FALL DETECTION SYSTEM

Justin Haryanto

Nhan Nguyen

CONCEPT OF OPERATIONS

REVISION – 2.0

30 April 2023

CONCEPT OF OPERATIONS

FOR

FALL DETECTION

TEAM <56>

APPROVED BY:

_____

Project Leader Date

_____

Prof. Kalafatis Date

_____

T/A Date

Change Record

Rev.	Date	Originator	Approvals	Description
1.0	15 Sep 22	Justin Haryanto		Draft Release
2.0	29 Apr 23	Justin Haryanto		Revision

Table of Contents

List of Tables

No table of figures entries found.

List of Figures

No table of figures entries found.

1. Executive Summary

A fall detection system is beneficial in providing life-saving help to fall victims by notifying emergency services and health personnel as soon as possible. Unfortunately, a lot of current programs require the use of wearables such as smart watches, therefore presenting a financial and physical burden and an inconvenience to those who wish to benefit from such a service.

The proposed system must be able to detect falling motion from humans within video footage at an appropriate frame rate and be able to distinguish between falls and other similar motions. It should also be able to reliably handle interference by the presence of more people and other moving entities. This system utilizes code from pose estimation that have proven successful.

In order to give inputs and receive outputs, a way to interact with the system will need to be developed. Since the detection program is completely online, access will come in the form of a website.

2. Introduction

Training AI to identify human actions is an area of interest for computer scientists with a variety of applications. Action recognition is beneficial in safety, where it is important to handle life-threatening situations quickly. Falls are especially dangerous in areas with an absence of people, or the victim is not treated quickly enough. The objective of this project is to provide an easy to use fall detection system that could potentially interface with existing video recording equipment.

The object of this project is to develop a fall detection system that determines if a fall has occurred after processing given video and image files. Also, a way to interact with the system will need to be developed.

2.1. Background

Fall detection would allow emergency services to help the victim as soon as possible. However, a lot of pre-existing fall detection systems may require extra data from sources such as smart watches which might not always be easily available.

2.2. Overview

Work in the development of fall detection software already exists, however, there is still room to improve speed and accuracy, as well as removing the need for wearables and smart devices. There also needs to be a way to interact with the system that is intuitive, simple, and attractive. With these changes, the safety of the people is both greatly improved and accessible.

2.3. Referenced Documents and Standards

- Fall Detection using Pose Estimation: https://towardsdatascience.com/fall-detection-using-pose-estimation-a8f7fd77081d
- An Overview of Human Pose Estimation with Deep Learning: https://medium.com/beyondminds/an-overview-of-human-pose-estimation-with-deep-learning-d49eb656739b
- History of Keypoint Detection in Computer Vision: https://medium.com/@SeoJaeDuk/history-of-keypoint-detection-in-computer-vision-be798a32ff4a
- Action Recognition: https://paperswithcode.com/task/action-recognition-in-videos
- What is Action Recognition: https://chooch.ai/computer-vision/what-is-action-recognition
- Action Recognition in Video: https://uwaterloo.ca/vision-image-processing-lab/research-demos/action-recognition-video
- Using Artificial Intelligence to detect and prevent falls: https://hospitalnews.com/using-artificial-intelligence-to-detect-and-prevent-falls/
- Fall Detection: https://viso.ai/application/fall-detection/
- Detecting & Preventing Falls with Artificial Intelligence: https://www.virtusense.ai/blog/detecting-preventing-falls-with-artificial-intelligence
- University of Rzeszow Datasets: http://fenix.univ.rzeszow.pl/mkepski/ds/uf.html
- Bournemouth University: https://falldataset.com/

3. Operating Concept

3.1. Scope

The fall detection system should be able to detect falls of humans when given video and image files. The user-interface should be able to communicate with the system directly and display the results back.

3.2. Operational Description and Constraints

The user uploads videos and image files to the website, which then gets sent to the fall detection system. Pose estimation will process the files by applying keypoint and bounding boxes on people. With this information, the system uses machine learning models to determine if a person has fallen, where it labels each frame where it detects a fall. The results are then sent back to the website and displayed for the user to see.

The main constraint is that test videos do not cover every potential scenario, which makes it more difficult to test the system for flaws.

3.3. System Description

There are two parts to the system: fall detection and externals.

The Fall Detection subsystem contains the pose estimation and fall detection components. Pose estimation focuses on detecting people in the video, then applying keypoint and bounding boxes on them. Fall Detection utilizes machine learning models to determine if a fall has occurred and outputs the result.

The Externals subsystem contains the datasets and website GUI components. Datasets focuses on the collection and creation of datasets for use in training and testing the model. Website GUI focuses on developing a website that allows the user to interact with the fall detection system.

3.3.1. Pose Estimation

Pose estimation identifies people in a video and overlays bounding boxes and key points for use in the fall detection system. Several programs were examined to determine their viability for our project: OpenPose, AlphaPose, and OpenPifPaf.

OpenPose needed training files that were no longer available. Fortunately, those files were found, and it was able to process a video. OpenPifPaf was promising because an author had used it in their fall detection system. However, difficulties running the required programs prevented its use. There was little trouble running AlphaPose, making it the most straightforward of the three.

Their accessibility makes OpenPose and AlphaPose viable candidates for our pose estimation system. Both also have the capability to identify multiple poses at 20 FPS, and their codes are adjustable to work with our fall detection system.

3.3.2. Fall Detection

Fall detection uses the information from the pose estimation system to determine if a person has fallen. This system can distinguish falls from similar motions and can function with poor video quality and unusual poses.

3.3.3. Output

The output can send information to other systems for future processing, but our system only displays a video and text showing which frames a fall was detected.

3.3.4. Datasets

Datasets focuses on collecting and processing datasets used to train and test the fall detection machine learning models.

Training sets were made using preexisting fall datasets from the University of Rzeszow and Bournemouth University, each set containing several hundred images. Only the relevant information was extracted from these sets and processing of the data was performed to make it usable with the system.

On the other hand, testing sets were recorded at the martial arts facility on campus due to the mats the practitioners used, which reduced the risk of injury when performing realistic falls. A total of 78 videos were recorded at 1920x1080 60 FPS, each clip several seconds long. These were compressed down to 480x640 15 FPS, making them more manageable for the system.

3.3.5. Website GUI

The ability to interact with the system using a UI is beneficial in improving the user experience, giving off a finished look, and simplifies the process of testing the system. It was desirable to keep the fall detection system completely online on Google Colab, so several UI solutions were looked at. In the end, using a website as the system's UI was the best option because it was able to upload to the system directly without using Google Drive, offered the most freedom in design, and allowed for communication with the system on any device.

Hosting of the UI was done using Ngrok, which provided the url to access the website. A combination of Flask, HTML, and CSS was used in the development and design of the website.

4. Scenarios

4.1. Senior Care

Falls are a concern for the elderly due to their high likelihood of injury. Implementation of a fall detection system in the camera systems of hospitals and homes would allow medical personnel to react quickly to a fallen individual.

4.2. Isolated Locations

Situations where a person is alone are common, such as when they live by themselves or are walking at night. A fall sustained in this state is dangerous, since seeking immediate help is possibly difficult for many reasons. Security cameras installed with fall detection would improve the safety of the community by automatically contacting emergency services. Such a system is also affordable if the area or house already has such surveillance systems.

5. Analysis

5.1. Summary of Proposed Improvements

This project seeks to create a system to detect falls regardless of body position, improving that ability through the use of datasets, and allow the user to interact with the system through a user interface.

5.2. Disadvantages and Limitations

It is difficult to run the code provided by previous authors, so testing various methods is time-consuming. Another issue is that our group may not have the necessary hardware to run the fall-detection system.

If the system is kept completely on Google Colab, then the challenges that may arise with working in an online environment (such as slow uploads, having a stable connection, and limited affordable software) will need to be dealt with.

5.3. Alternatives

An alternative to fall-detection software is for humans to monitor the cameras, set up patrols around an area, or have every potential victim carry a method of contacting emergency services (such as wearables or apps). Such approaches are expensive, unscalable, and prone to error.

5.4. Impact

Privacy is a possible concern of the public as our system would encourage the use of cameras in the community. As a result, companies would likely attempt to collect information and breach people’s privacy.

Another possible impact is the false sense of security that would arise from our program. There is the possibility of our system failing, and excessive trust in it would encourage some people to take more risks. Unless redundancies are put in place, future fall-detection systems may lead to more injuries.