

# **Fall Detection**

Justin Haryanto

Nhan Nguyen

## **INTERFACE CONTROL DOCUMENT**

REVISION – Draft  
03 October 2022

# INTERFACE CONTROL DOCUMENT FOR Fall Detection

PREPARED BY:

---

Author Date

APPROVED BY:

---

Project Leader Date

---

John Lusher II, P.E. Date

---

T/A Date

## Change Record

Rev.	Date	Originator	Approvals	Description
0	03 Oct 2022	Justin Haryanto		Draft Release
1	04 Dec 2022	Nhan Nguyen		Revision

## Table of Contents

<b>Table of Contents .....</b>	<b>III</b>
<b>List of Tables .....</b>	<b>IV</b>
<b>No table of figures entries found.....</b>	<b>IV</b>
<b>List of Figures.....</b>	<b>V</b>
<b>No table of figures entries found.....</b>	<b>V</b>
<b>1. Overview.....</b>	<b>1</b>
<b>2. References and Definitions.....</b>	<b>1</b>
2.1. References.....	1
2.2. Definitions .....	1
<b>3. Video Processing.....</b>	<b>Error! Bookmark not defined.</b>
<b>4. Pose Estimation.....</b>	<b>2</b>
4.1. AlphaPose.....	2
<b>5. Fall Detection .....</b>	<b>2</b>
5.1. Convolution Neural Network (CNN) .....	2
5.2. Classification Model .....	<b>Error! Bookmark not defined.</b>

## **List of Tables**

No table of figures entries found.

## List of Figures

No table of figures entries found.

## 1. Overview

This document describes the interfaces between the subsystems of the fall detection system. There are two interface sections for the pose estimation and fall detection subsystems. Videos are inputted into the pose estimation system. The results of this are inputted into the fall detection system. The ICD will explain in detail how the subsystems interface together to achieve the goal of the whole system.

## 2. References and Definitions

### 2.1. References

Pytorch - 1.12.1

TensorFlow - 2.10.0

AlphaPose - Sep 2022

### 2.2. Definitions

CNN	Convolution Neural Network
FPS	Frames per Second

### **3. Pose Estimation**

Video processing is performed by AlphaPose for videos up to 30 FPS. This subsystem applies bounding boxes, predict key points, and estimates the poses of people in the video. AlphaPose takes the processed video and detects people in the video, applying key points and bounding boxes on them.

#### **3.1. Video Processing**

Training videos have been divided into frames with labels indicating the action of the person in the video. However, it was found that videos that are too long caused AlphaPose to stall, therefore, videos will need to be kept short. This subsystem could cut the video to the desired length and temporarily hold onto a bank of videos to process.

#### **3.2. AlphaPose**

AlphaPose has video processing built into its program and only needs to output to the fall detection system. This system is a multi-person keypoint detection system that can take video or camera inputs. It places bounding boxes on individuals and estimates the pose by feeding the bounding boxes to a pipeline.

AlphaPose was chosen because it can overcome several problems common in other systems, such as inaccurate bounding boxes. It was found that AlphaPose will stall when given a video file with a duration that is too long. Unless improvements are made, video durations will need to be short.

### **4. Fall Detection**

This subsystem uses information on key points and bounding boxes generated from the pose estimation subsystem to determine if a fall has occurred.

#### **4.1. Convolution Neural Network (CNN)**

CNN is a deep learning algorithm that can recognize patterns in images. This makes it suitable for analyzing images and bounding boxes to determine if a fall has occurred. TensorFlow is used because it already has the framework set up for the CNN model.

#### **4.2. Random Forest**

Random forest is a simple classification model that can analyze key points and determine if a fall has occurred. Having an algorithm that is simple will reduce resource costs.