**Title Slide (~Justin)**

* Howdy
* Justin introduce, Nhan Introduce
* And our project is to develop a fall detection system that only takes a video input

**Problem Statement (Justin)**

* According to the CDC, out of 3 million unintentional deaths since 1999, falls account for around 20% of deaths, or 0.6 million people
* Third leading cause of unintentional deaths in the US, after poison and motor accidents
* It is also known that elderly are at a higher risk of physical injury from falls due to physiological changes.
* If we implemented a fall detection system, first responders will be able to react to these falls faster, thus mitigating the consequences.

**Objective (Nhan)**

* “Our objective is to improve upon previous works on fall detection systems”
* **Read slide**

**Pose Estimation (Nhan)**

* “We will use pose estimation programs, which automatically track the poses of people using key points and bounding boxes for fall detection by evaluating sequences of poses.”

**Previous Literature (Nhan)**

* “We looked at existing fall detection systems and some pose estimation programs, but these had some shortcomings”
* “The fall detection system by Wei for example is accurate but was tested on a small data set. It also didn’t account for actions that looked like falls, such as tying shoes, and operates at a low FPS.“
* “Openpose and Alphapose are both capable of processing videos at 20 fps and have a wide range of capabilities, such as real time detection. However, they had issues with non-typical poses, overlapping bodies, and differentiating people from animals or statues.”

**System Overview x2 (Justin)**

* Our project consists of 3 subsystems: video processing, pose estimation, and fall detection
* Video processing converts video to a usable form for the pose estimation program, where it will identify people and apply certain attributes to them
* Taking the processed video, the fall detection subsystem determines if a fall has occurred and sends information to an external program
* Nhan will be working on the video processing and pose estimation portion of this project
* I will work on developing a machine learning based fall detection subsystem for the other part

**Video Processing and Pose Estimation (Nhan)**

* “For video processing and pose estimation, our system takes in video and converts it into a form usable for the pose estimation system. This system would be able to take in video of various qualities and finds people in them. It then applies bounding boxes and key points to their bodies, allowing us to evaluate their poses”

**Fall Detection (Justin)**

* The fall detection subsystem will take in the processed video from the pose estimation part and identify falls using machine learning
* After training with various datasets, the program will be able to distinguish falls from similar motions and unusual poses
* This system will output the number of falls detected, but later can be changed depending on its implementation
* We have considered deep learning, but decided against it due to its high resource requirements and complexity

**Dataset (Nhan)**

* “The data sets we are currently using come from the UR Fall Detection Dataset, Youtube, and our own videos. The UR data set was gathered by the University of Rzeszow and contains 30 falls and 40 daily activities. An example is shown here
* We will also create our own data sets using video from Youtube and our own recordings

**Execution Plan (Nhan)**

* **Read on slide**
* For our execution plan, we have goals that we have completed represented as green, goals in progress as yellow, and goals not started as grey. We have done our literature review on pose estimation and are working on experimenting with and improving pose estimation systems, as well as researching machine learning and gathering data sets. In the future, we aim to practice with machine learning algorithms, and experiment with our datasets.

**Validation Plan (Nhan)**

* **Read what is on the slide**
* We want the pose estimation system to process videos at 30 FPS. To test this, videos with different fps is fed into the program.
* For the fall detection and the whole system , we want them to detect more than 90% of the falls from the videos we provided.

**Status (Justin)**

* **Read the slide**
* As of right now, we have looked into the OpenPose and Alphapose estimation algorithms.
* After experimentation, we have determined that AlphaPose is the most usable out of the two.
* We also looked a bit into machine learning and search for some fall detection data sets