FALL DETECTION SYSTEM

Justin Haryanto

Nhan Nguyen

INTERFACE CONTROL DOCUMENT

REVISION – 3.0

30 April 2023

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
1.0	03 Oct 2022	Justin Haryanto		Draft Release
2.0	04 Dec 2022	Nhan Nguyen		Revision
3.0	30 Apr 2023	Justin Haryanto		Revision

Table of Contents

List of Tables

List of Figures

1. Overview

This document describes the interfaces between the subsystems of the fall detection system. There are two interface sections for the GUI and fall detection subsystems. Videos are inputted into the GUI 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
GUI	Graphic User Interface

3. GUI

This subsystem acts as a convenient interface between the user and the fall detection subsystem by allowing video and image uploads into the system.

3.1. Website

A website acts as the GUI for the system. Users would upload images or videos to be sent to the fall detection system, then the fall detection system analyzes the input and returns results in the form of a video back to the website. The website possesses error checking to prevent problematic inputs, such as videos and images being uploaded simultaneously. It can also adjust its size to match whatever device screen it is displayed on.

4. Fall Detection

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

4.1. 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.

4.2. AlphaPose

AlphaPose has video processing built into its program and only needs to output to the fall detection system with the video frames being normalized to fit the 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. Its code is also convenient to work with as it has an implementation in google colab, a useful tool for collaboration.

4.3. 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.

4.4. 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.