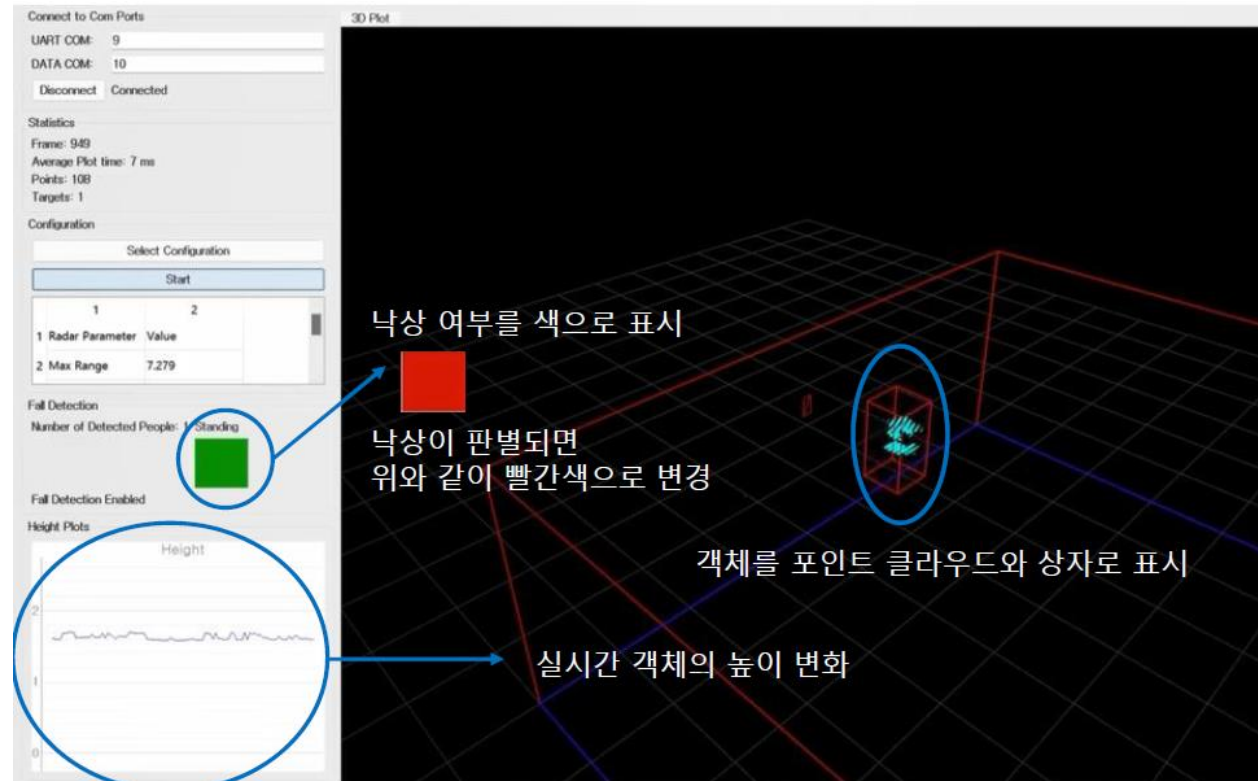
기존 시스템의 한계

- 최대 1인 까지 인식 및 추적 가능
- PyQt 기반 UI
- 높이 값만으로는 낙상감지
- 실시간 데이터 처리 불가능
- 구현 가능성 조사에 초점

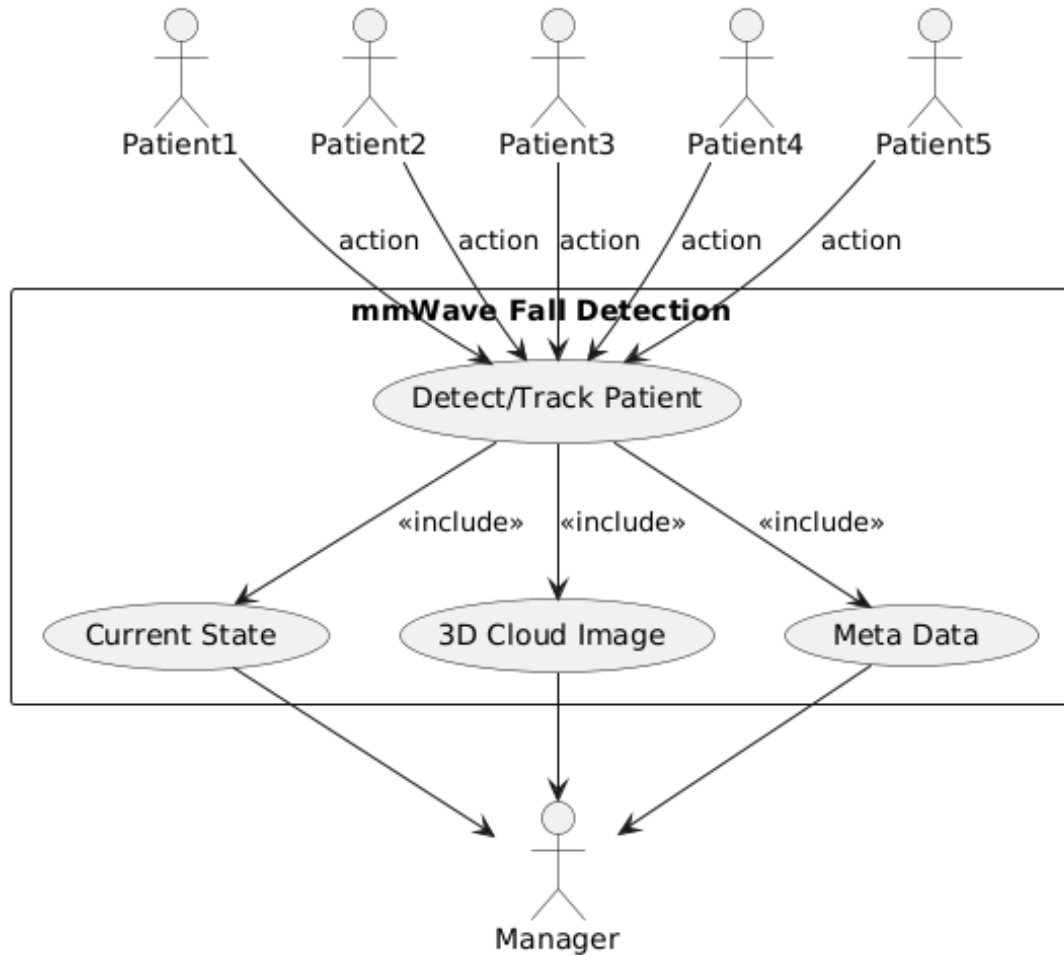
기업체 요구사항(시스템 고도화)

- 최대 5인 인식 및 추적(multi-label)
- 실시간 데이터 처리
- 낙상 외의 다양한 카테고리로 인식
- 개선된 UI/UX
- 다양한 파라미터를 처리하는 개선된 Ai 모델

Frame	DetectObject	x	y	z	V	AbsoluteHeight	RelativeHeight	Length	Width	Age	AverageHeight
3	0	-1.40866	3.207616	0.875	-0.06882	1.708473347	0.315683685	0.624659761	0.631763392	0	1.708473347
5	0	-1.40681	3.222103	0.889787	0.018543	1.813822782	0.776145074	0.489468175	0.484487049	0	0.090691139
7	0	-1.3796	3.220425	1.028345	0.127167	1.811009503	0.616403216	0.636572786	0.520297272	0	0.090550475
8	0	-1.36915	3.225611	1.06825	0.139881	1.804270476	0.40448963	0.377317427	0.350130423	0	0.176236475
9	0	-1.36422	3.233621	1.099951	0.127458	1.805502199	0.38815986	0.385983774	0.258078434	0	0.257699761
10	0	-1.35797	3.264298	1.068931	0.117643	1.806689081	0.473951349	0.37236661	0.39561755	0	0.335149227
11	0	-1.35147	3.315344	0.990115	0.105735	1.805645442	0.773617789	0.459303944	0.497957821	0	0.408674038
12	0	-1.34508	3.341379	0.985271	0.100638	1.799131668	0.579906908	0.419883908	0.331714664	0	0.47819692
13	0	-1.35606	3.350156	0.986061	0.042196	1.730271459	0.719694708	0.434762791	0.425663907	0	0.540800647
14	0	-1.3541	3.380324	0.939381	0.031723	1.790241182	1.039491777	0.494431214	0.380943004	0	0.603272673
15	0	-1.34777	3.401837	0.914487	0.037948	1.78659038	0.605135209	0.523200315	0.33204223	0	0.662438559



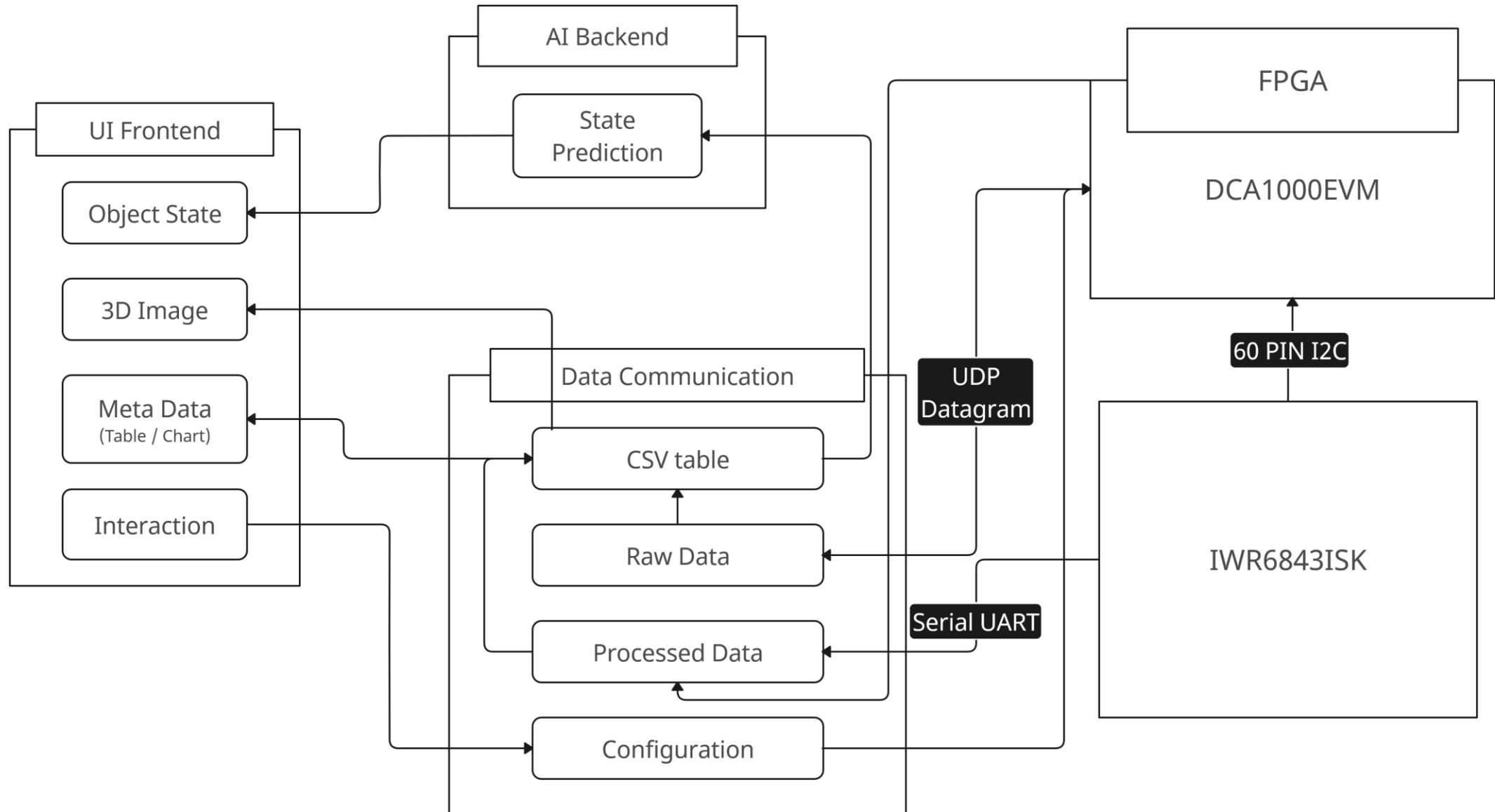
시스템 요구분석(Usecase Diagram)



기업 요구사항

- 낙상만을 감지하는 기존 시스템과 다르게 앉기, 작업, 이동 등 여러 State를 인지
- 3D 이미지는 기존 클라우드를 개선하여 스켈레톤 등으로 인지하기 쉬운 형태로 제공
- 개선된 UI/UX 기반 접근성 향상

전체 시스템 구조



Data Communication – 다양한 구현 방법



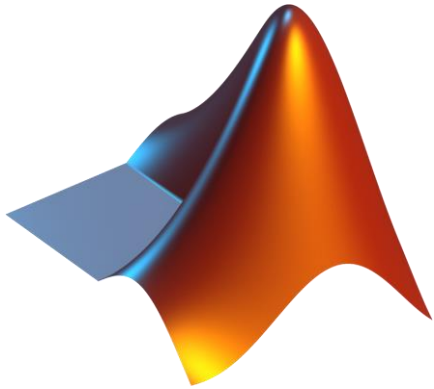
Figure 3-22. DCA1000EVM mode

- mmWave Studio – Communicate via EVM
- mmWave SDK
- UniFlash – Firmware upload
- FPGA HDL(Optional) – Lattice Diamond Programmer
- C++ - Data Parse & Convert
- CCS studio – Firmware Programming

UI Frontend – 라이선스 및 복잡한 코드



- 기존 시스템은 People Count 예제 기반으로 추정 (3D 구현에 한계점 및 복잡한 Api 구성)
- PyQt5에서의 라이선스 문제
- 내부 데이터 처리에 따라서 다른 플랫폼 사용또한 고려중 – 기존 작업물의 확인이 필요



AI Frontend - 예상 성능지표 및 평가모델



- 기존 GRU 모델 계승하여 발전시킬 예정
(전처리 -> 라벨링 -> 데이터 정규화 -> GRU)
- Multi-Label / Multi-Class 구현과 Input 다변화에서
가장 큰 차이
- 회사가 요구하는 성능지표 X
- 검증 : F1-Score (평가 모델)
- 손실함수 : Cross Entropy Loss (성능 지표)

일정 진행 상황

9/13

센서 수령 및 문제 확인

9/22

장비 기동 및 테스트

추석
각 파트별
자료 조사

9/23

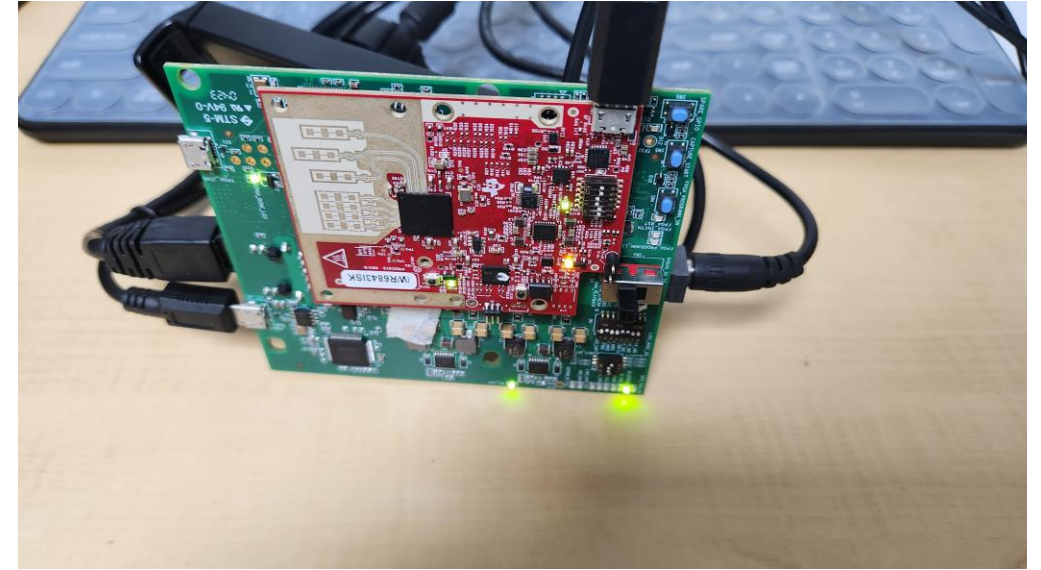
멘토님과 대면
회의 진행

mmWave Demo Visualizer

mmWave Studio

Out of Box Demo Binary ->
UniFlash 통한 업로드 필요

대면 회의 진행 및 피드백 (9/23)



-피드백

Smart Radar System 자료 참조 / UI 구현 가능 범위 조사 / 1학기 진행 일정

향후 진행방향

- 코드 수령 후 진행방향 확정 가능
- Github repository 관리 및 Coding Convention, Branch 관리
- 회의 진행방향 및 플랫폼 : Zoom 회의, Discord 통한 아이디어, 파일 교환
- 여비 사용계획 : 공용 계정에 ChatGpt, Colab pro 구독(각 3개월)