

1-1-2021

Automatic Fall Risk Detection based on Imbalanced Data

Yen-Hung Liu

Patrick C. K. Hung
University of Ontario Institute of Technology

Farkhund Iqbal
Zayed University

Benjamin C. M. Fung
McGill University

Follow this and additional works at: <https://zuscholars.zu.ac.ae/works>



Part of the [Computer Sciences Commons](#)

Recommended Citation

Liu, Yen-Hung; Hung, Patrick C. K.; Iqbal, Farkhund; and Fung, Benjamin C. M., "Automatic Fall Risk Detection based on Imbalanced Data" (2021). *All Works*. 4728.
<https://zuscholars.zu.ac.ae/works/4728>

This Article is brought to you for free and open access by ZU Scholars. It has been accepted for inclusion in All Works by an authorized administrator of ZU Scholars. For more information, please contact scholars@zu.ac.ae.

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Automatic Fall Risk Detection based on Imbalanced Data

YEN-HUNG LIU¹, PATRICK C. K. HUNG¹, FARKHUND IQBAL² AND BENJAMIN C. M. FUNG³

¹Faculty of Business and IT, Ontario Tech University, Canada (e-mail: yen-hung.liu, patrick.hung@ontariotechu.ca)

²College of Technological Innovation, Zayed University, Abu Dhabi, UAE (e-mail: farkhund.iqbal@zu.ac.ae)

³School of Information Studies, McGill University, Canada (e-mail: ben.fung@mcgill.ca)

Corresponding author: Yen-Hung Liu (e-mail: yen-hung.liu@ontariotechu.ca).

ABSTRACT In recent years, the declining birthrate and aging population have gradually brought countries into an ageing society. Regarding accidents that occur amongst the elderly, falls are an essential problem that quickly causes indirect physical loss. In this paper, we propose a pose estimation-based fall detection algorithm to detect fall risks. We use body ratio, acceleration and deflection as key features instead of using the body keypoints coordinates. Since fall data is rare in real-world situations, we train and evaluate our approach in a highly imbalanced data setting. We assess not only different imbalanced data handling methods but also different machine learning algorithms. After oversampling on our training data, the K-Nearest Neighbors (KNN) algorithm achieves the best performance. The F1 scores for three different classes, *Normal*, *Fall*, and *Lying*, are 1.00, 0.85 and 0.96, which is comparable to previous research. The experiment shows that our approach is more interpretable with the key feature from skeleton information. Moreover, it can apply in multi-people scenarios and has robustness on medium occlusion.

INDEX TERMS Anomaly Detection, Data Sampling, Fall Detection, Machine Learning, Pose Estimation

I. INTRODUCTION

WITH the advent of an ageing society, population ageing has become a common issue for many nations in the world. However, the most significant impact of population ageing on society is the rapid increase in medical support and long-term care demand. Sometimes an accident may cause a substantial financial burden on a family. According to the World Health Organization (WHO), falls are the second leading cause of accidental death globally, and 37.3 million falls require medical care each year. Among them, adults over 65 have the most life-threatening falls [34]. However, death is not the main result of immediately falling, as it can occur due to various complications caused by falls. Since the elderly often have a high prevalence of coexisting diseases, such as osteoporosis and organ function degradation, even a slight fall may cause great danger. Individuals that live alone, and are 65 and older, make up 24.6% of the Canadian population [43]. If an accident occurs, it is difficult for the elderly to be found. As a result, older adults are more prone to missing their golden treatment time. The shortage of caregivers promotes the health care system to automate. Developing an automatic fall risk detection system can effectively reduce the falls rate and the associated medical cost. Therefore, fall detection in the elderly has emerged as an important research

problem recently.

A. FALL RISK

Fall risk is a common threat that can affect all individuals, including the elderly or young children. Regardless of the victimized person, it can induce significantly harmful and dangerous results. Unfortunately, there are countless ways to assess factors that cause people to fall, but some may include one's poor eyesight, poor balance, use of medications that cause one to be drowsy, and more. Fall risk is to understand to take proper actions for prevention, as injuries associated with this accident can fracture one's physical health.

The significance of fall risk goes beyond identifying who is more at risk. Even though older adults are more susceptible to this injury, the importance is to assess the prevention methods and other valid reasons that cause fall risks. There are many ways to analyze and conduct assessments, such as studying human behaviour and the utilizing vital technological advancements to monitor movement and balance. This can depict one's actions, triggering specific causes and reasons that may lead to falls. Many studies show different assessment methods and evidence-based approaches to truly deliver accurate data regarding the causes of fall risks and proper ways of prevention.

Regarding the topic of fall risk, many recognized environmental hazards can be another cause for this mishap. We somehow do not have control over these factors. However, factors can be minimized through the practice of cautious behaviour. The flooring situation, such as rugs, carpets, mopped floors, and overly-packed areas, can lead to the dangerous falls among individuals. This shows that regardless of an individual's personal issues, falls can occur due to environmental factors.

Although the elderly are more at risk for falls, younger individuals, such as children, are also prime victims. Falls can happen just about anywhere, but children become vulnerable to their surroundings and environment, especially at a very young age. Depending on the child's physical surroundings, it can lead them to face serious danger, if they are not guarded or being taught how to take proper steps to be cautious. However, when looking at physical factors, as children are still producing more substantial muscular balance, this can factor in risks when it comes to falls. Also, those who are weak, and are diagnosed with a specific illness, are at risk of this injury.

Taking proper preventive measures for fall risk will allow patients to be safe from harm and away from hospitals. However, it is impossible to entirely prevent falls, as many possibilities could factor in some type of risk. This could be an individual's occupied space and area. It is essential to identify an unsafe place by looking at how it is structured and framed. Perhaps, staying in an area with handle bars and other tools to hold on could be a helpful tactic to prevent falls. It is an effective tool to assess one's physical surroundings, as anyone could be susceptible to this injury.

Fall risk is an important topic to discuss and assess, as it has many sets of factors that can cause it to happen, endangering many people's lives. Methods and other applications are utilized in many different settings to predict a patient's risk of falling. As stated before, elderly people are put at risk for this injury due to the changes in their body frame, physical and mental health, and cognitive alterations. The level of risk depends on the individual, their physical environment, as well as their lifestyle. Some individuals enjoy adventure, but living a safe lifestyle is always a better choice, as it can prevent falls and other medical issues.

Fear of falling is an issue that individuals are prone to feel after their fall incident. It can change one's perspective on their surroundings and may also be another reason for falls to occur again. [27] This is due to the trauma that can affect their mental health. Falls can happen anywhere. The injury of falls can result in broken body parts. However, fatality can be an outcome of severe falls. It is very important to shed light on the issue of falls, as it can be more severe than what others may imagine. It is a risk that many researchers are still assessing in order to identify the proper steps to prevent it from occurring. Although older people are more prone to this type of risk, children and other individuals can also be prime victims. Falls are serious, and this risk can be prevented once people understand their dangers. Fall risk is

an essential subject to study due to its strong relevance in the people's lives to this day. It is a risk that can be prevented, but individuals need to put the effort in to protect themselves.

B. CURRENT ISSUES IN FALL DETECTION RESEARCH

In the previous research, fall detection research can be divided into wearable-based methods, ambient-fusion-based methods, and vision-based methods. In wearable-based methods, they make use of accelerometers and gyroscopes to obtain relevant data and make predictions. Although this method can perform in real-time and has no privacy issues, the elderly's views on wearable devices will be uncomfortable, and inconvenient to wear for an extended period of time. As for ambient-function-based methods, the type of method combines multiple sensors to obtain environmental data and perform detection. This method's advantages result in fewer privacy issues and is less intrusive, but its performance is easily affected by external factors. In this case, the false alarm rate is high. As for vision-based methods, the advancement of image processing capabilities and Convolution Neural Networks (CNN) bring computer vision to a new level. However, although CNN can get high accuracy in many computer vision tasks, CNN is like a black box. The decision-making is hardly interpretable. The advantage of this method is more convenient and non-intrusive, but the disadvantages are privacy issues and interpretability.

Obtaining fall data often has privacy and moral restrictions, so most available fall datasets are recorded in experimental environments. For example, Shehroz et al. [10] questioned whether simulated fall data could represent real-life fall events. Since falls are rare events in real life, there should be a data imbalance problem.

C. RESEARCH CONTRIBUTION

In our paper, we propose a pose estimation-based fall detection algorithm. We use OpenPose's pose estimation algorithm to extract skeleton information and transfer them into interpretable features. Then, we train our model by machine learning methods. We used four public fall datasets, along with one gait dataset. We divide them into three different classes, *Normal*, *Fall*, and *Lying*. The class distribution is highly imbalanced. The purpose of this research is not only to propose a more interpretable vision-based method but also to evaluate the fall events in an imbalanced data perspective. Since surveillance cameras are everywhere, our approach is more suitable for the current society.

Our contribution can be summarized as follows:

- We evaluate our approach from a highly imbalanced data perspective that meets the real-world situation.
- Instead of using keypoint coordinates as features, we transfer the skeleton information into the interpretable feature.
- Our approach can work in multi-people scenarios, and even the person is occluded.

This paper is organized as follows. Section II describes the related works. We compare the pros and cons of the different-based approach. Section III discusses the dataset collections. We introduce four public fall dataset and one gait dataset. Section IV shows our fall detection architecture. We present the whole structure of our approach, feature preprocessing, and how to handle imbalanced datasets. Section V shows the experiment and the process in different scenarios. The experiment results will be shown in this section. Section VI discusses machine learning performance, the pose estimation-based approach's pros and cons, compared with the previous vision-based work, and further improvement recommendations. Lastly, section VII gives a conclusion.

II. RELATED WORK

Previous fall detection research can be divided into two categories: vision-based methods and sensor-based methods. Vision-based methods can also be divided into using Red, Green, and Blue (RGB) images or depth images, and sensor-based methods can be divided into wearable sensor-based methods and ambient fusion-based methods. In terms of the proportion of research, wearable sensor-based methods have the most significant proportion, followed by vision-based methods. The last is the ambient fusion-based method. We introduce the sensor-based method in the following subsections. Additionally, we will also discuss a more detailed review of the vision-based method.

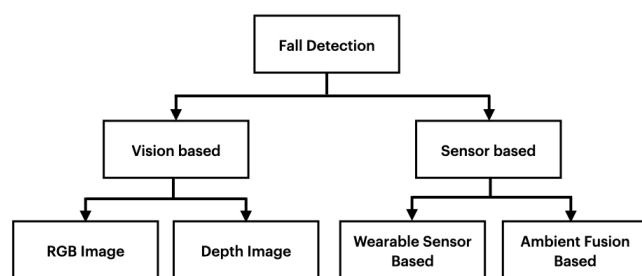


FIGURE 1. Taxonomy of fall detection methods

A. WEARABLE SENSOR-BASED

With the prevalence of wearable devices, an increasing amount of people have started to invest in wearable device research. Wearable devices' fall detection system uses the sensor to detect the body motion and status. The most commonly used sensor is accelerometer and gyroscope [31].

In the accelerometer method, Perry et al. [8] compared the method without the accelerometer and the method with the accelerometer. The experimental results show that the false alarm rate of the method without an accelerometer is relatively high. However, the position of the accelerometer is also significant. For example, Kangas et al. [9] placed the accelerometer in various body positions. The results showed that the false alarm rate was the lowest when placed on the

waist, while the false alarm rate was higher on the head and wrist. However, because their detection methods are mainly identified by setting the threshold, this method usually has a higher false alarm rate. In the gyroscope method, the gyroscope can get the angular velocity information to get the body orientation. This type of method is usually combined with the accelerometer for classification. In the study of Wu et al. [11], combining two kinds of sensors could further improve accuracy. A section of the research primarily focuses on mobile phones because these gadgets are necessary in daily human life. More sensors can be placed on mobile phones [38][39][40]. However, the main disadvantages include battery consumption and insufficient memory. In some cases, only high-end mobile phones are equipped with these sensors.

The advantage of the wearable device method is that the overall development is relatively mature. Their high accuracy can be used indoors and outdoors, and the setup is not complicated. The disadvantage of the wearable device is that power consumption and computing power are limited. Because this wearable device needs to be worn for an extended period of time, its weakness is that the elderly often forget to wear it. They may also use it in a way that makes them feel uncomfortable. Compared with the other two methods, it is the most intrusive.

B. AMBIENT FUSION-BASED

The ambient fusion-based method usually requires setting up various sensors around the environment. These sensors include vibration sensor, acoustic sensor, pressure sensor, infrared sensor, doppler sensor, and near electric field. Usually, these sensors are used to cooperate with other sensors.

Vibration sensors and pressure sensors are usually the most common methods. Vibration sensors are generally placed on the floor. For example, Werner et al. [12] believed that the vibration generated by a fall event is different from the Activity of Daily Living (ADL) event. The pressure sensor can be placed in any position, but the distance will affect the pressure's strength. Next, Daher et al. [13] used pressure sensors to form smart tiles, but this method can only detect the fall in which the acceleration is relatively large, but not for the slow fall. In the research of acoustic sensors, it is tough to obtain the data. Most of the fall data are obtained by rescue dolls. Then the hardness of a doll is different from that of a human. Moreover, everyone's weight is different in the real world. The method using the acoustic sensor can only detect hard falls.

The ambient, fusion-based approach's advantage is that it is less intrusive to people and has fewer privacy and security issues. However, fall detection can only be detected in a specific environment. In previous research, most of the research solely focused on single-person fall detection, and there was no way to cope with a multi-person environment. Although the ambient fusion-based approach combines more environmental factors, the actual situation often contains other unpredictable factors. Moreover, it is more complicated

in installation and setup. The high false alarm rate is a challenging shortcoming.

C. VISION-BASED

With the popularity of surveillance systems and computer vision advancement in recent years, vision-based methods have become a hot research field. We can detect the human body with different computer vision techniques. Traditional computer vision can extract the body contour with background subtraction and track the body movement with Optical flow [55][56]. Deep learning-based methods, object detection, can recognize the human and surrounded object efficiently [54][55]. Although detecting human objects is not hard with available techniques, identifying the activity such as falls becomes a challenging problem. Since the human body contains different parts which can move freely, some research focuses on the specific body part to design their methods [52][53][57]. Bosch et al. [52] use head, waist, and feet to extract the key features. Hazelhoff et al. [53] use the speed of the head to identify fall events, which has fewer occlusion problems due to the fact that the head is visible more frequently. Also, the camera can monitor a wide range of areas, and it is contactless. In general acceptance, the vision-based method is the most favourable. Regarding cameras, different types of camera sensors can extract different features from the image. The image data acquisition methods can be divided into the following two categories: RGB image and depth image.

1) Depth Image

The most common research equipment for depth images is the Microsoft Kinect sensor. Kinect is a low-cost device that uses an infrared projector combined with an RGB camera to extract depth information. Kinect is used to detect human body movement, and light conditions do not affect its performance. In the study of Volkhardt et al. [14], they installed Kinect on the robot and used different feature extraction and classification methods to determine fall events. Among the classification methods, SVM performed the best. Apichet et al. [15] proposed a new bounding box framework called Directional Bounding Box (DBB) based on a depth camera and Microsoft Kinect. Kinect's keypoint and depth information rotated the key point to get the appropriate angle to form the DBB. The identification method was easy to cause false positives due to different camera angles using the height and width ratio, to identify in the past. But now, the combined depth information, height-width-depth ratio, and center of gravity are used to identify the fall event. Kinect can also extract the skeleton information. Thi-Lan et al. [59] used skeleton information extracted from Kinect as features. They made use of the SVMs as a classifier to identify the fall events. Even though the Kinect sensor obtains the most information, there are still some drawbacks. Kinect can work in a dark environment but is very sensitive to sunlight. This indicates that it is unsuitable for the outdoor environment. In

addition, the distance detection depth is limited, making it challenging to monitor broad areas [58].

2) RGB Image

RGB cameras are relatively cheap and easy to set up. RGB cameras have a wider field of view, so most surveillance cameras are RGB cameras. Although the depth information is lacking, the most common vision-based approaches are the RGB cameras. Traditional computer vision methods are usually used for background subtraction, capture body contour, and use tracking techniques in head and shape change. In addition, machine learning methods are used as classifiers. The emergence of CNN has brought feature extraction to a new level. One of the CNN applications is object detection. One image can include multiple objects, which belong to different classes. The object detection algorithm can use the bounding box to capture the object and identify its class, which is widely used in facial recognition and defect detection. Some well-known object detection algorithms are You Only Look Once (YOLO) and SSD-MobileNet. Both can work in real-time performance. Kun-Lin et al. [16] used YOLO V3 to design their fall detection system. They focused on detecting general fall events and falling events from sitting to standing posture and events where the body is blocked after a fall. Fall accidents in the elderly, such as sitting down, getting up, and leaving the chair, account for the majority. Since YOLO V3 can detect other objects, they consider the relationship between humans and chairs. In their fall detection system, the first uses YOLO V3 to detect people and chairs. Then, it uses Continuously AdaptiveMeanShift (Camshift) to track the human body and build fall detection algorithms constantly. The system can handle a situation where the chair blocks an individual's body. Kiran et al. [60] also, used YOLO as a detection method, as it is based on the height and width ratio of the bounding box to identify fall events. However, humans are a particular category in computer vision. Although object detection can capture the human body, the information is not enough to understand human motion. The performance of this approach heavily relies on the camera angles. To overcome this problem, pose estimation can be the solution.

3) OpenPose

Pose estimation is a computer vision technique used for identifying human postures. Pose estimation can detect a human's body skeleton, which can be used to identify human activity. OpenPose [62] is a well-known pose estimation method used a lot in human action recognition. It can efficiently detect multiple people, and the processing time stays stable when the number of people in the image increases. The whole OpenPose model structure includes two stages. First, they extract the features from the image through CNN (10 layers of VGG-19). Visual Geometry Group 19 (VGG-19) is one of the CNN architectures. Then, they use the image features as input and send it to the first stage. In the first stages, the CNN predicts the part affinity fields. Part affinity is a 2D vector, which associates the different body parts. Part affinity

fields can make the model understand the orientation of the limb, and it can help estimate the body part in the second stage. In the second stage, the CNN predicts the confidence map with part affinity fields. Confidence maps detect each individual's body part location. With the part affinity fields and confidence maps, OpenPose can efficiently form a body skeleton.

Compared to the bounding box, the skeleton is more suitable for detecting fall events. Therefore, some researchers start to use OpenPose as input to boost their research. Chen et al. [17] used OpenPose to extract the human skeleton information based on the unstable center of gravity and symmetry collapse when a fall event happens. They proposed three key features to identify a fall. These three parameters are the descent speed of the hip joint center, the centerline's angle of the human body to the ground, and the human body's height-width ratio. In their research, we found that the rate of speed descent of the hip joint center significantly influences the prediction of fall events. In addition to general fall events, they also considered whether people could stand up independently after falling. Guangmin et al. [18] used OpenPose plus Single Shot MultiBox Detector (SSD)-MobileNet for their fall detection. The function of SSD-MobileNet is to avoid Openpose's false detection of non-human objects. As for how to identify a fall event, they used Support Vector Data Description (SVDD) classification. The experiments show that their method can effectively reduce the false positive rate in a complex environment [18]. Zhanyuan et al. [19] used OpenPose to perform preprocessing first to obtain pictures with keypoints, and the keypoint's coordinates. Then, they put the two into different model identification. In images with keypoints, they used the VGG-16 for transfer learning and then binary classification to identify fall events. In the keypoint coordinate, Support Vector Machine (SVM) and Gaussian Kernel are used for identification. Finally, combining the two results effectively improved sensitivity. Most of the fall data is mainly videographed, so it can also be regarded as time sequence data. Sungil et al. [20] used OpenPose extract skeleton data and then used Long Short-Term Memory (LSTM) to determine fall events. Their approach extracts the coordinates of the shoulders, buttocks, knees, and ankles and the acceleration of these parts as features. In their fall detection, they also divided fall events into *ADL*, *Falling*, and *Lying*. Their experiments, the acceleration of a specific part, and the two states of *Falling* and *Lying* have the highest impact on accuracy.

In the vision-based method, the advantage of using depth images is that more information can be obtained, but people have to stand within a certain distance. For example, Kinect can only detect people within 3 meters. However, the RGB camera detection range is vast, and most current surveillance systems use RGB cameras. The advantage of the vision-based method is that it is more convenient and less intrusive. However, the disadvantage of this method is that it requires more computing resources and it acquires privacy issues. Moreover, light, occlusion problems, and different

background conditions have constantly challenged computer vision.

Regarding three different based approaches, each of them has its own pros and cons. Wearable sensors can detect detailed human status but are most intrusive. Ambient sensors are easily affected by the environment. Vision-based can capture multiple people but is easily influenced by occlusions. Despite the data acquisitions being different, fall event classifiers are based on a threshold or machine learning. Since people have different body shapes, setting the threshold is challenging, and the same threshold may not work on different people [10][36][37]. Therefore, some researchers are focused on machine learning methods, which have better accuracy [31]. At present, most of the fall detection products on the market are mainly sensor-based devices [37]. Depending on the application and use case, we assume vision-based approaches have more potential. Depending on the application and future, we undertake vision-based approaches that have more potential. Vision-based approaches have a wide range of views that can monitor multiple people. In general situations, we identify the fall events through vision. This is an intuitive and more natural element for a detection system.

Therefore, we consider the above drawbacks during designing our own approach. First, we use a vision-based method, which is less intrusive and has more potential in future works. Second, we use skeleton information as features instead of bounding boxes, which understands a more detailed human posture. Third, we use OpenPose instead of Kinect, because we want to apply it on a broader view. Finally, the occlusion problem has less influence on us since even partial body parts are missing. We can still predict through the remaining skeleton.

III. DATASET PREPARATION

Some previous research uses their simulated fall data to train the model, performing great evaluation results. However, when we use other fall datasets to evaluate the performance, the performance drops significantly. This indicates that single simulated fall data did not have enough understanding of variations of fall. Thus, in our approach, we utilized the most commonly used University of Rzeszow (UR) Fall dataset [21] and three other public datasets related to falls and divided them into ADL events and fall events. Moreover, because the abnormal gait has a high correlation between falling events, we also collected the gait data from "Aplicaciones de la Visión Artificial" (A.V.A) Multi-View Dataset [25] to enhance our model's robustness.

A. UR FALL DETECTION DATASET

UR Fall Detection Dataset [21] contains 70 videos, including 30 fall events and 40 ADL events. The video resolution is 640 x 480, and the Frame per Second (FPS) is 30. For fall events, the videos are recorded with 2 Microsoft Kinect cameras, and one is parallel to the floor, while the other is mounted on the ceiling. Fall situations include falls while walking and falling from a chair, and the direction is mostly side falls and forward

TABLE 1. All Datasets Description

	UR-Fall	UMA-Fall	SisFall	Multiple Camera Fall	AVAMVG
Data type	RGB Video (640x640, 30fps)	RGB Video (854x480, 30fps)	RGB Video (1920x1080, 30fps)	RGB Video (720x480, 30fps)	RGB Video (640x480, 30fps)
Number of Fall and ADL	Fall:30 ADL:40	Fall:3 ADL:8	Fall:15 ADL:19	Fall:192	ADL:1200
Camera View Point	1	1	1	8	6
Scenario	Side Fall, Forward Fall	Side Fall, Forward Fall, Backward Fall	Side Fall, Forward Fall, Backward Fall	Side Fall, Forward Fall, Backward Fall	Walking

falls. As for ADL events, the videos are recorded with one camera parallel to the floor. ADL situations include sitting, bending, squatting, grabbing something from the floor, and lying on the bed. In our application, we only use videos that are parallel to the floor.

B. UMAFALL

Universidad de Malaga (UMA) fall dataset [22] contains 11 videos, including 3 fall and 8 ADL events. The video resolution is 854 x 480, and FPS is 30. Fall situations include side fall, forward fall, and backward fall. ADL situations include bending, hopping, jogging, sitting, and walking. Unfortunately, some video angles cannot fully capture the whole body of the tester, so we did not include those videos in our dataset.

C. SISFALL

SisFall dataset [23] contains 34 videos, including 15 fall events and 19 ADL events. The video resolution is 1920 x 1080, and FPS is 30. It includes a more detailed scenario for fall events, including slip, trip, fainting, etc. The fall direction includes side fall, forward fall, and backward fall. As for ADL events, it includes walking, jogging, walking up and down the stairs, etc. The way they recorded the video is different from other datasets. For example, they moved the camera parallelly and followed the tester instead of setting the camera in a fixed place.

D. MULTIPLE CAMERA FALL DATASET

Multiple-Camera fall datasets [24] used 8 cameras to capture different angles. The author did not divide the video into two categories. Instead, the videos start with the tester performing some ADL events, and then the tester falls during the sessions. The ADL events contain walking, housekeeping, and activities that are similar to fall. The fall events have side falls, forward falls, and backward falls. The factors include inappropriately getting up, sitting down, or losing balance. The whole dataset contained 24 different scenarios and was recorded by 8 cameras. Therefore, in total, the number of videos is 192.

E. AVAMVG DATASET

The AVA Multi-View Dataset (AVAMVG)[25] for gait recognition used 6 cameras to capture different angles. The video resolution is 640x480, and FPS is 30. This dataset only

collects the walking event, which is the most common event in our daily life. For the tester, there are 4 females and 16 males participating in 10 different walking activities. Some activities are walking in a straight path, and some are walking in a curved path. Due to the windows' ambient illumination, the video's brightness is different based on the camera's location. The total number of the video is 1200.

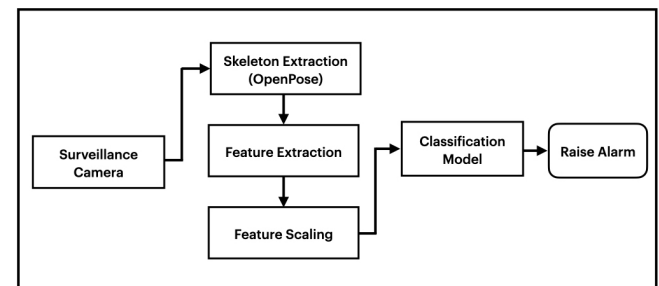
Referring to Table 1, we have 1507 videos in total. 240 are fall videos, and 1267 are ADL videos. The number of videos is imbalanced, which meets the real-world situation. However, since the length of each video is not the same, we use the number of frames to represent the ratio of the data in the following sections.

IV. OUR FALL DETECTION SYSTEM

In this section, we explain the whole process of our approach, which includes data preprocessing, feature extraction, and model selection.

A. ARCHITECTURE

In our approach, our image data are RGB images. First, we use OpenPose to extract the skeleton information from the images. Then, we further perform the feature extraction and feature scaling on the skeleton information to help the model learn more effectively. Regarding the classification model, since the fall events need to be dealt with urgently, we use the machine learning approach to classify instead of the deep learning approach. When the classification model thinks that a fall event has occurred, an alarm will sound in the decision part. Otherwise, if everything is considered normal, it will continue to identify the next frame. Figure 2 is our fall detection architecture.

**FIGURE 2.** Fall Detection System Architecture

B. PREPROCESSING AND FEATURE EXTRACTION

In most machine learning tasks, data preprocessing and feature extraction are the most important stages. The importance of features is more significant than model selection. Therefore, in the preprocessing part, we use pose estimation to extract skeleton information. In some vision-based research, they use Kinect to extract the skeleton feature, but the detecting distance is one of the drawbacks for Kinect. Therefore, we use a deep learning-based method called OpenPose as our pose estimation method. OpenPose is robust in multiple people scenarios. Furthermore, since the skeleton size may be different in a variety of distinct distances, we further extract the key features from skeleton information to minimize the effect of distance.

1) Pose Estimation - OpenPose

To extract the skeleton information from RGB images, we use OpenPose as our pose estimation method, an open-source library developed by Cao et al. [26][62] from Carnegie Mellon University (CMU). OpenPose is a 2D pose estimation method that effectively detects 25 body parts and forms a body skeleton. More detailed information is shown in Figure 3. OpenPose is the bottom-up method that detects the body part first and then forms the skeleton in terms of time performance. It performs well in environments with many or few people or even different light conditions. Therefore, the runtime performance does not interfere with the increase of people.

The most important thing is that OpenPose's detection distance is not limited compared with Kinect, which meets our fall detection system requirement very well.

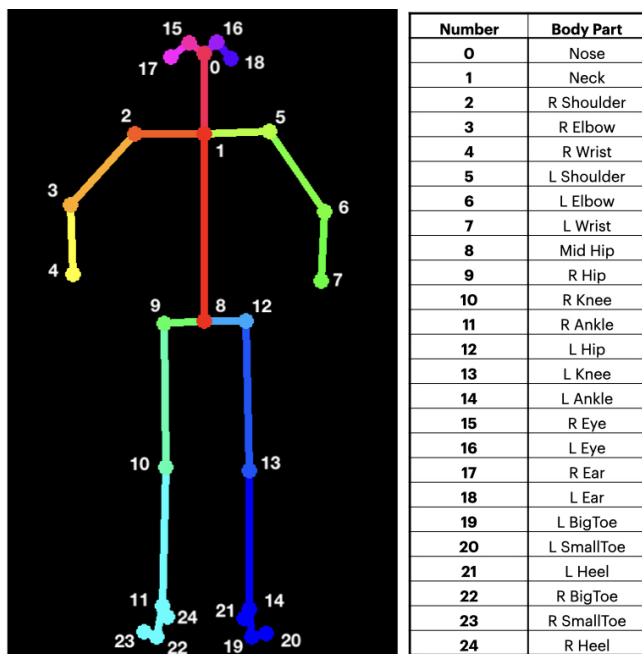


FIGURE 3. OpenPose Keypoint Diagram and Keypoint Table

In our approach, we only use 0-14 body keypoints to

represent the human body. Furthermore, we exclude 15-24 body keypoints because even if these points are excluded, we can still see whether the body tends to fall. In this case, we consider these keypoints to be unrelated to a fall event.

2) Feature Extraction

Physical characteristics are significant in the past fall events, such as body deflection and acceleration during the fall. Therefore, besides extracting the skeleton feature, we further extract the physical features from the skeleton information. By doing a further feature extraction, the model can learn the difference more effectively during the training, and the whole prediction becomes more interpretable. Next, we construct the features, including ratio, distance, acceleration, and deflection.

Ratio Feature: When the fall events occur, the most apparent feature is the change in the posture of the human body. In most ADL events, the human body skeleton's height is larger than the width. However, when the fall events occur, the skeleton's height tends to decrease, and the skeleton's width increases. Thus, we use the height and width ratio (*HW ratio*) as the feature to represent the body outline. Nonetheless, when the direction of the fall faces the camera, the width may not increase. Therefore, we refer to the *Spine ratio* from Han et al. [30] to supplement the *HW ratio*. For example, the length from keypoint 1 to keypoint 8 is spine-length, and the length from keypoint 9 to keypoint 12 is waist-length. Therefore, we use spine length divided by waist-length as the *Spine ratio*. In this way, because the spine-length decreases and the waist-length stays stable when the fall direction faces the camera, the *Spine ratio* decreases. Thus, the *Spine ratio* can be used as a feature to identify the fall.

The detailed calculation process is as follows.

$$HW\ Ratio = \frac{Height}{Width} \quad (1)$$

$$SpineRatio = \frac{Spine\ Length}{Waist\ Length} \quad (2)$$

In Figure 4, the *HW ratio* changes rapidly when the Fall event occurs. In Figure 5, although the test person falls facing the camera, and the *HW ratio* does not drop significantly, the *Spine ratio* drops. The *Spine ratio* is 2.24, 2.09, 1.35, and 1.01, from left to right, respectively.



FIGURE 4. Simple illustration of HW Ratio

Distance Feature: There is usually a significant difference in height between the head, hips, and feet in everyday movement. However, the vertical distance between the hips,

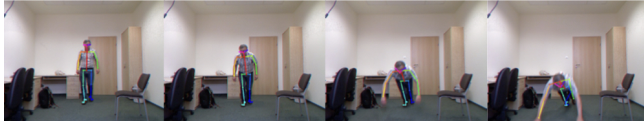


FIGURE 5. Falling direction faces the cameras

head, and the ground, decreases significantly when fall events occur. In some cases, the head and hips are even parallel to the feet. Therefore, we use the midpoint of keypoint 11, 14 to represent feet location. Then we calculate the vertical distance from the keypoint 0 (Neck) to the feet and the vertical distance from the keypoint 8 (Mid Hip) to the feet as features to further identify the body's current status. The detailed calculation process is as follows.

$$\text{Neck to Feet Distance} = (y \text{ of Feet}) - (y \text{ of Neck}) \quad (3)$$

$$\text{Hip to Feet Distance} = (y \text{ of Feet}) - (y \text{ of Hip}) \quad (4)$$

Acceleration Feature: In wearable sensor-based fall detection, the accelerometer is an effective indicator. The occurrence of fall events often causes a dramatic acceleration change in a short time. Therefore, we extract the acceleration feature from the skeleton information. In the research of Kangas et al. [28], they put the accelerometer in multiple body parts and tested the performance for fall detection. In their experiment, the position of the head and waist has the highest sensitivity and specificity. Thus, in our approach, we calculate the acceleration of keypoint 0, 1, 8 (Nose, Neck, Center of Waist). We only calculate the change of negative, vertical acceleration. The detailed calculation process is as follows.

$$\text{Head Acceleration} = \frac{(y \text{ of Head}) - (y \text{ of preHead})}{5} \quad (5)$$

$$\text{Neck Acceleration} = \frac{(y \text{ of Neck}) - (y \text{ of preNeck})}{5} \quad (6)$$

$$\text{Hip Acceleration} = \frac{(y \text{ of Hip}) - (y \text{ of preHip})}{5} \quad (7)$$

Deflection Feature: When a fall event occurs, the angle between the body and the ground usually changes significantly, in addition to the change in the body's contour. Therefore, we used the deflection angle proposed by Han et al. [30] to capture the deflection feature. To calculate the deflection angle of each body part, we use the gravity vector and 6 body vectors. The gravity vector is any vector parallel to the y axis. Body vector is a vector formed by two keypoints; spine vector is keypoint 1 to 8; waist vector is keypoint 9 to 12; right left (RL) calf vector is keypoint 10 to 11 and keypoint 13 to 14; and RL thigh vector is keypoint 9 to 10 and keypoint 12 to 13. Using the cosine function, we can find the angle between the body part and the ground, as shown in Figure 6.

In addition to the above vector, we also measure the whole body tilt angle. We calculate the angles of the neck point with mid-hip, mid knees, and mid ankles, respectively. Finally, we choose the smallest angle as the body tilt angle. As shown in Figure 7, since the mid ankles can form the smallest angle, we use this angle to represent the body's tilt angle.

The detailed calculation process is as follows.

$$\text{Deflection Angle} = \cos^{-1} \left(\frac{\overrightarrow{Body} \cdot \overrightarrow{Gravity}}{|\overrightarrow{Body}| \times |\overrightarrow{Gravity}|} \right) \quad (8)$$

$$\text{Body Tilt Angle} = \tan^{-1} \left(\frac{y \text{ of Neck} - y \text{ of BodyPart}}{x \text{ of Neck} - x \text{ of BodyPart}} \right) \quad (9)$$

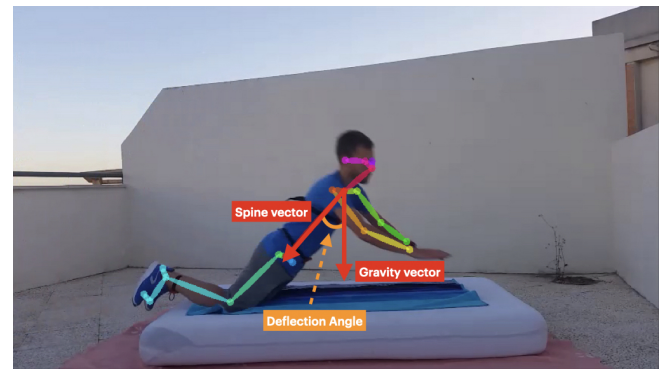


FIGURE 6. Spine Deflection Angle [22][30]

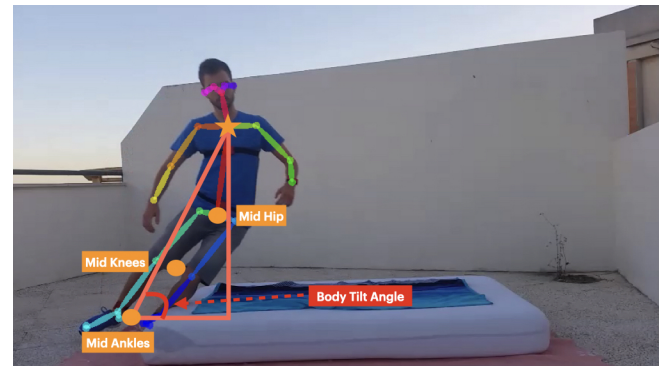


FIGURE 7. Body Tilt Angle [22][30]

C. IMBALANCED DATA HANDLING

Data imbalance is a common problem in machine learning tasks. Most machine learning algorithms assume that the data is uniformly distributed. However, in a real-world problem, the data is often unevenly distributed. When the number of data is extremely different, the minority of classes are not represented in the model. This results in the majority of the class dominating the learning performance. To solve the imbalanced data problem, we use standard methods, including sampling methods and anomaly detection.

1) Sampling Methods

Sampling methods are mainly used to balance data distribution and can be divided into Oversampling and Undersampling. Oversampling can increase the size of rare samples to achieve a balance. Oversampling can also be divided into random sampling and synthetic sampling. The data in random sampling may repeatedly appear, so that it may cause overfitting. Synthetic sampling can use existing data to generate more samples, thereby avoiding overfitting. Undersampling can reduce the number of samples to achieve a balance. But the disadvantage is that the data is not complete, so the model can only learn a part of the whole. Thus, we usually use undersampling when all the class data are insufficient. Although our data is imbalanced, the number of each class is sufficient, so we did not use undersampling methods. In our approach, we use one kind of random sampling and three kinds of synthetic sampling.

2) Anomaly Detection

Isolation Forest: Isolation forest is an unsupervised and non-parametric method suitable for continuous, numerical data. Isolation forest assumes that the outlier is sparsely distributed and far from the high-density data group. The idea of the method is that if the data is normal, you need more decision trees to separate the data. Conversely, if the data is abnormal, you can separate the data with fewer decision trees.

One Class SVM: One Class SVM is an unsupervised method. It only uses one class of data to train the model. Utilizing the majority class to train the model, the model can learn a decision boundary and use that boundary to determine whether the new data is similar to the training data. If the boundary is exceeded, it is regarded as an anomaly. The kernel function we used is Radial Basis Function (RBF), which can effectively project features to high dimensions, and make data have a good aggregation. Thus, this method usually performs well when the data dimension is high.

Elliptic Envelope: Elliptic Envelope is an unsupervised algorithm. This algorithm assumes that the distribution of the data conforms to the Gaussian distribution. By estimating the covariance, this method encloses the data in an oval area. Any data outside this area will be identified as an outlier. It performs well when the data conforms to the Gaussian distribution.

D. CLASSIFICATION MODEL

K-Nearest Neighbours: K-nearest Neighbours (KNN) is a well-known pattern classification algorithm. The reason we use KNN as one of our classification models is that KNN is widely used in previous research, no matter the approach is sensor-based or vision-based. It is one of the mature, and straightforward supervised machine learning algorithms. It is a method of classification based on the local distance feature. KNN is used on both classification and regression problems. In regards to classification problems, KNN begins with calculating the distance between a predicted datapoint. Then, it surrounds the data points and collects the closest data

points of “k.” Second, KNN determines the class based on the most common class in the “k” closest data points. The predicted value is the average value of the “k” closest data points in the regression problem. K value is typically small.

In Figure 8, the most common class in the 3 closest data points is Class A, when $K = 3$. The most common class in the 10 closest data points is Class B, when $K = 10$. Thus, the predicted class can be different, depending on the K value.

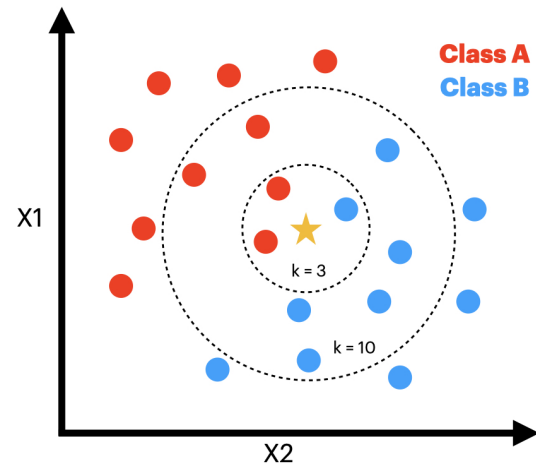


FIGURE 8. Simple illustration of KNN

Support Vector Machines: Support Vector Machines (SVMs) is a supervised machine learning algorithm, and it is widely used in industrial applications. SVMs use training data to find a decision boundary, called an optimal hyperplane. The optimal hyperplane separates the different classes with a possible wide gap, known as the highest margins. In addition to linear analysis cases, the SVMs are well-known in linearly inseparable cases. The SVMs’ kernel functions can effectively transform the inseparable features from a low dimension to a high dimension, where it becomes easier to separate with a hyperplane. Due to the optimal hyperplane, the SVMs have robustness on sparse data.

Boosting: Boosting is an ensemble machine learning algorithm that is an effective and widely used supervised learning method. It iteratively reweights the training data and trains the weak learners. Also, it finally consists of all the weak learners into strong learners. The false predicted data can gain more weight in the following training so that the next weak learner can improve the previous weak learner. Therefore, the boosting method is considered to be a practical approach when underfitting happens. In our approach, we use two boosting methods, AdaBoost and XGBoost [29].

Adaptive Boosting (AdaBoost) is used with various machine learning algorithms to improve accuracy. It weights the training sample and trains multiple weak classifiers to compose one strong classifier, though the weak classifier can only become a bit better than random guessing. First, if we have N numbers of data samples in the weighting process, we equalize each data sample’s weight to $1/N$. After

TABLE 2. Experiment Configuration

Experiment Configuration	
System	Ubuntu 18.04
CPU	Intel(R) Xeon(R) Silver 4114 CPU @ 2.20GHz
GPU	Tesla P100-PCIE-16GB
RAM	16GB
Cuda Version	11.1.1
Cudnn Version	8.0.5.39

the first weak classifier training, the accurate predicted data sample's weight is deducted, and the false predicted data sample's weight is increased. Second, the next weak classifier is trained with reweighted data samples, focusing on hard-to-classify data and improving accuracy. We are finally composing all the weak classifiers with specific weight to one strong classifier. AdaBoost is a high-precision method, and it is not easy to overfitting.

XGBoost [29] is a tree ensemble model. XGBoost uses additive training to preserve the model and attach a new tree in each iteration, to improve the previous tree. Comparing to AdaBoost, it uses weight to strengthen the hard-to-classify data. On the other hand, XGBoost uses residual to improve the accuracy. Although both algorithms follow the boosting concept, XGBoost has made significant improvements in algorithm optimization and system optimization, so it has robust scalability, speed, and accuracy. Moreover, XGBoost is good at handling missing values and has features of automatic feature selection. As a result, it is a popular and widely used method in most data science projects and Kaggle competitions.

V. EXPERIMENTAL RESULTS

A. EXPERIMENT CONFIGURATION

We implemented the platform in Ubuntu 18.04. Python 3.8 is the main coding language, and the developing IDE is jupyter notebook. Machine learning, sampling method, and anomaly detection are based on the scikit-learn package. Pose estimation is based on the OpenPose package. We use OpenCV to display and process every image. Since OpenPose needs GPU resources to speed up the image processing time, we use the Tesla P100-PCIE-16GB for our GPU resources. More detailed information is shown in Table 2.

B. DATASET

1) Data Labelling

To train our model, we manually labelled the data that we collected. We divided the data into three classes, *Normal*, *Fall*, and *Lying*.

The definition of *Lying* indicates a person's posture is settling on the ground or lying on the bed. We all identify as *Lying* class. The definition of *Fall* refers to the interval between 'Normal' status that changes to *Lying* status. Regardless of whether the action taken is sitting or walking, this interval belongs to the *Fall* class. This is only as long as the

TABLE 3. Data Proportion

	Normal	Fall	Lying
Frames	280770	8126	20841
Percentage	90.65%	2.62%	6.73%

person ends up lying on the ground. We do not have many restrictions for the *Normal* class. Any status that does not fall into *Lying* and *Fall*, belongs to the *Normal* class. Therefore, some easily misidentified actions such as squatting, sitting, and jumping, belong to the scope of *Normal*

After the data labelling, the *Normal* class has 280770 frames, the *Fall* class has 8126 frames, and the *Lying* class has 20841 frames. Thus, the data is highly imbalanced. Table 3 is the result of the data proportion.

2) Data Preprocessing

In some cases, when body occlusions happen, OpenPose cannot effectively detect every body keypoints, which causes the missing value of the feature and outlier in ratio features. To exclude the outlier in *HW_ratio* and *Spine_ratio*, we only keep the ratio data in the 10-90 percentile range. To make the model understand whether the feature is missing, we add new columns to the corresponding feature to indicate whether the feature is lost. For example, if the *Spine_ratio* is missing, the corresponding new value of *Have_Spine_ratio* is 0. On the other hand, if the *Spine_ratio* is not missing, the value of *Have_Spine_ratio* is 1. We do not have corresponding columns for *HW_ratio*, *Head Acc*, *Neck Acc*, and *Spine Acc* because those features have no missing values. The last step is to fill in the missing value. We replace the missing value according to the data's label. If the label is *Normal*, we replace the missing value with the mean value of the *Normal* class and so on. Figure 9 is the illustration of our missing value handling approach.

3) Split Training Data and Testing Data

To evaluate whether our model is overfitting or not, we use 80% of data for training and 20% of data for testing. Thus, we have three different experiments. First, we trained the model directly with training data to assess the model performance on imbalanced data. Second, we use the oversampling method on training data to balance out the number of data in each class. Finally, we evaluate the performance with testing data that still stays imbalanced.

C. EVALUATION METRICS

A good evaluation metric can evaluate the model's performance effectively. However, when the data is skewed, the standard evaluation metrics, such as accuracy and error, can lead to misleading information. For example, suppose we have 5 positive data and 95 negative data. In that case, the model will receive a 95% accuracy rate when predicting every data to negative data, regardless of whether it can predict any positive data. Therefore, in the unbalanced dataset, it

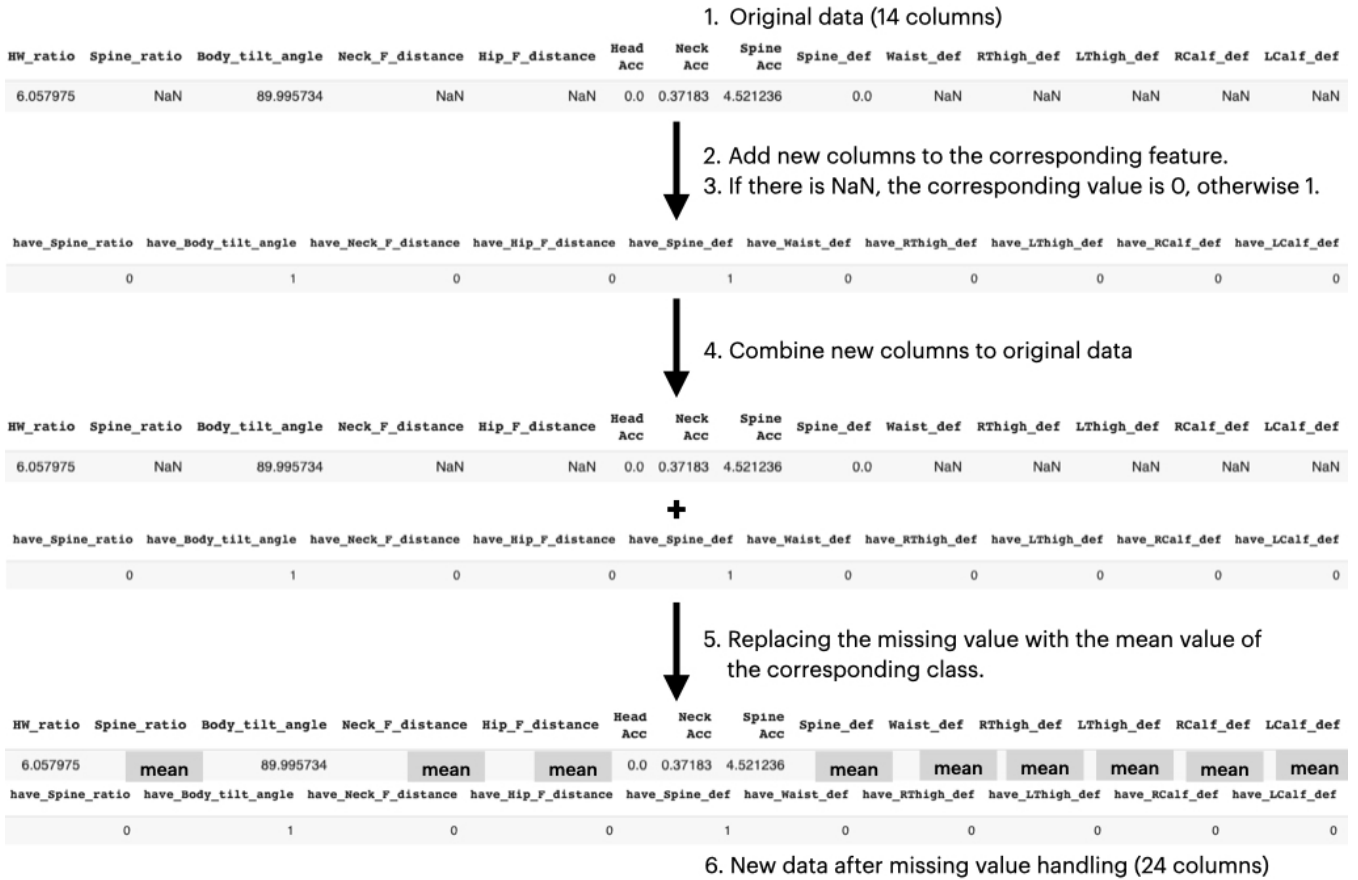


FIGURE 9. Missing value handling illustration

is more important to predict the minority of data successfully. Based on Vallabh et al. [31]’s fall detection review, we evaluate our approach via Precision, Recall, F1 Score, Specificity, and Area Under the Curve (AUC) of Receiver Operating Characteristic (ROC), which are commonly used in previous Fall detection research.

Most of the values can be calculated by the following outcomes. True Positive (TP), the model predicts fall events to occur, and fall events occur. False Positive (FP), The model predicts fall events to occur, and no fall events occur. True Negative (TN), the model predicts no fall events to occur, and no fall events occur. False Negative (FN), the model predicts no fall events to occur, but fall events occur.

Precision can represent how much of the data is predicted to fall, as they eventually lead to a fall. Recall can represent how much the model can predict the actual fall. Thus, if we want the model to identify every possible fall event, the higher the Recall rate, the better.

F1-Score is the combination of precision and recall. The specificity can represent how much the model can predict actual ADL events. Thus, we can check the recall and

specificity to ensure the model learns all the different label characteristics.

ROC curve is the combination of recall and specificity. The value of AUC is the area below the ROC curve. Generally, the AUC score is between 0.5 and 1. The larger AUC, the better the classification performance.

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

$$Recall = \frac{TP}{TP + FN} \quad (11)$$

$$F1\ Score = \frac{2 \times (Precision \times Recall)}{Precision + Recall} \quad (12)$$

$$Specificity = \frac{TN}{FP + TN} \quad (13)$$

D. ANALYSIS OF EXPERIMENT RESULT

1) Machine Learning without Oversampling

In the beginning, we test the performance of each model on an imbalanced dataset. In the proportion of our dataset,

TABLE 4. Different machine learning model performance

Normal	Precision	Recall	Specificity	F1-Score	AUC
KNN	1.00	1.00	0.96	1.00	0.99
AdaBoost	0.99	0.99	0.95	0.99	0.97
XGBoost	0.99	1.00	0.92	1.00	1.00
SVMs	0.99	1.00	0.94	1.00	1.00
Fall	Precision	Recall	Specificity	F1-Score	AUC
KNN	0.92	0.73	1.00	0.81	0.94
AdaBoost	0.67	0.66	0.99	0.66	0.83
XGBoost	0.91	0.52	1.00	0.66	0.98
SVMs	0.81	0.63	1.00	0.71	0.98
Lying	Precision	Recall	Specificity	F1-Score	AUC
KNN	0.94	0.98	0.99	0.96	1.00
AdaBoost	0.92	0.93	0.99	0.93	0.96
XGBoost	0.91	0.97	0.99	0.94	1.00
SVMs	0.92	0.97	0.99	0.94	1.00

Normal accounts for 91%, Fall accounts for 2.6%, and Lying accounts for 6.4%. Training data and testing data maintain the same ratio.

As experiment result shown in Table 4, each model has good classification performance on the Normal class, and both precision and recall are above 0.99. Lying class has the second-best performance. Both precision and recall are above 0.9 and KNN has the best performance. Although Lying accounts for only 6.4%, Lying characteristics are quite different from the other classes, so even if the data is less, the model can still learn the characteristics of Lying. As for the Fall class, since there is less data, most models' performance drop. For example, XGBoost, AdaBoost and SVMs have recall lower than 0.7. On the other hand, KNN has the best performance in every class, surpassing the common machine learning competition algorithm, XGBoost. The possible reason is that KNN only calculates the nearest neighbours to make a prediction, so the computation is local, and the effect from the imbalanced data is less.

2) Machine Learning with Oversampling

Oversampling is an effective way to deal with imbalanced data. It can generate new data via random sampling or data synthesis. Since we have the best performance on KNN in the previous experiment, we use different oversampling methods to balance our data and evaluate the performance via KNN. First, we divide the data into training and testing data then oversampling on training data. Therefore, the proportion of training data becomes balanced data, while testing data retains the original imbalanced distribution.

After oversampling, we use the testing data to test whether the model has overfitting. As experiment result shown in Table 5, both Normal and Lying classes have good performance. Furthermore, the recall of the Fall class improves from 0.73 to 0.88, which means the model can detect 88% of real fall events. Finally, comparing different oversampling methods, SMOTE Tomek and SMOTE have the best recall, and random sampling has the most favourable precision. To consider both

TABLE 5. Different oversampling methods performance on KNN

Normal	Precision	Recall	Specificity	F1-Score	AUC
Random Sampling	1.00	1.00	0.97	1.00	0.99
SMOTE	1.00	0.99	0.99	1.00	1.00
SMOTE Tomek	1.00	0.99	0.98	0.99	0.99
ADASYN	1.00	0.99	0.98	0.99	0.99
Fall	Precision	Recall	Specificity	F1-Score	AUC
Random Sampling	0.82	0.80	0.99	0.81	0.91
SMOTE	0.81	0.88	1.00	0.85	0.95
SMOTE Tomek	0.72	0.86	0.99	0.78	0.94
ADASYN	0.70	0.85	0.99	0.77	0.94
Lying	Precision	Recall	Specificity	F1-Score	AUC
Random Sampling	0.94	0.98	1.00	0.96	0.99
SMOTE	0.95	0.97	1.00	0.96	0.99
SMOTE Tomek	0.96	0.97	1.00	0.96	0.99
ADASYN	0.95	0.97	1.00	0.96	0.98

TABLE 6. Different anomaly detection methods performance

Normal	Precision	Recall	Specificity	F1-Score	AUC
Isolation Forest	0.97	0.95	0.73	0.96	0.84
Elliptic Envelope	0.96	0.95	0.70	0.96	0.82
One Class SVM	0.92	0.97	0.24	0.94	0.61
Abnormal	Precision	Recall	Specificity	F1-Score	AUC
Isolation Forest	0.65	0.73	0.95	0.68	0.84
Elliptic Envelope	0.61	0.70	0.95	0.65	0.82
One Class SVM	0.48	0.24	0.97	0.32	0.61

precision and recall, we can refer to the F1 score. SMOTE has the highest F1 score, 0.85. In conclusion, four methods effectively enhance the model's learning of a minority class, and there is not much difference in the performance.

3) Anomaly Detection

Another method to deal with imbalanced data is to identify data via anomaly detection. Since anomaly detection is a binary classification, we merged the Fall and Lying data into the Abnormal class. Using anomaly detection, we consider the minority class as outliers. Thus, Normal data accounts for 91% of the data distribution, and Abnormal' data accounts for 9%.

As the experiment result shown in Table 6, the isolation forest has the best performance. However, most of the model's precisions are low. The possible reason is that the model predicts more normal data as abnormal, which causes the recall to rise, but the precision drop. Thus, evaluating the performance with F1-Score, the machine learning-based methods perform better than anomaly detection-based methods. The possible reason is that there is a lot of overlap in the dataset, so the anomaly detection method cannot separate Normal and Abnormal well.

E. PERFORMANCE ON IMAGE

After the previous experiments, we use the KNN with the synthetic minority oversampling Technique SMOTE, to test

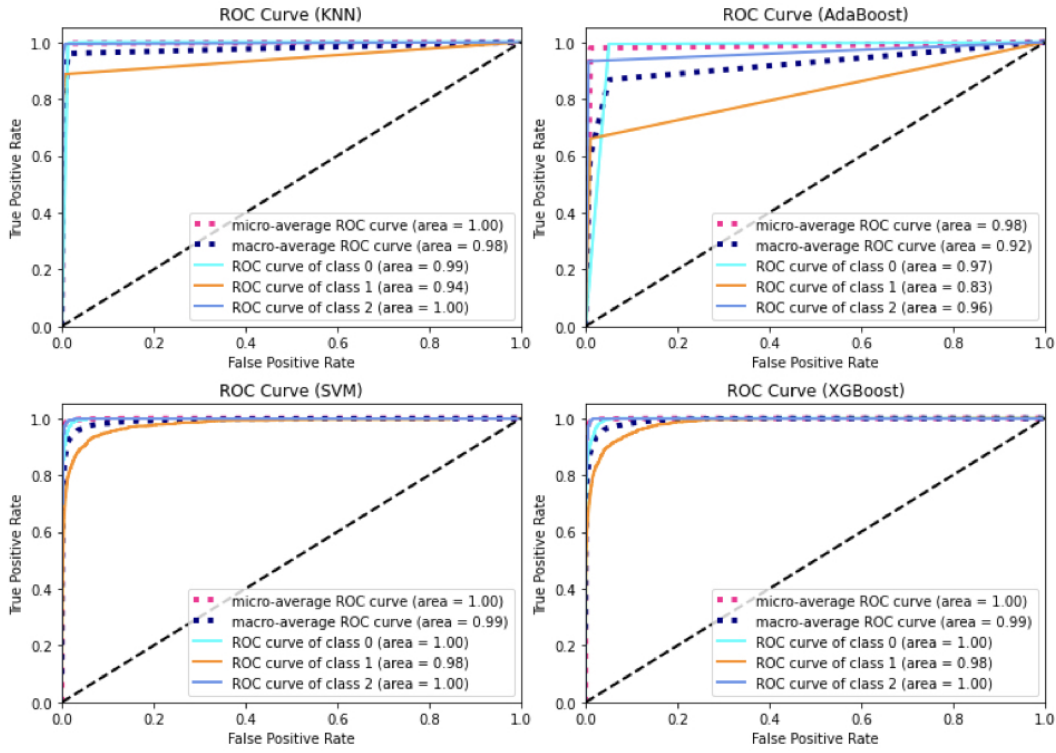


FIGURE 10. ROC Curve - Different machine learning approaches

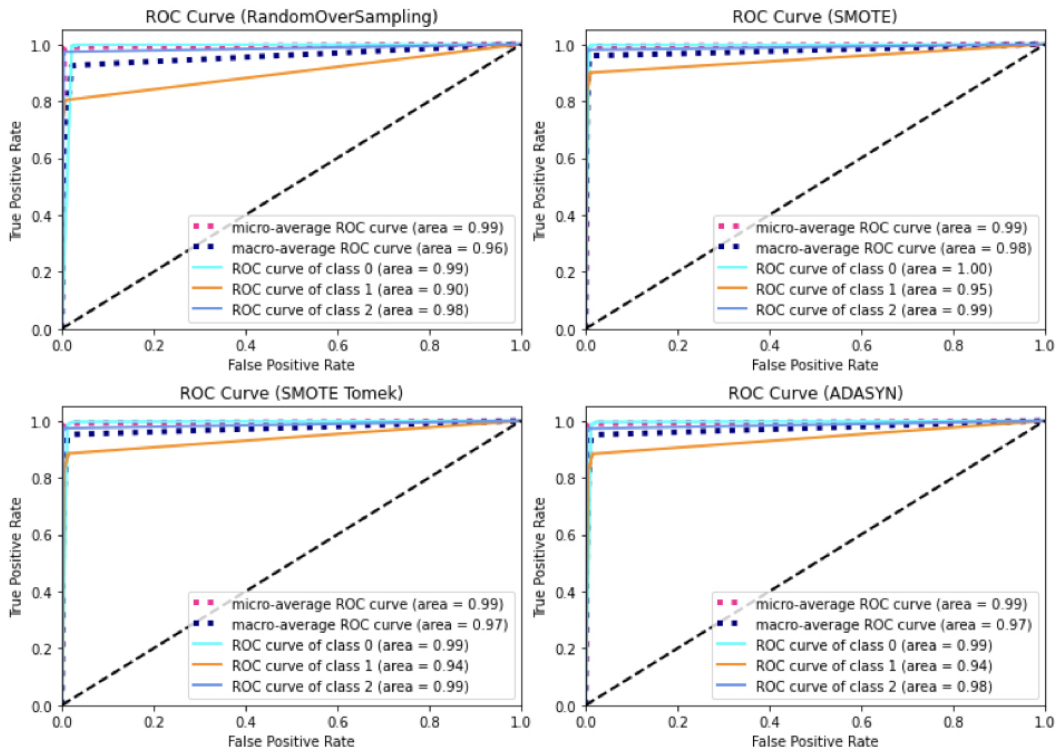


FIGURE 11. ROC Curve - Different oversampling approaches

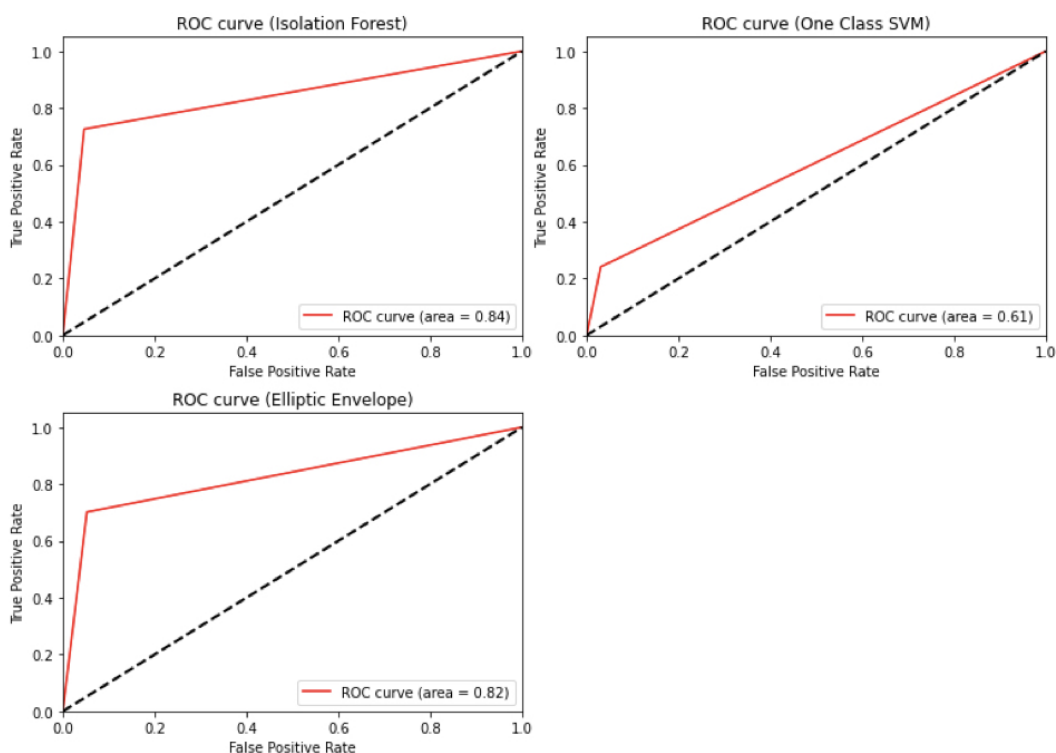


FIGURE 12. ROC Curve - Different anomaly detection approaches

the performance on the image. In Figure 13, results show that our approach can successfully identify the different scenarios. Furthermore, although OpenPose sometimes misses detecting the body keypoints, our approach can still predict based on the remaining body keypoints.

Moreover, our methods not only can work on single-person scenarios but also multi-people scenarios. In multi-people scenarios, we test our approach with IASLAB-RGBD Fallen Person Dataset [35]. The images are taken in the lab environment, and each image has more than one person. Although we never train those images with our model, our model can still identify the fall events in the image. Performance on multi-people scenarios is shown in Figure 14.

Since we extract the interpretable features from the skeleton information, we can show the key features on the images to have more information of when abnormal events happen. Figure 15 shows the interpretability of our approach.

VI. DISCUSSION

A. MACHINE LEARNING PERFORMANCE

From previous vision-based approaches, they mainly use threshold and machine learning approaches as classifiers. Machine learning is preferable since it can adjust the different shapes of a person. Using traditional machine learning methods as classifiers, it can receive good performance in previous research. However, before performing oversampling

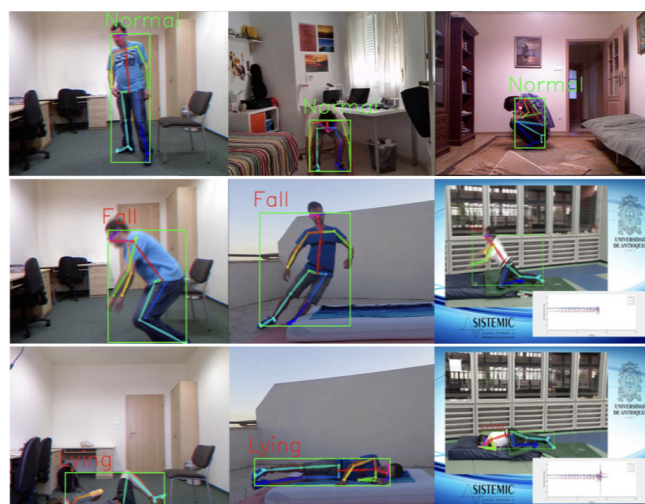


FIGURE 13. Performance on single-person scenarios [21][22][23]

methods, our machine learning approaches do not perform well in the experiment, except for KNN. The possible reason may be our dataset. We collect four different fall datasets and one gait dataset. The data proportion is 90.6% Normal, 2.6% Fall, and 6.8% Lying, which is highly imbalanced. Each dataset is recorded in different environments. Although

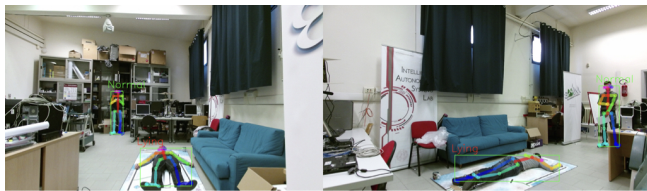


FIGURE 14. Performance on multi-people scenarios [35]



FIGURE 15. Abnormal event with key information [22]

this added more variation into the experiment, it is close to real-world situations since fall events have lots of variation. After performing oversampling methods, the performance improves. Performance in test data shows that we do not have an overfitting problem. Performance in the video can identify fall events correctly, even with the fallen image from another dataset. Note that, *Fall* class has the worst performance compared to other classes. The possible reason is that since fall is structured to move consecutively, each class has a large overlap with each other and we lack a clear definition for *Fall* class. This makes it difficult for models to identify the difference.

B. PROS AND CONS OF POSE ESTIMATION-BASED APPROACHES

A survey paper from Biswas et al. [45] summarized the main challenges in vision-based fall detection approaches. The challenges are poor identification of pose, the area at home or public area, number of people in the frame, poor lighting conditions, occlusion, subject's distance, and usage of aid accessories [44]. Since we use OpenPose as our pose estimation method, OpenPose's performance significantly influences our approaches. OpenPose can detect multiple people and even at different distances. If a person's lower body is occluded, our approach can predict based on upper body features. Thus, occlusion's problem is lighter than previous research. Using skeleton information is much interpretable and easier to understand, which is useful when cooperating with healthcare workers. The cons are the same as OpenPose. Some common failure cases are rare poses, overlapping with other people, and acquiring a false-positive on a statue or reflection. That false skeleton information causes a false alarm in our approach. The solution can be utilizing object

detection techniques to check whether skeleton data is included in the bounding box.

C. COMPARISON WITH PREVIOUS RESEARCHES

In Table 7, we compare our approach with the vision-based research after 2015. Since most studies use accuracy as evaluation metrics, we have to deal with our imbalanced dataset. Otherwise, the performance will be biased. Firstly, we randomly select 1000 data from each class from test data and form balanced data. Secondly, we evaluate the model performance with the balanced data. Thirdly, we iterate both steps 10 times and calculate the average accuracy. Our approach's average accuracy is 94.2%, which is comparable to previous research. However, it is difficult to compare different studies because each experiment setting and dataset are different [31]. Lack of standard evaluation setup causes the comparison unfair. Moreover, most of the data is simulated data, so the performance in the real world is questioned.

D. RECOMMENDATION FOR FURTHER RESEARCH

Regarding further research, pose estimation-based approaches are a new trend, which has potential in future applications. Instead of using keypoint coordinates as features, we recommend extracting the features from skeleton information. This is more robust on different frame sizes. Since pose estimation is used a lot in motion recognition, training models with more activities can improve the robustness of the model. In our experiment, most of the data in the Normal class is gait data. Normal data also include a few activities such as sitting, squatting, etc. Adding more kinds of activities can allow the application to deal with different situations. Adding more abnormal events for detection can also be another direction since cameras can be used in multi-people scenes, the more functional, the better.

The health science field is definitely a hot research field for further application. Due to the shortage of caregivers, ageing society is a problem that we cannot ignore. Health care costs are increasing in developed countries, such as the USA and Canada. Automation is the solution to this financial burden. Combining with health science and technique, more domain knowledge is required. Fall prevention is a more important challenge than fall detection. Most research concludes that falls are a mix-factor event. There are intrinsic and extrinsic factors that can strongly influence people's safety. Although the most decisive factor has not been realized, gait assessment is considered the most significant fall risk assessment. With the help of pose estimation-based approaches, automated fall risk assessment can be deployed everywhere with cameras. The tilt angle of the body, and the moving distance of the leg, can all be analyzed. Pose estimation can fill the research gap between fall detection and fall prevention.

The Robotics field is also a potential direction. The robotic technique is developing in industry and health science. Robotics and automated manufacturing are used in factories to enhance efficiency and save budget. Robotics can assist in surgery and diagnoses in health care. The robot mainly acts

TABLE 7. Performance comparison with the research after 2015

Study	Sensors	Features	Accuracy
Nizam et al.[63]	Depth image	Body Joints	93%
Yun et al.[64]	RGB image	Acceleration and motion	93.3%
Kwolek et al.[65]	Depth image, accelerometer	Posture and motion	98.6%
Fouzi et al.[66]	RGB image	Fall index and area ratios	96.8%
Our approach	RGB image	Body Shape, acceleration, deflection	94.2%

as an assistant nowadays. However, in the future, we expect robots to do more. We research human-robot interaction to add humanity to machines. Those humanity features can build trust and support systems with humans. For example, we want a health assistive robot to take care of our family and support us physically and mentally. Nowadays, we already have those robots in the market with human features [48][49][50][61] and families and the elderly have a more positive and acceptable attitude towards robots [47]. Figure 13 shows the current assistive robots in the market. Pose estimation approach can help robots understand human posture, preventing accidents. From assistant to protector, robotics still has lots of potentials to explore.



FIGURE 16. Current assistive robots in the market [48][49][50][61]

VII. CONCLUSION

In this paper, we address the problem of fall risk detection. We discuss the factors for fall risk and the associated costs in health care. Fall risk is dangerous to all individuals, especially for the elderly. The consequence of falls may cause physical injury and mental trauma. We need to take proper action and steps to prevent it. Due to the shortage of health workers and the increasing financial burden on the health care system. We propose a new pose estimation-based fall detection algorithm via RGB camera. We use OpenPose as a feature extractor to extract the skeleton data and then transfer them into 14 new features. New features are more interpretable compared to skeleton data. As for the dataset, we compose our dataset with four fall datasets and one gait dataset. The dataset is highly imbalanced, which meets real-world situations. In the experiment, we evaluate the performance of sampling methods and anomaly detection on imbalanced data. KNN plus oversampling has the best performance. The F1 scores of the three different classes, *Normal*, *Fall*, *Lying*, are 1.00, 0.85 and 0.96, respectively. This result is comparable to previous research. Although OpenPose misses some body keypoints sometimes, our approach can base on the remaining feature to make a decision. And most importantly, compared to previous research on

fall detection, our fall detection approach can handle multi-people scenarios. Next step, more data is needed to increase the diversity of fall and ADL events. Surveying more domain knowledge in health science can help us decide on more crucial features.

REFERENCES

- [1] P. Canada, "Seniors' Falls in Canada: Second Report," Government of Canada, April 10, 2014. [Online]. Available: <https://www.canada.ca/en/public-health/services/health-promotion/aging-seniors/publications/publications-general-public/seniors-falls-canada-second-report.html>
- [2] S. E. Lamb, E. C. Jørstad-Stein, K. Hauer, and C. Becker, "Development of a common outcome data set for fall injury prevention trials: the Prevention of Falls Network Europe consensus," *Journal of the American Geriatrics Society*, vol. 53, no. 9, 2005, pp. 1618-1622.
- [3] A. F. Ambrose, G. Paul, and J. M. Hausdorff, "Risk factors for falls among older adults: a review of the literature," *Maturitas*, vol. 75, no. 1, 2013, pp. 51-61.
- [4] S. Deandrea, E. Lucenteforte, F. Bravi, R. Foschi, C. La Vecchia, and E. Negri, "Risk factors for falls in community-dwelling older people: a systematic review and meta-analysis," *Epidemiology*, 2010, pp. 658-668.
- [5] S. L. Greenspan, E. R. Myers, D. P. Kiel, R. A. Parker, W. C. Hayes, and N. M. Resnick, "Fall direction, bone mineral density, and function: risk factors for hip fracture in frail nursing home elderly," *The American journal of medicine*, vol. 104, no. 6, 1998, pp. 539-545.
- [6] C. Smeesters, W. C. Hayes, and T. A. McMahon, "Disturbance type and gait speed affect fall direction and impact location," *Journal of biomechanics*, vol. 34, no. 3, 2001, pp. 309-317.
- [7] W. S. Leung, H. T. Chi, M. H. Hu, and M. T. Liu, "Fall mechanism and injury severity in community dwelling older people," *Formosa J Phy Ther*, vol. 30, 2005, pp. 105-15.
- [8] J. T. Perry, S. Kellog, S. M. Vaidya, J. H. Youn, H. Ali, and H. Sharif, "Survey and evaluation of real-time fall detection approaches," In 2009 6th International Symposium on High Capacity Optical Networks and Enabling Technologies (HONET), December 2009, pp. 158-164.
- [9] M. Kangas, A. Konttila, P. Lindgren, I. Winblad, and T. Jämsä, "Comparison of low-complexity fall detection algorithms for body attached accelerometers," *Gait posture*, vol. 28, no. 2, 2008, pp. 285-291.
- [10] S. S. Khan and J. Hoey, "Review of fall detection techniques: A data availability perspective," *Medical engineering physics*, vol. 39, 2017, pp. 12-22.
- [11] W. Wu, S. Dasgupta, E. E. Ramirez, C. Peterson, and G. J. Norman, "Classification accuracies of physical activities using smartphone motion sensors," *Journal of medical Internet research*, vol. 14, no.5, e130, 2012.
- [12] F. Werner, J. Diermaier, S. Schmid, and P. Panek, "Fall detection with distributed floor-mounted accelerometers: An overview of the development and evaluation of a fall detection system within the project eHome," In 2011 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) and Workshops, May 2011, pp. 354-361.
- [13] M. Daher, A. Diab, M. E. B. El Najjar, M. A. Khalil, and F. Charpillet, "Elder tracking and fall detection system using smart tiles," *IEEE Sensors Journal*, vol. 17, no. 2, 2016, pp. 469-479.
- [14] M. Volkhardt, F. Schneemann, and H. M. Gross, "Fallen person detection for mobile robots using 3d depth data," In 2013 IEEE International Conference on Systems, Man, and Cybernetics, October 2013, pp. 3573-3578.
- [15] A. Yajai, A. Rodtook, K. Chinnasarn, and S. Rasmeequan, "Fall detection using directional bounding box," In 2015 12th International Joint Conference on Computer Science and Software Engineering (JCSSE), July 2015, pp. 52-57.

- [16] K. L. Lu and E. T. H. Chu, "An image-based fall detection system for the elderly," *Applied Sciences*, vol. 8, no. 10, 1995, 2018.
- [17] W. Chen, Z. Jiang, H. Guo, and X. Ni, "Fall detection based on key points of human-skeleton using openpose," *Symmetry*, vol. 12, no. 5, 744, 2020.
- [18] G. Sun and Z. Wang, "Fall detection algorithm for the elderly based on human posture estimation," In *2020 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC)*, April 2020, pp. 172-176.
- [19] Z. Huang, Y. Liu, Y. Fang, and B. K. Horn, "Video-based fall detection for seniors with human pose estimation," In *2018 4th International Conference on Universal Village (UV)*, October 2018, pp. 1-4.
- [20] S. Jeong, S. Kang, and I. Chun, "Human-skeleton based Fall-Detection Method using LSTM for Manufacturing Industries," In *2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, June 2019, pp. 1-4.
- [21] B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Computer methods and programs in biomedicine*, vol. 117, no. 3, 2014, pp. 489-501.
- [22] E. Casilari, J. A. Santoyo-Ramón, and J. M. Cano-García, "Umafal: A multisensor dataset for the research on automatic fall detection," *Procedia Computer Science*, vol. 110, 2017, pp. 32-39.
- [23] A. Sucerquia, J. D. López, and J. F. Vargas-Bonilla, "SisFall: A fall and movement dataset," *Sensors*, vol. 17, no. 1, 198, 2017.
- [24] E. Auvinet, C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Multiple cameras fall dataset," *DIRO-Université de Montréal, Tech. Rep.*, 2010, 1350.
- [25] D. López-Fernández, F. J. Madrid-Cuevas, and Á. Carmona-Poyato, M. J. Marín-Jiménez, and R. Muñoz-Salinas, "The AVA multi-view dataset for gait recognition," In *International workshop on activity monitoring by multiple distributed sensing*, Springer, Cham, August 2014, pp. 26-39.
- [26] Z. Cao, T. Simon, S. E. Wei, and Y. Sheikh, "Realtime multi-person 2D pose estimation using part affinity fields," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [27] A.X. Araya and E. Iriarte, "Fear of Falling among Community-dwelling Sedentary and Active Older People," *Invest Educ Enferm*, vol. 39, no. 1, 2021.
- [28] M. Kangas, A. Konttila, P. Lindgren, I. Winblad, and T. Jämsä, "Comparison of low-complexity fall detection algorithms for body attached accelerometer," *Gait posture*, vol. 28, no. 2, 2008, pp. 285-291.
- [29] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, August 2016, pp. 785-794.
- [30] K. Han, Q. Yang, and Z. Huang, "A two-stage fall recognition algorithm based on human posture features," *Sensors*, vol. 20, no. 23, 6966, 2020.
- [31] P. Vallabh and R. Malekian, "Fall detection monitoring systems: a comprehensive review," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 6, 1809-1833, 2018.
- [32] T. Huang, "Ensemble learning Bagging Boosting AdaBoost," *Medium*, June 20, 2018. [Online]. Available: <https://chih-sheng-huang821.medium.com/>
- [33] K. Rocha, "People walking on pedestrian lane during daytime," (n.d.). [Online]. Available: <https://www.pexels.com/photo/people-walking-on-pedestrian-lane-during-daytime-109919/>
- [34] "Falls," *World Health Organization*, April 26, 2021. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/falls>
- [35] M. Antonello, M. Carraro, M. Pierobon, and E. Menegatti, "Fast and robust detection of fallen people from a mobile robot," In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, September 2017, pp. 4159-4166.
- [36] S. G. Miaou, P. H. Sung, and C. Y. Huang, "A customized human fall detection system using omni-camera images and personal information," In *1st Transdisciplinary Conference on Distributed Diagnosis and Home Healthcare*, April 2006, pp. 39-42.
- [37] R. Luque, E. Casilari, M. J. Morón, and G. Redondo, "Comparison and characterization of android-based fall detection systems," *Sensors*, vol. 14, no. 10, 2014, pp. 18543-18574.
- [38] B. Andò, S. Baglio, C. O. Lombardo, and V. Marletta, "A multisensor data-fusion approach for ADL and fall classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 9, 2016, pp. 1960-1967.
- [39] M. Á. Á. De la Concepción, L.M.S. Morillo, J. A. Á. García, and L. González-Abril, "Mobile activity recognition and fall detection system for elderly people using Aमेva algorithm," *Pervasive and Mobile Computing*, vol. 34, 2017, pp. 3-13.
- [40] "Fall Detection System Market," *Persistence Market Research*, September 2019. [Online]. Available: <https://www.persistencemarketresearch.com/market-research/fall-detection-system-market.asp>
- [41] R. Bhardwaj, A. R. Nambiar, and D. Dutta, "A study of machine learning in healthcare," In *2017 IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, vol. 2, July 2017, pp. 236-241.
- [42] J. Silva, J. Madureira, C. Tonelo, D. Baltazar, C. Silva, A. Martins, and I. Sousa, "Comparing machine learning approaches for fall risk assessment," In *International Conference on Bio-inspired Systems and Signal Processing*, vol. 5, February 2017, pp. 223-230.
- [43] "Our Aging Population: Statistics," *Comfort Life*, (n.d.). [Online]. Available: <https://www.comfortlife.ca/retirement-communities/our-aging-population-statistics>
- [44] Z. Cao, G. Hidalgo, T. Simon, S. E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 1, 2019, pp. 172-186.
- [45] A. Biswas and B. Dey, "A Literature Review of Current Vision Based Fall Detection Methods," In *International Conference on Communication, Devices and Networking*. Springer, Singapore, December 2019, pp. 411-421.
- [46] Y. R. Serpa, M. B. Nogueira, P. P. M. Neto, and M. A. F. Rodrigues, "Evaluating Pose Estimation as a Solution to the Fall Detection Problem," In *2020 IEEE 8th International Conference on Serious Games and Applications for Health (SeGAH)*, August 2020, pp. 1-7.
- [47] T. Huang and H. Liu, "Acceptability of robots to assist the elderly by future designers: A case of Guangdong Ocean University industrial design students," *Sustainability*, vol. 11, no. 15, 4139, 2019.
- [48] "ElliQ robot by Yves Behar helps ageing adults stay connected to the world," *YouTube*, January 11, 2017. [Online]. Available: <https://www.youtube.com/watch?v=jsffC2gWCow>
- [49] "The Mabu Personal Healthcare Companion from Catalia Health," *YouTube*, December 21, 2017. [Online]. Available: <https://www.youtube.com/watch?v=A3XwzlvOW7k>
- [50] "Meet a robot offering care and companionship to seniors," *YouTube*, June 7, 2018. [Online]. Available: https://www.youtube.com/watch?v=Q98NIE_SRot=87s
- [51] "The Shortage of Personal Support Workers in Canada – Is There A Solution," *Living Assistance Services*, February 1, 2019. [Online]. Available: <https://www.laservices.ca/the-shortage-of-personal-support-workers-in-canada-is-there-a-solution/>
- [52] M. Bosch-Jorge, A. J. Sánchez-Salmerón, Á. Valera, and C. Ricolfe-Viala, "Fall detection based on the gravity vector using a wide-angle camera," *Expert systems with applications*, vol. 41, no. 17, 2014, pp. 7980-7986.
- [53] L. Hazelhoff and J. Han, "Video-based fall detection in the home using principal component analysis," In *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, Berlin, Heidelberg, October 2008, pp. 298-309.
- [54] A. Bochkovskiy, C. Y. Wang, and H. Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv preprint arXiv:2004.10934*, 2020.
- [55] A. Núñez-Marcos, G. Azkune, and I. Arganda-Carreras, "Vision-based fall detection with convolutional neural networks," *Wireless communications and mobile computing*, 2017.
- [56] K. De Miguel, A. Brunete, M. Hernando, and E. Gamba, "Home camera-based fall detection system for the elderly," *Sensors*, vol. 17, no. 12, 2864, 2017.
- [57] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Monocular 3D head tracking to detect falls of elderly people," In *2006 international conference of the IEEE engineering in medicine and biology society*, August 2006, pp. 6384-6387.
- [58] M. Karbasi, S. Bilal, R. Aghababaeian, A.E. Rad, Z. Bhatti, and A. Shah, "Analysis and Enhancement of the Denoising Depth Data Using Kinect Through Iterative Technique," *Jurnal Teknologi*, vol. 78, no. 9, 2016.
- [59] S. W. Yang and S. K. Lin, "Fall detection for multiple pedestrians using depth image processing technique," *Computer methods and programs in biomedicine*, vol. 114, no. 2, 2014, pp. 172-182.
- [60] K. P. Kamble, S. S. Sontakke, H. Donadkar, R. Poshattiwar, and A. Ananth, "Fall Alert: A Novel Approach to Detect Fall Using Base as a YOLO Object Detection," In *International Conference on Advanced Machine Learning Technologies and Applications*. Springer, Singapore, February 2020, pp. 15-24.
- [61] "Zenbo -Your Smart Little Companion," *YouTube*, August 19, 2016. [Online]. Available: <https://www.youtube.com/watch?v=lzVHDgItx6o>
- [62] Z. Cao, G. Hidalgo, T. Simon, S.E. Wei, and Y. Sheikh, "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields," *IEEE*

transactions on pattern analysis and machine intelligence, vol. 43, no. 1, 2019, pp. 172-186.

- [63] Y. Nizam, M.N.H. Mohd, and M.M.A. Jamil, "Human fall detection from depth images using position and velocity of subject," *Procedia Computer Science*, 2017, 105: 131-137.
- [64] Y. Yun, and I.Y.H. Gu, "Human fall detection in videos by fusing statistical features of shape and motion dynamics on riemannian manifolds," *Neuro-computing*, 2016, 207: 726-734.
- [65] B. Kwolek, and M. Kepski, "Fuzzy inference-based fall detection using kinect and body-worn accelerometer," *Applied Soft Computing*, 2016, 40: 305-318.
- [66] F. Harrou, N. Zerrouki, Y. Sun, and A. Houacine, "An intergrated vision-based approach for efficient human fall detection in a home environment," *IEEE Access*, 2019, 7: 114966-114974.



YEN-HUNG LIU received the bachelor of science degree in Electronic Engineering from the National Taipei Tech University, Taiwan. He is currently studying for the master of science degree in computer science from Ontario Tech University, Canada.

From 2019 to 2021, he was a Teaching Assistant for cloud service, object-oriented programming, the special topic of service robotic. Since 2020, he is a Research Assistant with Zayed University. His

research fields are the application of machine learning and deep learning. He also has experience in data mining, computer vision, cloud service, and service robotic. The work experience includes two internships. The first is related to customer analysis. The second is related to system.



PATRICK C. K. HUNG is a Professor, Graduate Program Director in Computer Science and Director of International Programs at the Faculty of Business and Information Technology at Ontario Tech University, Canada. Patrick worked with Boeing Research and Technology at Seattle on aviation services-related research with two U.S. patents on mobile network dynamic workflow systems. Before that, he was a Research Scientist with the Commonwealth Scientific and Industrial

Research Organization in Australia. He is a founding member of the IEEE Technical Committee on Services Computing and IEEE Transactions on Services Computing. He is an editorial board member for the IEEE Transactions on Engineering Management, an associate editor for Electronic Commerce Research and Applications. In addition, he is coordinating editor of the Information Systems Frontiers. He has a Ph.D. and Master in Computer Science from Hong Kong University of Science and Technology, a Master in Management Sciences from the University of Waterloo, Canada, and a Bachelor in Computer Science from the University of New South Wales, Australia.



FARKHUND IQBAL is the team lead for Cybersecurity and Digital Forensics (CAD) research group and is the Director for Advanced Cyber Forensics Research Laboratory (ACFRL) in the college. He has secured 3M AED (as internal and external funding) to upgrade ACFRL and to pursue research projects as PI and Co-PI. He holds a Masters (2005) and a Ph.D. degree (2011) in Computer Science from Concordia University, Canada. His research expertise focuses on using

Artificial Intelligence technologies including deep learning and data analytics techniques for problem solving in healthcare, cybersecurity and cybercrime investigation in smart and safe city domains. He has published more than 90 papers in high rank journals and conferences. His latest research culminated in a highly anticipated book titled "Machine Learning for Authorship Attribution and Cyber Forensics", (as lead author) Springer NATURE, 2020. He is an Affiliate Professor at Ontario Tech University, Canada. He introduced multiple new courses (at graduate and undergrad level) in cybersecurity and digital forensics. He has supervised multiple postgraduate students including Master's and PhD.



BENJAMIN C. M. FUNG is a Canada Research Chair in Data Mining for Cybersecurity, a Full Professor with the School of Information Studies, and an Associate Member with the School of Computer Science at McGill University in Canada. He received a Ph.D. degree in computing science from Simon Fraser University in Canada in 2007. He has over 130 refereed publications that span the research forums of data mining, privacy protection, cybersecurity, services computing, and

building engineering. His data mining works in crime investigation and authorship analysis have been reported by media worldwide. Prof. Fung is a licensed Professional Engineer of software engineering in Ontario, Canada.

...