

## 1. 개발 과제의 요약

어르신들의 건강 및 안전 케어에 대한 관심이 높아지고 있으며, 이를 위한 다양한 서비스가 검토되고 있다. 특히 어르신들의 거주 공간에서 생활 정보를 바탕으로 어르신들의 응급 안전 대응 및 건강 관리를 할 수 있다면 이는 매우 효과적인 서비스가 될 것이다. 본 개발의 목표는 디스플레이형 AI 단말기를 활용하여 65세 이상의 시니어를 대상으로 일상생활 데이터를 모으고, 앞에서 말한 서비스 중 실시간 행동 분석을 통한 어르신들의 안전 케어 및 위험 상황 판단에 대한 스마트 서비스를 제공하는 시스템을 개발하는 것이다.

## 2. 설계사양

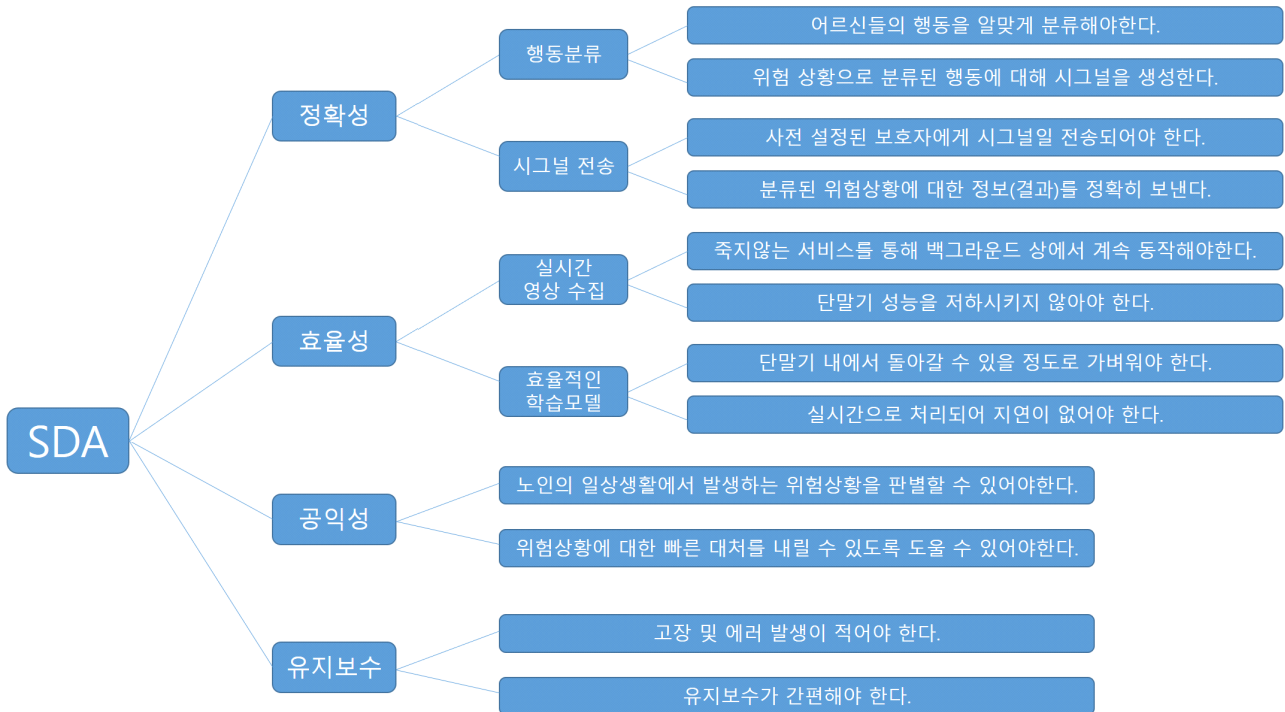
### 가. 제품 요구사항

번호	요 구 사 항	D or W	비고
1	죽지 않는 서비스를 구현한다.	D	中
2	단말기 카메라를 이용하여 매초마다 일정 개수의 이미지를 수집한다.	D	中
3	보호자 정보를 설정한다.	D	小
4	위험 상황 판단 시, 보호자에게 알림을 보낸다.	D	大
5	학습모델을 단말기 내에서 실행될 수 있도록 이식한다.	D	大
6	영상처리를 통해 인간 객체를 스켈레톤화한다.	D	大
7	스켈레톤화된 데이터를 딥러닝을 가지고 행동분류한다.	D	大
8	학습 모델의 처리 속도가 10초를 넘기지 않는다.	D	大
9	카메라가 비추지 못하는 구역을 소음분류를 통해 판단한다.	W	小
10	이미지를 주기적으로 저장/삭제한다.	W	小
11	단말기에서 보호자 측으로 보내는 알림이 5초 안에 송신된다.	W	中
12	분류해야하는 행동의 종류를 5개 이상으로 한다.	W	小

## 나. 평가 내용

- ◇ 테스트 행동에 대한 분류
- ◇ 행동 분류 모델 처리
- ◇ 알림 메시지 전송

## 다. 목적 계통도

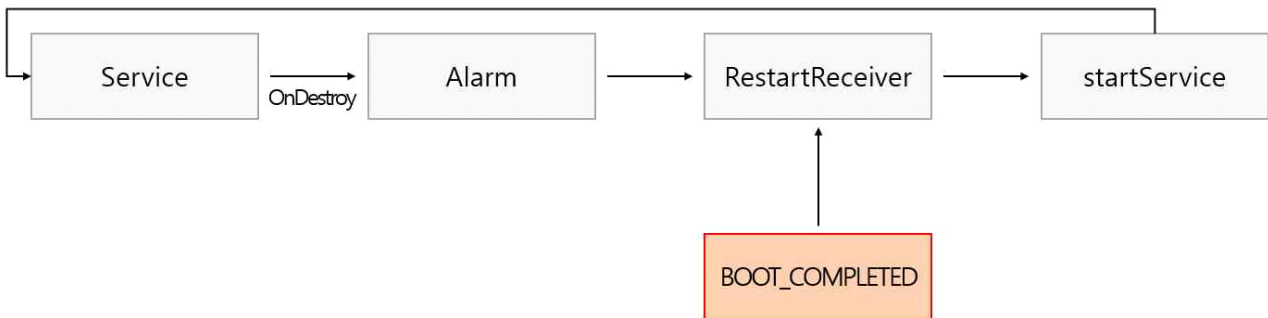


### 3. 개념설계안

#### 가. 죽지 않는 서비스

서비스는 앱이 UI 없이 백그라운드에서 특정 시간 동안 실행되는 것을 의미한다. 많은 앱은 서비스를 데이터 처리, 연산 등에서 많이 사용한다. 일반적인 안드로이드 앱들은 죽지 않는 서비스(Immortal Service)를 주로 이용한다. 왜냐하면, 백그라운드 서비스는 OS가 Task Kill을 해서 서비스를 종료시켜버릴 수 있지만 죽지 않는 서비스는 여기에서 서비스가 종료되었을 때, 다시 서비스를 재실행시키는 방식으로 구현하여 서비스가 완전히 종료되지 않도록 하기 때문이다.

죽지 않는 서비스의 구현은 일반적인 서비스를 구현하는 것과 비슷하지만 차이점이 존재한다. 먼저 서비스를 실행시키고 서비스가 어떠한 조건으로 죽으면 알람을 통해 다시 startService 메서드를 사용, 서비스를 재실행시킨다. 제일 중요하게 처리해야 하는 조건은 위의 문단에서 말했던 OS가 서비스를 죽이는 경우이다. 앱을 종료하거나 내리거나 화면을 끄는 경우는 서비스의 특성상 괜찮지만 OS가 서비스를 죽이는 경우에는 서비스가 시작돼야 하는 단이 백그라운드이기 때문에 조금 더 복잡하다. 지금 안드로이드 버전이 Snow Cone(12.1)까지 나왔는데 안드로이드 Oreo(8.0) 이후 버전에서는 서비스를 백그라운드에서 실행하는 것을 금지하기 때문에 포그라운드에서 실행해야 한다. 그래서 죽지 않는 서비스를 구현할 때 startForegroundService 메서드라는 한 가지 과정이 더 필요하지만, 우리가 사용하는 단말기의 버전은 Oreo 이전 버전(7.1)이기 때문에 startService 메서드와 리시버만 가지고 죽지 않는 서비스를 구현할 수 있다. 이외의 죽지 않는 서비스를 위하여 구현이 필요한 부분은 재부팅 시 서비스 자동 실행이다. 이 부분은 브로드캐스트리시버를 사용하여 액션을 받아 현재 부팅이 되었다는 것을 인식하면 startService를 실행시키는 것으로 해결할 수 있다.

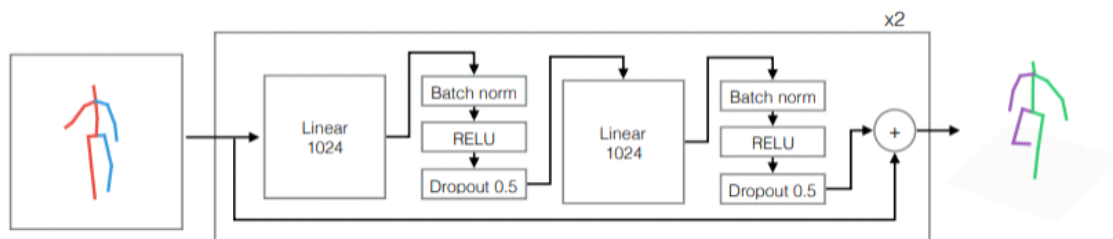


## 나. 영상에서의 행동 인식

영상을 입력받으면, 영상을 실시간 행동 분석이 가능한 프레임 수로 나누어 인간 객체를 Skeleton으로 벡터화시키는 과정을 진행한다. 이때, 구글에서 만들어서 배포한 딥러닝 API인 Pose Detection API를 사용한다.

Pose Detection API를 거친 영상은 프레임별로, 인간 객체의 골격 위치 정보를 저장할 수 있다.

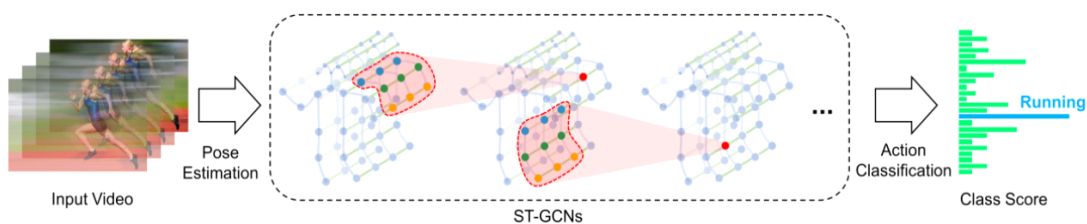
이러한 정보를 곧바로 모델의 입력에 사용할 수 있지만, 2차원의 관절 정보로는 분류 정확도가 낮음을 확인했다. 실제로, 실시간 행동 인식 모형의 State of Art는 Kinetic 카메라로부터 얻어진 RGB+D 영상을 기준으로 처리하는 모형이 많다. 하지만, 실제 AI 단말기 내에서 사용되는 카메라는 RGB를 출력하는 일반 카메라로, 분류 성능을 위한 정보의 양이 한정되어있다.



이러한 문제를 해결하기 위해, deep-learning 기반 모델로 주어진 2D 관절 좌표를 3차원으로 매핑해주는 오픈소스인 3D-Pose-Baseline을 이용한다. 이는 RGB 카메라에서 얻은 정보를 3차원 공간정보로 바꾸어 주는 과정을 추가하여, 카메라의 성능 개선 부담을 줄이면서, 정보의 양은 늘리는 효과를 가진다.

즉, 앞선 두 과정을 파이프라인으로 설정하여, 입력 데이터가 들어오면, 프레임별로 2D-Skeleton 정보를 추출하고 이를 3D-Skeleton 정보로 변환하는 과정을 거치게 된다.

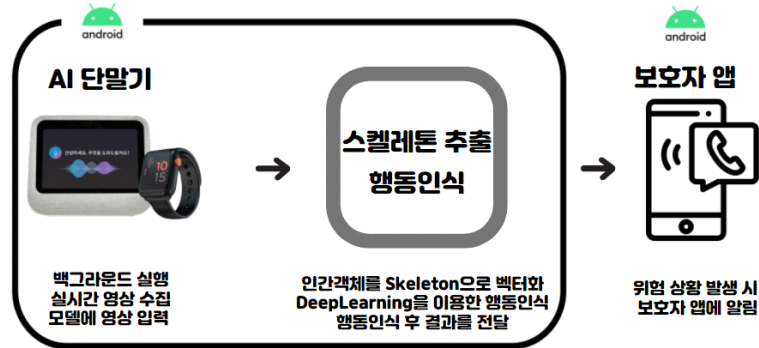
입력 파이프라인을 거친 3D 공간정보는 행동 분류 모델의 입력이 된다. 행동 분류 모델은 ST-GCN(Spatial Temporal Graph Convolutional Networks)을 사용한다.



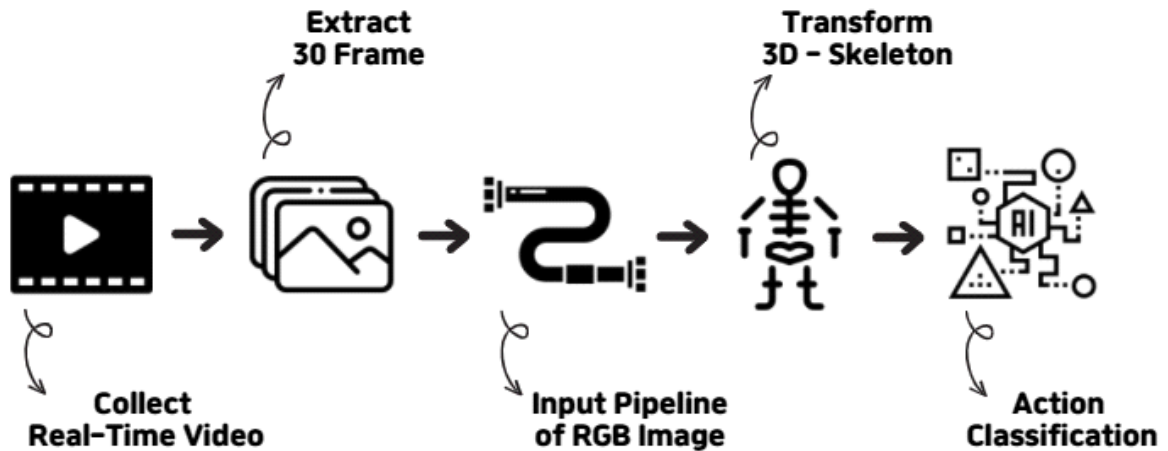
어르신들의 70세 이상의 고령자 53명의 자택을 방문하여 기상부터 취침까지의 하루 행동을 직접 관찰한 데이터에서 245개의 일상 활동 유형 중 빈번하게 나타나는 행동 55가지 중에 쓰러지기 행동은 위험상황으로 정의하고 나머지 TV 시청, 식사 관련 활동, 화장실 사용, 식사 준비, 전화 통화, 약 복용, 요리, 청소 등은 쓰러지기가 아닌 일상생활로 정의한다. 해당 라벨들은 분류 모형의 예측 클래스가 되며, 각 행동들에 대한 로짓값을 출력하도록 한다. 그렇게 쓰러지기와 일상생활을 이진분류하는 모델을 만든다.

## 다. AI 단말기 내에서 실행되는 실시간 행동분석

앞의 두 가지 개념을 이용해서 결과적으로 다음과 같은 시스템을 만들고자 한다.



실시간 행동 분석을 위해, AI 단말기의 전원을 켜면 단말기 내의 카메라를 제어하여 영상(또는 프레임)을 자동으로 수집하여, 윈도우 환경에서 학습한 입력 파이프라인과 행동 분류 모델의 입력으로 넣는다.

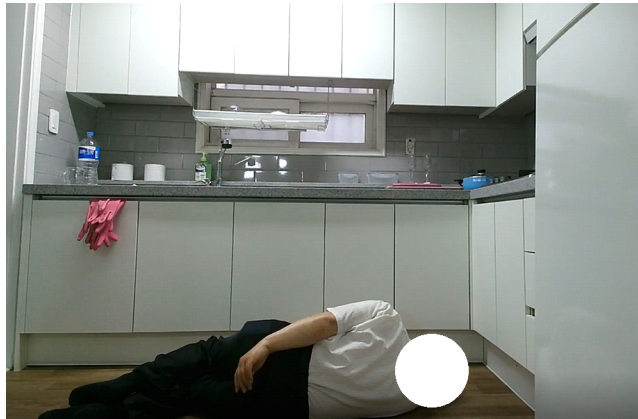


실시간 영상(일정 프레임)에서의 행동을 분류하기 위해, 입력 이미지의 개수(프레임의 개수)를 30개로 제한하여 매 30 프레임마다 어떤 행동인지 판단하게 된다. 윈도우 환경에서 작성한 모델은 텐서플로 라이트로 변환 후 이식을 진행한다. 모델 과정의 출력 중 이상 행동이 검출되면, 사전에 연결된 보호자에게 해당 상황을 설명하는 알림을 전송한다.

## 라. 위험 상황에 대한 정의 및 범위

저희 조가 정의한 위험 상황은 '시니어 분들이 집 안에서 혼자 계실 때, 어떠한 이유로 보호자 측에 연락을 취할 수 없는 상황이 되어 빠른 대처가 되지 않았을 때, 시니어 분들의 건강 악화가 우려되는 경우나 목숨의 위험이 있는 경우 또는 시니어분들이 돌아가셨지만 보호자 측에서 인지하지 못하고 있는 경우'이다.

그리고 위험상황을 분류할 행동들은 낙상이다. 낙상을 감지하는 기준은 모델을 학습시킬 때 들어간 ETRI-Activity3D 데이터셋의 영상들이 기준이 될 것이다.



## 4. 평가 및 분석

### 가. 평가 기준

평 가 항 목	평 가 방 법	적 용 기 준	개 발 목표치	비중 (%)
분류 정확도	테스트 셋 평가와 팀원들이 직접 상황을 재현해보며 평가	Accuracy, 100회 이상 평가	평균 90% 이상	40
모델 속도	입력값을 넣었을 때, 최종 결과까지의 출력 시간을 계산	System time, 200회 이상 측정	평균 10초 이하	40
응답 시간	위험 상황 데이터를 입력 후, 보호자 앱에 전송되는데 걸리는 시간 계산	System time, 50회 이상 측정	평균 5분 이하	20
			합계	100