IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 2 February 2024, accepted 13 February 2024, date of publication 20 February 2024, date of current version 28 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3368065

RESEARCH ARTICLE

Computer Vision Based Transfer Learning-Aided Transformer Model for Fall Detection and Prediction

SHELDON MCCALL[1], SHINA SAMUEL KOLAWOLE[1], AFREEN NAZ[1], LIYUN GONG[1], SYED WAQAR AHMED[2], PANDEY SHOURYA PRASAD[3], MIAO YU[1], JAMES WINGATE[1], AND SAEID POURROOSTAEI ARDAKANI[1]

[1]School of Computer Science, University of Lincoln, LN6 7TS Lincoln, U.K.
[2]Department of Sciences and Humanities, FAST-National University of Computer and Emerging Sciences, Islamabad 44000, Pakistan
[3]International Institute of Information Technology Bangalore, Bengaluru 560100, India

Corresponding author: Miao Yu (myu@lincoln.ac.uk)

This research has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 778602 ULTRACEPT.

ABSTRACT Falls bring about significant risks to individuals' well-being and independence, prompting widespread public health concerns. Swift detection and even predicting the risk of falls are crucial for implementing effective measures to alleviate the adverse consequences associated with such incidents. This study presents a new framework for identifying and forecasting fall risks. Our approach utilizes a novel transformer model trained on 2D poses extracted through an off-the-shelf pose extractor, incorporating transfer learning techniques. Initially, the transformer is trained on a large dataset containing 2D poses of general actions. Subsequently, we freeze the majority of its layers and fine-tune only the last few layers using relatively smaller datasets for fall detection and prediction tasks. Experimental results indicate that our proposed method outperforms traditional machine learning (e.g., SVM, Decision Tree, etc.) and deep learning approaches (e.g., LSTM, CNN, ST-GCN, PoseC3D, etc.) in both fall detection and prediction tasks across various datasets.

INDEX TERMS Computer vision, deep learning, fall detection, fall prediction, healthcare, transfer learning, transformer.

I. INTRODUCTION

Fall among older adults represents a critical public health concern, posing a significant threat to their well-being and independence. According to [1], approximately one-third of individuals aged 65 and above, and half of those aged 80 and older, experience at least one fall annually. The consequences of falling can result in severe physiological and psychological damage to an elderly person's overall health. Notably, falling ranks among the top three most common causes of Traumatic Brain Injury (TBI) in the United States, as reported by [2]. Tragically, around 10% of all falls in seniors lead to major injuries, including intracranial injuries (ICIs) and fractures, as documented in [3]. The gravity of the issue becomes even more apparent when considering the World Health Organisation's (WHO) report [4], which identifies falling as the second leading cause of accidental or unintentional injury-related deaths worldwide. Even when falls don't result in serious injuries, fallers often struggle to get up without assistance, leading to prolonged periods of lying on the floor, known as "long lies." These "long lies" can lead to dehydration, pressure sores, pneumonia, hypothermia, and even death [5]. Fall detection and prediction play a crucial role in mitigating the negative effects of falls mentioned earlier. Fall detection aims to promptly identify falls as they occur, ensuring timely assistance can be provided to prevent prolonged lying on the floor. On the other hand, fall prediction assesses the likelihood that an individual will experience a fall, allowing for proactive interventions to be implemented, thus preventing potential falls and associated injuries.

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

In this study, our emphasis is on addressing the research challenge of employing cutting-edge sensor technologies and artificial intelligence techniques for the detection and prediction of falls. This particular research problem has garnered significant attention in recent years. Up until 2024, researchers have diligently explored and validated a multitude of approaches, utilizing diverse sensor modalities including cameras and wearable sensors. These investigations have seamlessly integrated advanced signal processing and machine learning techniques into the realm of fall detection and prediction [6], [7]. Among different techniques, the computer vision-based approach [8] utilizing cameras has gained significant attention in the fall detection/prediction community. The key advantage of this method lies in its non-intrusive nature, as it does not require users to wear any additional equipment. This makes it a convenient and user-friendly option for fall detection and prediction.

This study presents a unified framework that effectively addresses fall detection and prediction tasks using state-of-the-art computer vision and machine learning techniques. The proposed approach comprises several key steps. Initially, we extract body key points from video frames by employing an "on-the-shelf" key points detector. These detected key points are then pre-processed and input into a novel transformer model, which serves as the foundation for fall detection and prediction. Due to the limited availability of fall data, which may not be sufficient to train a complex transformer model, we incorporate transfer learning [9] into our methodology. We train the transformer model on the MPOSE dataset [10], a comprehensive collection of 2D pose sequences involving various actions like walking, jogging, running, and kicking. This pre-training step helps the model learn useful representations from abundant data related to human poses and actions. Following the pre-training phase, we fine-tune the transformer model for fall detection and prediction. Most of the network parameters are frozen, retaining the knowledge gained during pre-training, while only the last few layers are fine-tuned to adapt the model specifically to fall-related tasks. By combining transfer learning with advanced computer vision and machine learning techniques, our proposed framework demonstrates enhanced performance in both fall detection and prediction tasks, even when working with limited fall-specific data.

In contrast to traditional computer vision-based techniques for fall detection and prediction, our proposed approach offers distinct contributions. Firstly, we present pioneering work that leverages a novel transformer model. Secondly, we employ transfer learning to efficiently train our model with limited data, enhancing the accuracy of fall detection and prediction. Furthermore, our technique transcends the confines of a singular task, catering to both effective fall detection and prediction. Extensive evaluation studies demonstrate the superiority of our proposed technique over other counterparts in the realm of dual fall detection and prediction tasks.

II. LITERATURE REVIEW

Lately, numerous research endeavors have focused on the realm of computer vision for fall detection and prediction. The following is a summary of these studies.

A. FALL DETECTION

In the domain of fall detection, researchers have explored threshold-based methodologies in several studies ([11], [12], [13], [14]). These approaches involve the extraction of specific features such as head velocity ([11], [12]), the height-to-width ratio ([13]), and movement amplitude along with shape changes ([14]) through image processing techniques. These features are subsequently compared to predefined threshold values to identify falls. However, it's worth noting that such threshold-based techniques may struggle to accurately distinguish between falls occurring in various directions within the images due to their reliance on a single threshold value which is not robust enough to distinguish falls in different directions.

Apart from the threshold-based approach, more sophisticated machine-learning techniques have gained significant traction in the field of fall detection. In the study [15], researchers employed a combination of OpenPose for human key point identification and DeepSORT for tracking. These key points were then input into various classifiers including Gradient-Boosted Trees (GDBT), Decision Trees (DT), Random Forest (RF), Support Vector Machine (SVM), and k-Nearest Neighbor (KNN) to detect falls. In [16], a Hidden Markov Model (HMM) was utilized, leveraging silhouette data extracted through background subtraction techniques for fall detection. Remarkably, this approach achieved an accuracy of 84.72% based on their recorded dataset. Reference [17] employed a range of machine learning algorithms, including Naïve Bayes, k-Nearest Neighbor, Neural Networks, and Support Vector Machines. These algorithms were applied to video sequences using features extracted from human silhouette regions obtained through background subtraction algorithms. An evaluation was performed on the FDD and URFD datasets, with Support Vector Machines yielding the most robust performance. In [18], Support Vector Machines were again utilized, this time based on motion history images and histograms of oriented gradient features. This approach achieved remarkably high recall rates and precision rates of 98.1% and 96.8%, respectively, in a dataset consisting of realistic image sequences of both simulated falls and daily activities. Reference [19] introduced a Directed Acyclic Graph Support Vector Machine (DAGSVM) approach for posture classification and fall detection. Results from self-recorded datasets demonstrated superior accuracy compared to classical machine learning model counterparts.

Conventional machine learning-based approaches for fall detection heavily rely on manually crafted features. To advance the field and automatically capture crucial aspects of fall detection, deep learning methods have been introduced. In [20], Convolutional Neural Networks (CNNs) were

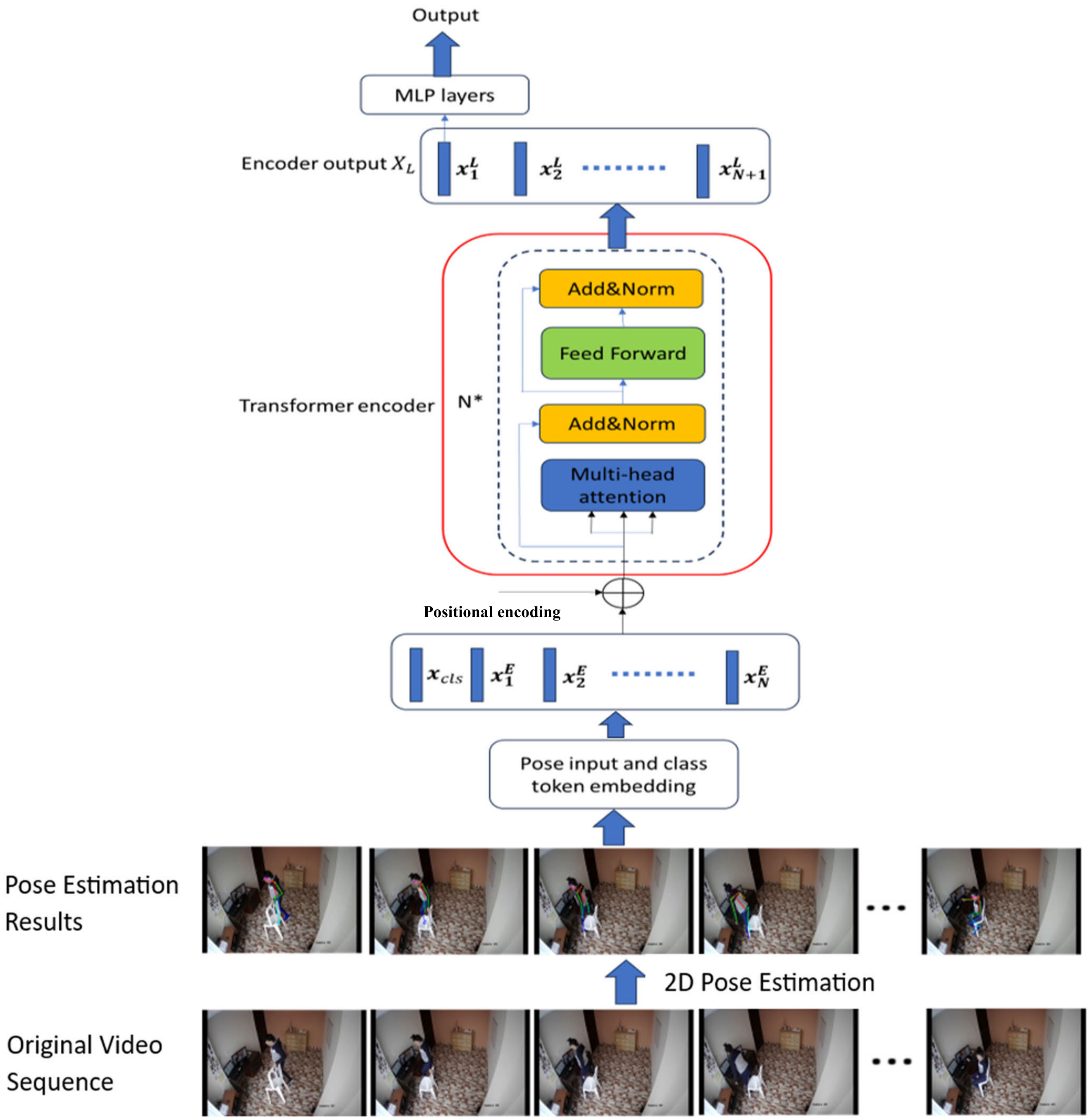


FIGURE 1. The flowchart of the proposed technique.

employed. They processed silhouettes extracted via background subtraction to classify various postures, including falling, standing, sitting, and bending. The results demonstrated that the CNN outperformed SVM models in falling classification [21] introduced a Three-Dimensional Convolutional Neural Network (3)-D CNN) to extract motion features from temporal sequences. To enhance spatial understanding, a Long Short-Term Memory (LSTM)-based spatial visual attention mechanism was incorporated. This combined approach effectively captured both temporal and spatial information for precise fall detection. In [22], the YOLO (You Only Look Once) network was harnessed to extract features using the darknet backbone from video frames. These features were then processed by the YOLO network's heads to detect fall regions in images. Impressively, this YOLO-based system achieved over 90% accuracy on the UR Fall dataset while being deployed on an edge device. Reference [23] utilized an ST-GCN (Spatiotemporal Graph Convolutional

Network) deep neural network model to classify fall and non-fall activities. This model operated on 3D skeleton sequences representing human actions. Experimental results revealed that the proposed ST-GCN model surpassed other counterparts such as Random Forest, SVM, and CNN in terms of fall detection accuracy.

B. FALL PREDICTION

In the realm of fall risk prediction using computer vision techniques, there has been comparatively less research, but notable strides have been made. In [24], a computer vision-based method was employed to predict the Timed-Up-and-Go (TUG) score, an indicator of fall risk. This was achieved by utilizing regression models, including linear regression and SVM regressors, based on 3D poses derived from video recordings captured by 2D/3D cameras. Experimental results demonstrated that the SVM regressor yielded more accurate TUG predictions. Reference [25] highlighted the significance of gait variables, such as cadence and step width/time, extracted from 2D human poses in video frames. These variables were found to be strongly associated with future falls through Poisson regression analysis, indicating their potential for fall risk prediction. In [7], gait parameters, alongside clinical assessment scores from STRATIFY, were combined to predict short-term fall risk for individuals with dementia in a domestic setting. A two-layered MLP network model was employed, achieving sensitivity and specificity rates of 72.8% and 73.2%, respectively. Reference [26] applied Computer Vision and Machine Learning techniques to differentiate normal gait patterns from those associated with fall risks. Four classification methods, including Convolutional Neural Networks (CNN), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Long Short-Term Memory (LSTM) neural networks, were used. Results indicated that SVM and KNN outperformed CNN and LSTM, delivering superior performance on the collected data. Reference [27] utilized a 3D vision sensor to capture 3D skeletons, from which features were extracted to train Random Forest and Support Vector Machine models for estimating the Berg Balance Scale (BBS) and assessing fall risk. A pilot test demonstrated high rates of fall risk prediction and a notable correlation with physiotherapists' BBS scores on individual motion tasks.

It is essential to recognize that both fall detection and prediction tasks addressed in this study revolve around the video sequence classification problem. This involves categorizing video sequences into fall/non-fall and high/low fall risk, constituting a typical pattern recognition classification challenge. In this context, our work aligns with other image classification studies, exemplified by [28], as they share a common focus on classification problems. However, while image classification, as demonstrated in [28], centers on categorizing individual images, our research delves into the classification of video sequences, comprising a series of images. This disparity necessitates distinct models and methodologies compared to single-image classification,

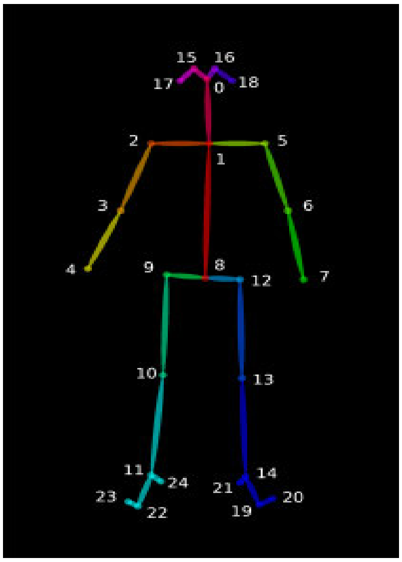


FIGURE 2. Illustration of 25 key points on the human body extracted by openpose.

aiming to fully leverage the temporal dependency information inherent in consecutive frames within a video sequence for precise fall detection and prediction.

In this study, we employ a transformer model complemented by transfer learning to address both fall detection and fall risk (high/low risk) prediction tasks, using 2D body key points extracted from the OpenPose detector [29]. In contrast to utilizing complete video frames and other video features, our approach leverages 2D skeletons, providing a streamlined representation of the human body. This reduction in data dimensionality enhances the computational efficiency of fall detection and prediction tasks. Additionally, the 2D skeleton representation mitigates sensitivity to variations in lighting, background clutter, and clothing. Distinguishing our research from prior work in the field, our study marks the pioneering use of a transformer model for fall detection and prediction tasks. Furthermore, we incorporate transfer learning by initially training the transformer model on the extensive MPOSE dataset. This approach allows the model to glean valuable insights from a wealth of motion data pertaining to human poses and actions, ultimately bolstering its performance in fall detection and prediction tasks.

III. METHODOLOGY

This study introduces an innovative approach to fall detection and fall risk prediction, leveraging a transformer model through the application of transfer learning. The related flowchart is presented in Figure 1, showing a visual representation of the methodology. A comprehensive explanation of the components depicted in Fig. 1 will be given in the next subsections.

A. POSE ESTIMATION AND PRE-PROCESSING

We utilize the Openpose algorithm [29] to extract 2D human poses from the original video sequences. Openpose allows us to extract the (x, y) coordinates of 25 key points from the subject's body for each individual frame in a scene, as visually illustrated in Fig. 2. Following the key point extraction by Openpose, we perform several preprocessing tasks. Initially, we establish the midpoint of the shoulders (point 1 in Fig. 2) as the origin and adjust the position of each key point accordingly to obtain relative key point positions. Subsequently, we standardize the key point positions using the length of the trunk, which is defined as the distance between points 1 and 8, as a normalization factor. Upon completing the preprocessing steps, we transform an individual key point denoted as $\boldsymbol{p}_i$ via the following formula:

$$\hat{\boldsymbol{p}}_i = \frac{\boldsymbol{p}_i - \boldsymbol{p}_1}{L} \tag{1}$$

where $\boldsymbol{p}_1$ represents the position of the shoulder and L represents the trunk length.

By employing the aforementioned pre-processing methods, we can ensure that the extracted 2D poses remain invariant with respect to scale and position. This enables the developed technique to effectively detect and predict fall risks across various positions and distances from the camera when processing videos. Based on the position of every pre-processed pose key point denoted as $\hat{\boldsymbol{p}}_i$, we further calculate its velocity between consecutive frames as $\boldsymbol{v}_i = \hat{\boldsymbol{p}}_i - \hat{\boldsymbol{p}}_{i-1}$ to glean more insights into the dynamics of the human body. For the i-th frame, we can obtain a concatenated vector denoted as $\boldsymbol{c}_i = [\hat{\boldsymbol{p}}_{i,1}; \ldots, \hat{\boldsymbol{p}}_{i,25}; \ldots, \boldsymbol{v}_{i,1}; \ldots \boldsymbol{v}_{i,25}]$ by combining both position and velocity information of all keypoints, where $\hat{\boldsymbol{p}}_{i,j}$ and $\boldsymbol{v}_{i,j}$ represent the position and velocity of the j-th keypoint at this i-th frame. Finally, we compile these concatenated vectors for all frames within a video sequence, denoted as $[\boldsymbol{c}_1, \ldots, \boldsymbol{c}_N]$ (where N is the sequence length) as the input for the subsequent transformer model.

B. TRANSFORMER FOR FALL DETECTION AND PREDICTION

The transformer architecture employed in this study draws inspiration from the methodology outlined in [30]. Unlike [30] for a single image classification, our developed model is adapted for video processing. It utilizes 2D skeletons extracted from a video sequence as its input. And a 'class token' [CLS] vector derived from the transformer encapsulating information from all skeletons, is taken as the feature for fall detection and prediction.

In specific, firstly embedding operation is performed by mapping every vector in the skeleton sequence to a sequence of higher dimension D tokens (denoted as $\boldsymbol{x}_1^E, \ldots, \boldsymbol{x}_N^E$) using a linear projection map $W \in \boldsymbol{R}^{P \times D}$, where P is the vector dimension. Besides, a trainable vector of dimension D (denoted as $\boldsymbol{x}_{cls}$) is prepended to the beginning of the embedded sequence as shown in Fig. 1, which is taken as a class token [CLS] leveraging the self-attention to aggregate information into a compact high-dimensional representation for fall detection/prediction. As the traditional transformer, the positional vectors are added to take into account the position information of vectors in a sequence for our fall detection/prediction tasks.

The embedded vectors and class token after being added by positional information, are then fed into a Transformer encoder. As shown in Fig. 1, the Transformer encoder contains multiple blocks while each block contains multi-head attention, additional&normalization and feed-forward layers. The pivotal element within the Transformer encoder is the multi-head attention layer [31], which leverages multiple 'heads' to generate outputs. For the i-th head, queries (Q_i), keys (K_i) and values (V) are computed as $Q_i = XW_{Q_i}$, $K_i = XW_{K_i}$ and $V_i = XW_{V_i}$ respectively, where X represents the input of the multi-head attention layer while W_{Q_i}, W_{K_i} and W_{V_i} denote the respective parameter matrices associated with the i-th head. Based on Q_i, K_i and V_i, the attention weights A_i for the i-th head are calculated as:

$$A_i = softmax(\frac{Q_i K_i^T}{\sqrt{D}^h}) \tag{2}$$

where D^h is a scale factor and $softmax(\cdot)$ is an activation function the as defined in [32] and the output of the i-th head denoted as H_i is calculated as:

$$H_i = A_i V_i \tag{3}$$

which is the weighted summation of V_i based on the attention weights. All head outputs are calculated in the same way and finally concatenated and linearly projected as the final output of the multi-head attention layer as:

$$MSA\ (X) = [H_1; H_2; \ldots; H_N] W_{MSA} \tag{4}$$

where $MSA\ (X)$ represents the multi-head attention layer output based on the input X, W_{MSA} is a projection matric and N is the head number in the multi-head attention layer. The output of a multi-head attention layer will then go through a series of add&norm operations and a small feed-forward network to generate the output of a block.

The final output of the whole transformer encoder is obtained through N blocks of multi-head attention operations as well as add&norm and feed-forward operations as shown in Fig. 1. The output of the transformer encoder–X_L is represented as:

$$X_L = F([x_{cls}; X_{embedded}] + X_{pos}) \tag{5}$$

where F(·) represents the operations associated with the transformer encoder. x_{cls} represents the class token, $X_{embedded}$ represents embedded vectors and X_{pos} represents positional vectors. Finally, only the first column of X_L corresponding to the class token is fed into a feed-forward network head, for performing the final tasks of classifying falls/non-fall and high/low fall risks.

C. TRANSFER LEARNING

As falls tend to be fairly uncommon in real scenarios, fall detection/prediction datasets tend to follow this trend of not being widely available and those that are available

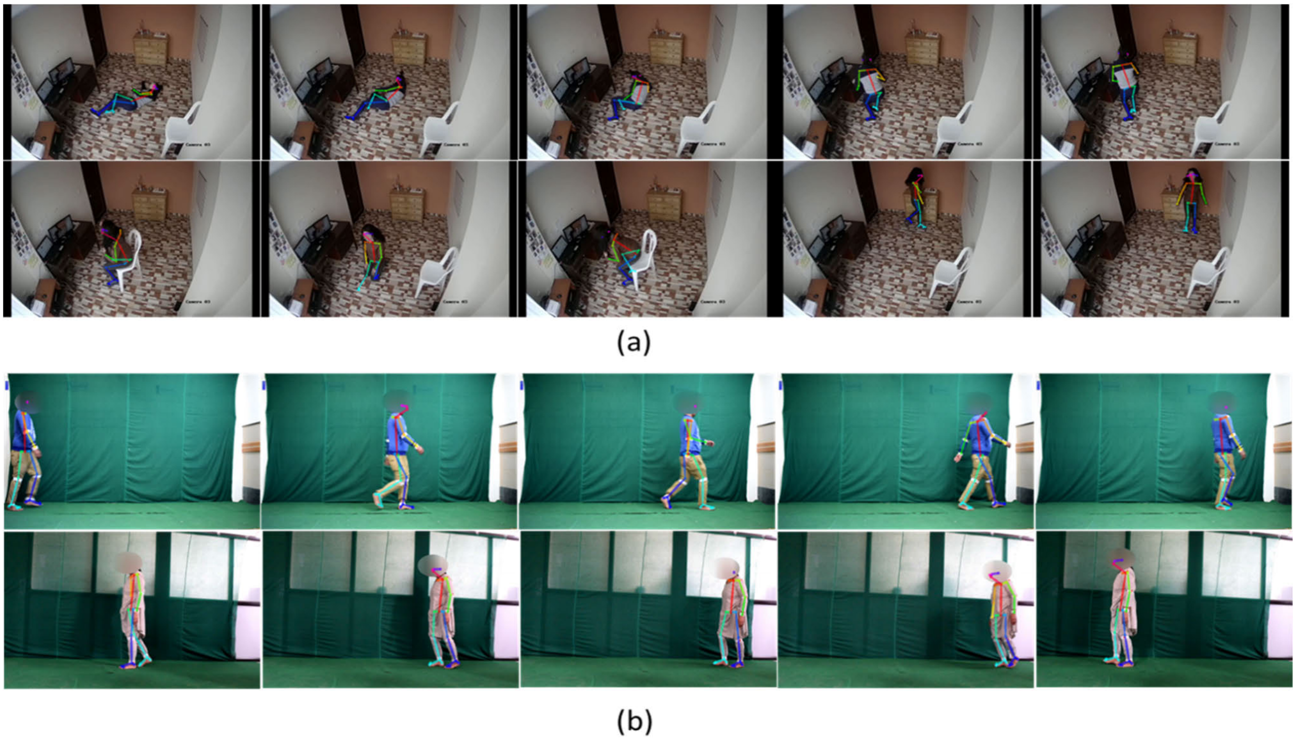


FIGURE 3. Representative frames and pose extraction results for CAUCAFall dataset (a) for fall (top line) and non-fall (bottom line) activities, gait dataset (b) for low-fall risk gait (top line) and high fall risk gait (bottom line).

typically will be smaller compared to those used for general action recognition. This makes it challenging to train complex transformer models as they will likely suffer from under-fitting. In order to address this issue, we decided to incorporate a transfer learning strategy into our work. We first train a transformer model on a large MPOSE dataset as in [33], comprising 15429 samples of 20 actions performed by 100 subjects. This pre-training empowers the model to learn useful feature representations from abundant data for action classification tasks (e.g., classifying fall or non-fall). Then we froze some of the layers of the model, such as the embedding and transformer encoder layers, which contain the majority of the network parameters, while leaving only the last two MLP layers trainable. Finally, we modify the architecture of the network to change the output of 20 actions to 2 for our specific fall detection (fall/non-fall) or fall risk prediction (high/low risk). In this way, we can significantly reduce the number of trainable parameters, allowing the model to be trained with a smaller amount of fall detection and fall risk prediction associated datasets.

IV. EXPERIMENTS

A. MPOSE DATASET EVALUATIONS

This section unveils the evaluation results of our devised methodology for fall detection and prediction. As delineated in Section III-C, our initial step involves the training of our transformer model using a relatively expansive MPOSE dataset, consisting of 15,429 samples representing 20 distinct actions (i.e., walking, jogging, handshaking, etc.) executed by 100 subjects. Each sample comprises 30 frames. We partitioned the dataset into 80% for training and the remaining 20% for testing. Exploring various transformer architectures, namely 'micro', 'small', 'base', and 'large', each progressively increasing in model complexity, as presented in Table 1. The statistical metrics, including mean and standard deviation of accuracy and balanced accuracy, acquired through multiple rounds of evaluation on the MPSOSE dataset, are summarized in Table 2. Notably, our analysis

TABLE 1. Transformer architectures with different complexities.

	micro	small	base	large
No. of heads	1	2	3	4
No. of blocks	4	5	6	6
Embedded Dimension	64	128	192	256
Dimension of MLP Layer	256	256	256	512

TABLE 2. Action recognition comparisons between different architectures on the MPOSE dataset.

	micro	small	base	large
Accuracy	89.55%±0.52%	**89.90%**±0.32%	89.21%±0.57%	88.87%±0.64%
Balanced Accuracy	84.76%±0.79%	**85.51%**±0.59%	84.47%±0.70%	84.37%±1.04%

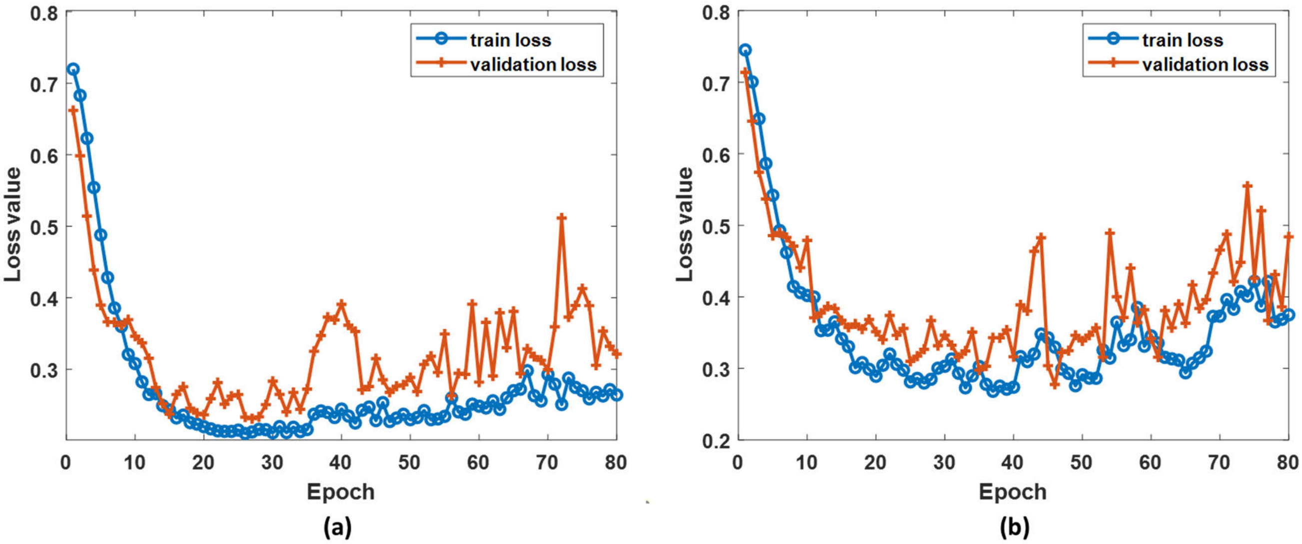


FIGURE 4. The evolutions of the loss function value during training for CAUCAFall dataset (a) and gait dataset (b).

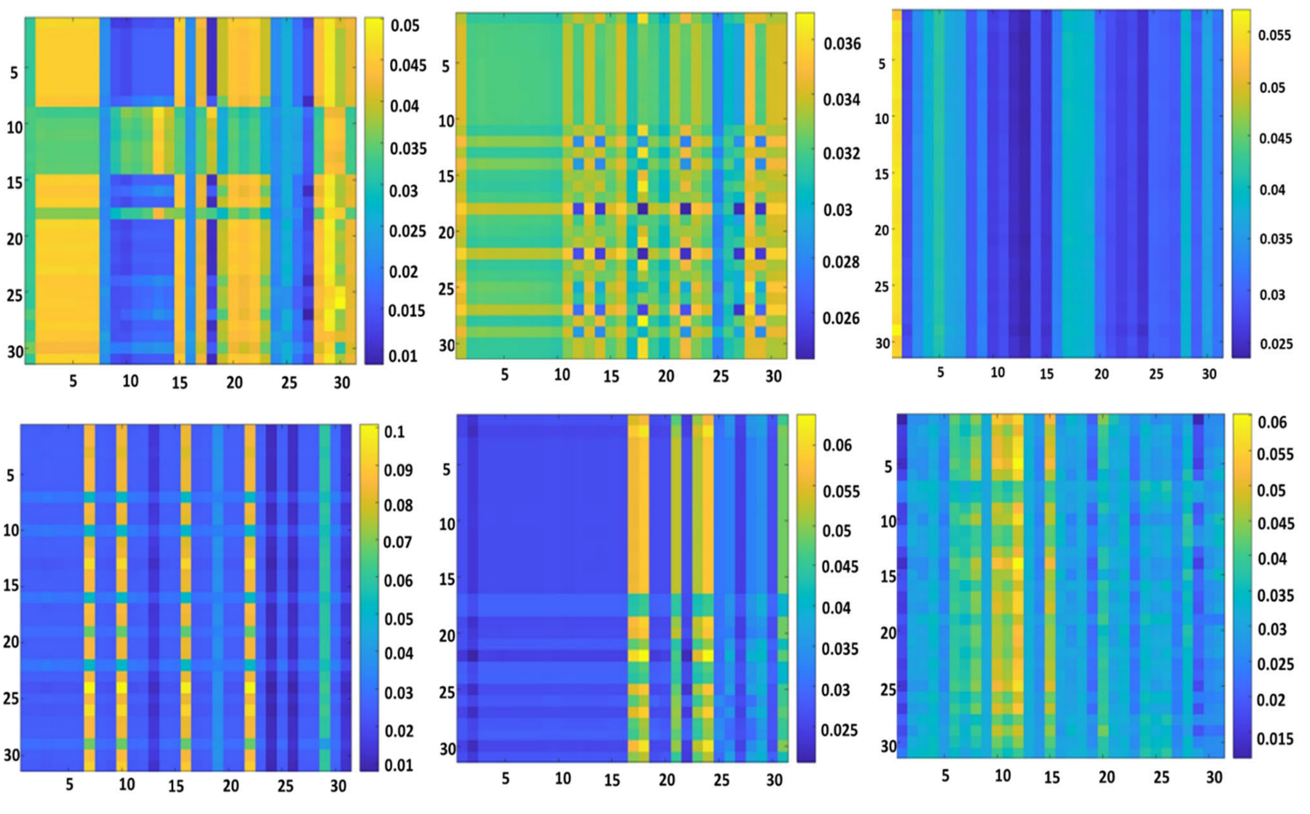


FIGURE 5. Generated attention maps for samples in the CAUCAFall dataset (top row) and gait dataset (bottom row).

reveals that the 'small' transformer architecture yields the highest accuracy, and further augmenting model complexity does not confer any advantages. So, we have chosen the 'small' transformer architecture for fall detection and prediction in our work.

B. DESCRIPTIONS ON FALL DETECTION&PREDICTION DATASETS

We conducted an extensive performance evaluation of our developed methods for both fall detection and fall risk prediction tasks using two distinct datasets: the CAUCAFall

TABLE 3. Comparisons of the proposed methodology with traditional classifiers for the fall detection (the best result is bolded).

	sensitivity	specificity	accuracy
Proposed	**98.37%**±0.01%	**97.58%**±0.02%	**97.95%**±0.87%
SVM	73.68%±0%	83.61%±0%	79.03%±0%
NN	72.25%±0%	88.11%±0%	80.79%±0%
Naive Bayesian	34.45%±0%	90.98%±0%	64.90%±0%
Decision Tree	89.57%±1.13%	95.16%±0.63%	92.58%±0.59%
MLP	87.80%±0.89%	89.63%±0.69%	88.79%±0.49%

TABLE 4. Comparisons of the proposed methodology with deep learning models for the fall detection(the best result is bolded.

	Trainable param.	sensitivity	specificity	accuracy
LSTM	61,402	97.89%±0.89%	96.43%±0.95%	97.11%±0.65%
TCN	78,882	94.11%±0.40%	91.07%±0.01%	92.25%±0.02%
Transformer (without transfer learning)	1,035,778	94.16%±2.39%	95.16%±1.73%	94.70%±1.28%
Proposed	33,538	**98.37%**±1.03%	**97.58%**±1.41%	**97.95%**±0.87%

dataset [34] and a gait dataset [35]. The CAUCAFall dataset is created in conditions of an uncontrolled home environment, with occlusions, changes in lighting (natural, artificial, and night), variety in the clothing of the participants, movement in the background, different textures on the floor and room, variety in the angles of fall, different distances from the camera to the fall, with participants of different age, weight, height, and even different dominant leg. It contains 10 subjects simulating 5 types of falls (forward falls, backward falls, falls to the left, falls to the right, and falls from sitting) and 5 types of activities of daily living (ADLs) including walking, hopping, object retrieval, sitting, and kneeling, which are recorded by a normal RGB camera with the size of approx. 8GB. The recorded video data are organized into 10 main directories corresponding to the subjects, each of which contains 10 folders with the different activities performed by the participants, in each folder there is a video of the action in .avi format, and the images of the actions in .png format, and each of the frame segmentation labels in .txt format. Videos are recorded using a standard RGB camera at a frame rate of 23 frames per second, with a 1080×960 pixels resolution. The gait dataset comprises 96 subjects which involves 50 Knee Osteoarthritis (KOA), 16 Parkinson's Disease (PD), and 30 Normal/Healthy (NM) subjects with different fall risk levels. For each subject, two gait video sequences (left to right and right to left) are recorded using a single NIKON DSLR 5300 camera placed 8m away from the walking mat in the hospital area, with a video resolution of 1920×1080 pixels and a frame rate of 23 frames per second. In total, the video recordings of this dataset have a size of approx. 3GB. More detailed descriptions of these two datasets can be found in [34] and [35].

C. MODEL TRAINING DETAILS

We divided the CAUCAFall and gait datasets into video clips containing 30 frames. We acquired 716 video clips depicting falling activities and 793 video clips showcasing non-fall activities from the CAUCAFall dataset. For the gait dataset, we collected about 2000 video clips representing high fall-risk gait patterns and about 1300 video clips representing low fall-risk gait patterns. Moreover, these obtained video clips were segregated into training and testing subsets, following a distribution of 70% for training and 30% for testing. To extract 2D poses from the original video frames, we used Openpose. Fig. 3 presents some examples of the extracted 2D poses from the original video frames for both datasets. We extract 2D poses for all the video clips in the training dataset and construct vector sequences of pre-processed key points' positions and velocities as per Section III-A. The constructed sequences for all video clips in the training dataset are then used to train the transformer model. For the model training, as mentioned in Section III-C, we froze the transformer encoder, after training our transformer model on the MPOSE dataset, while leaving only the MLP layers embedding layer and trainable. For training, the binary cross-entropy loss function is exploited as below:

$$L(y, p) = -(y \cdot \log(p) + (1 - y) \cdot \log(1 - p)) \quad (6)$$

where y is the true class label and p is the predicted probability. The AdamW optimizer [32] is used to minimize the loss function, with a batch size of, weight decay factor of 0.0001, and an adaptive learning rate strategy with the learning rate being 0.0001 after 80% of the training steps and calculated as in [31] before 80% of the training steps. 10% of the training dataset is taken as the validation dataset and the transformer model which achieves the best performance on the validation dataset is saved for testing.

Fig. 4 shows the loss function value with respect to the training epoch for both the fall detection and gait datasets, from which we can see that at around 20 epochs the loss function almost converges to a minimum. Thus 20 epochs are determined for model training. The training loss values are mostly smaller than the validation loss values (as the model is training on the training dataset, not the validation one).

The transformer model, being an attention-based model, assigns distinct attention weights to elements at various

TABLE 5. Comparisons of the proposed methodology with traditional classifiers for the fall risk prediction (the best result is bolded).

	sensitivity	specificity	accuracy
Proposed	**94.60%**±0.96%	**92.93%**±1.41%	**93.98%**±0.70%
SVM	72.51%±0%	47.43%±0%	63.17%±0%
NN	81.99%±0%	43.90%±0%	67.81%±0%
Naive Bayesian	8.20%±0%	87.53%±0%	37.74%±0%
Decision Tree	86.82%±0.31%	73.33%±1.14%	81.80%±0.52%
MLP	80.10%±3.75%	67.03%±9.44%	75.26%±4.61%

TABLE 6. Comparisons of the proposed methodology with deep learning models for the fall risk prediction (the best result is bolded).

	Trainable param.	sensitivity	specificity	accuracy
LSTM	61,402	92.81%±0.75%	92.17%±1.75%	92.57%±0.73%
TCN	78,882	81.95%±1.84%	71.44%±3.66%	78.03%±1.81%
Transformer (without transfer learning)	1,035,778	83.88%±5.75%	59.91%±16.67%	75.12%±7.62%
Proposed	33,538	**94.60%**±0.96%	**92.93%**±1.41%	**93.98%**±0.70%

positions within a sequence, thus different elements contribute differently to the final transformer output. In Figure 5, we visually depict the generated attention weight maps for several sequences taken from the two datasets. These visual representations vividly highlight the variations in attention weights among elements within a sequence, indicating how different elements contribute differently to the overall transformer output.

D. MODEL EVALUATIONS&COMPARISONS

The trained model is then evaluated on the test datasets. A variety of metrics are applied to evaluate its performance on the test dataset, including sensitivity, specificity, and accuracy, which are defined as below:

$$sensitivity = \frac{TP}{TP + FN} \tag{7}$$

$$specificity = \frac{TN}{TN + FP} \tag{8}$$

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

where TP, TN, FP, and FN represent true positive (correctly classified positive sample), true negative (correctly classified negative sample), false positive (incorrectly classified positive sample), and false negative (incorrectly classified negative sample) respectively.

We compared the performance metrics of our developed method with those of other machine learning models and deep learning models used in other research works for both fall detection and risk prediction. The results are summarized in Tables 3, 4, 5, and 6. Note, that multiple evaluations were performed for each model, and the mean and standard deviation were reported. From these four tables, we can observe that our developed approach achieves much better performance than the traditional machine learning models (SVM, nearest neighbor, Naive Bayesian, decision tree, and multi-layer perception (MLP)) with higher sensitivity, specificity, and accuracy values. Besides, our approach outperforms the other deep learning models (LSTM, temporal convolutional network (TCN), and the Transformer without transfer learning), with even fewer trainable parameters.

TABLE 7. Comparisons of proposed methodology with state-of-the-art ones.

	ST-GCN[31]	2s-AGN [32]	PoseC3D [33]	Proposed
Fall detection Accuracy	91.22% ±2.09%	95.62% ±1.50%	92.27% ±4.86%	**97.95%** ±0.87%
Fall prediction Accuracy	85.99% ±5.18%	89.99% ±2.42%	92.03% ±3.73%	**93.98%** ±0.70%

Moreover, we have also compared our proposed approaches with other state-of-the-art ones originally developed for action recognitions, including the ST-GCN [36], 2s-AGN [37], and PC3D [38], which are trained based on CAUCAFall and gait datasets for performing fall detection& prediction tasks. The comparison results are shown in Table 7, which shows that the proposed approach also achieves the highest accuracies on both fall detection and prediction tasks compared to these three approaches. The advantage of the proposed transformer-based method over graph model-based ST-GCN and 2s-AGN approaches and convolutional model-based PC3D approach can be attributed to the capability of the transformer for modeling the global information of input data as mentioned in [31], thus to extract the most representative features for the whole input skeleton sequence for performing fall detection and prediction tasks.

V. DISCUSSION

We present a pioneering transformer model for fall detection and prediction in this study, leveraging transfer learning to enhance model performance in the presence of limited data. Through extensive evaluations across multiple datasets, focusing on fall detection and prediction tasks, our proposed method demonstrates superior performance compared to both classical machine learning models and state-of-the-art action recognition models. While our proposed method proves effective, there remains room for improvement and potential benefits from recent advancements in the machine-learning community. One avenue for enhancement is the adoption of bilinear pooling [39] to fuse diverse features extracted from a video clip, thereby achieving more precise fall detection and prediction performance. Additionally, we can explore the application of decentralized federated learning [40] to train the machine learning model collaboratively using a cluster of machines. In this approach, each client in the cluster trains a model based on its local data and then communicates the model parameters with other clients, facilitating the aggregation of results from each client to obtain a final global model. This strategy capitalizes on the computational and storage resources of multiple machines, leading to more efficient training on large datasets. Importantly, decentralized federated learning ensures data privacy, as it shares only the model parameters rather than the raw data.

Concerning the implementation of the developed technique for real-world application, the related algorithms can either be deployed on the video analytics unit on the 'edge' (the place where the video data is captured) or a remote cloud server, to process the collected videos on a specific site (e.g., home, care home, hospital ward, etc.) for performing fall detection&prediction tasks. Achieving an optimal balance between algorithm complexity and computational/storage resources is crucial for practical real-world applications. For instances where the intention is to deploy the algorithms on an edge device with limited computational and storage resources, strategic considerations are required. In such cases, adopting a lightweight 2D pose detection model, such as [41], becomes essential. Additionally, simplifying our transformer model further becomes a necessity to align with the constraints of the edge device. This ensures that the overall system remains efficient and effective even within the limitations of the hardware. On the contrary, if the decision is made to deploy the algorithms on a robust cloud server with ample computational and storage capabilities, more complex models can be embraced. The increased processing power of the cloud server allows for the utilization of sophisticated algorithms, enhancing the system's overall performance and accuracy.

Privacy concerns are paramount in any computer vision-based healthcare application. Effectively addressing these concerns necessitates the adoption of encryption techniques. One approach involves employing both software [42] and hardware [43] based encryption methods, which can be applied to blur images within the captured videos, thereby safeguarding privacy. Furthermore, encryption techniques such as SSL (Secure Sockets Layer) and TLS (Transport Layer Security) [44] play a pivotal role in enhancing privacy during the data transmission process to the cloud server. In scenarios where our developed technique is deployed on a cloud server for real-world applications, implementing SSL and TLS encryption ensures a secure and private exchange of data between the edge device and the cloud server. This additional layer of encryption fortifies the protection of sensitive healthcare information during transit, addressing privacy concerns extensively.

VI. CONCLUSION

In this work, we have developed a novel transformer model approach aided by transfer learning for fall detection/prediction tasks. In specific, 2D human skeletons from a video clip are extracted by the on-the-shelf pose extractor, which is then pre-processed and fed into a transformer model for fall detection/prediction. To further improve the performance, a transfer learning approach is adopted to pre-train the transformer model on a larger human motion dataset which step helps the model learn useful representations from abundant data related to human poses and actions. Following the pre-training phase, we fine-tune the transformer model for fall detection and fall risk prediction. The experimental results show that our approach achieves better performance than other machine learning or deep learning models in both fall detection and prediction tasks.

In the current work, we only use the 2D pose features for the fall detection/prediction tasks. Not only limited to the pose features, in the future we will investigate more video features (e.g., optical flow, silhouettes) and investigate the optimal fusion of multiple video features together with the transformer model, to further improve the performance of fall detection/prediction. Moreover, more advanced transformer architecture and machine learning techniques (e.g., bilinear pooling, distributed machine learning) will also be investigated.

REFERENCES

[1] (2022). *U.K. Department of Health & Social Care, Falls: Applying All Our Health. GOV.U.K.* Accessed: Oct. 8, 2023. [Online]. Available: https://www.gov.uk/government/publications/falls-applying-all-our-health/falls-applying-all-our-health

[2] Centers for Disease Control and Prevention. (2023). *WISQARS-Web-Based Injury Statistics Query and Reporting System.* WISQARS Injury Data. Accessed: Oct. 8, 2023. [Online]. Available: https://webappa.cdc.gov/sasweb/ncipc/nfilead.html

[3] T. Al-Aama, "Falls in the elderly: Spectrum and prevention," *Can Fam Physician*, vol. 57, no. 7, pp. 771–776, 2011.

[4] (2021). *World Health Organization, Falls*. Accessed: Oct. 8, 2023. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/falls

[5] R. Broadley, J. Klenk, S. Thies, L. Kenney, and M. Granat, "Methods for the real-world evaluation of fall detection technology: A scoping review," *Sensors*, vol. 18, no. 7, p. 2060, Jun. 2018.

[6] X. Wang, J. Ellul, and G. Azzopardi, "Elderly fall detection systems: A literature survey," *Frontiers Robot. AI*, vol. 7, p. 71, Jun. 2020.

[7] V. Adeli, N. Korhani, A. Sabo, S. Mehdizadeh, A. Mansfield, A. Flint, A. Iaboni, and B. Taati, "Ambient monitoring of gait and machine learning models for dynamic and short-term falls risk assessment in people with dementia," *IEEE J. Biomed. Health Informat.*, vol. 27, no. 7, pp. 3599–3609, Nov. 2023.

[8] J. Gutiérrez, V. Rodríguez, and S. Martin, "Comprehensive review of vision-based fall detection systems," *Sensors*, vol. 21, no. 3, p. 947, Feb. 2021.

[9] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proc. IEEE*, vol. 109, no. 1, pp. 43–76, Jan. 2021.

[10] V. Mazzia, S. Angarano, F. Salvetti, F. Angelini, and M. Chiaberge, "Action transformer: A self-attention model for short-time pose-based human action recognition," *Pattern Recognit.*, vol. 124, Apr. 2022, Art. no. 108487.

[11] M. Yu, S. M. Naqvi, and J. Chambers, "Fall detection in the elderly by head tracking," in *Proc. IEEE/SP 15th Workshop Stat. Signal Process.*, Aug. 2009, pp. 357–360.

[12] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "3D head tracking for fall detection using a single calibrated camera," *Image Vis. Comput.*, vol. 31, no. 3, pp. 246–254, Mar. 2013.

[13] S. Miaou, F. Shih, and C. Huang, "A smart vision-based human fall detection system for telehealth applications," in *Proc. 3rd Int. Conf. Telehealth (IASTED)*, Anaheim, CA, USA, May 2007, pp. 7–12.

[14] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Fall detection from human shape and motion history using video surveillance," in *Proc. 21st Int. Conf. Adv. Inf. Netw. Appl. Workshops (AINAW)*, Nov. 2007, pp. 875–880.

[15] B.-H. Wang, J. Yu, K. Wang, X.-Y. Bao, and K.-M. Mao, "Fall detection based on dual-channel feature integration," *IEEE Access*, vol. 8, pp. 103443–103453, 2020.

[16] S. Taghvaei, M. H. Jahanandish, and K. Kosuge, "Autoregressive-moving-average hidden Markov model for vision-based fall prediction—An application for Walker robot," *Assistive Technol.*, vol. 29, no. 1, pp. 19–27, Jan. 2017.

[17] N. Zerrouki, F. Harrou, A. Houacine, and Y. Sun, "Fall detection using supervised machine learning algorithms: A comparative study," in *Proc. 8th Int. Conf. Modeling, Identificat. Control (ICMIC)*, Nov. 2016, pp. 665–670.

[18] Q. Feng, C. Gao, L. Wang, M. Zhang, L. Du, and S. Qin, "Fall detection based on motion history image and histogram of oriented gradient feature," in *Proc. Int. Symp. Intell. Signal Process. Commun. Syst. (ISPACS)*, Xiamen, China, Nov. 2017, pp. 341–346.

[19] M. Yu, A. Rhuma, S. M. Naqvi, L. Wang, and J. Chambers, "A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment," *IEEE Trans. Inf. Technol. Biomed.*, vol. 16, no. 6, pp. 1274–1286, Nov. 2012.

[20] M. Yu, L. Gong, and S. Kollias, "Computer vision based fall detection by a convolutional neural network," in *Proc. 19th ACM Int. Conf. Multimodal Interact.*, Nov. 2017, pp. 416–420.

[21] N. Lu, Y. Wu, L. Feng, and J. Song, "Deep learning for fall detection: Three-dimensional CNN combined with LSTM on video kinematic data," *IEEE J. Biomed. Health Informat.*, vol. 23, no. 1, pp. 314–323, Jan. 2019.

[22] A. Raza, M. H. Yousaf, and S. A. Velastin, "Human fall detection using YOLO: A real-time and AI-on-the-edge perspective," in *Proc. 12th Int. Conf. Pattern Recognit. Syst. (ICPRS)*, Jun. 2022, pp. 1–6.

[23] O. Keskes and R. Noumeir, "Vision-based fall detection using ST-GCN," *IEEE Access*, vol. 9, pp. 28224–28236, 2021.

[24] J. Ma, "Predicting TUG score from gait characteristics with video analysis and machine learning," in *Proc. Chin. Intell. Automat. Conf.*, 2023, pp. 1–12.

[25] K.-D. Ng, S. Mehdizadeh, A. Iaboni, A. Mansfield, A. Flint, and B. Taati, "Measuring gait variables using computer vision to assess mobility and fall risk in older adults with dementia," *IEEE J. Transl. Eng. Health Med.*, vol. 8, pp. 1–9, 2020.

[26] B. Chen, C. Chen, J. Hu, Z. Sayeed, J. Qi, H. F. Darwiche, B. E. Little, S. Lou, M. Darwish, C. Foote, and C. Palacio-Lascano, "Computer vision and machine learning-based gait pattern recognition for flat fall prediction," *Sensors*, vol. 22, no. 20, p. 7960, Oct. 2022.

[27] A. Masalha, N. Eichler, S. Raz, A. Toledano-Shubi, D. Niv, I. Shimshoni, and H. Hel-Or, "Predicting fall probability based on a validated balance scale," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Seattle, WA, USA, Jun. 2020, pp. 1224–1231.

[28] W. Zhu, B. Peng, H. Wu, and B. Wang, "Query set centered sparse projection learning for set based image classification," *Appl. Intell.*, vol. 50, no. 10, pp. 3400–3411, Oct. 2020.

[29] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime multi-person 2D pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, Jan. 2021.

[30] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16×16 words: Transformers for image recognition at scale," in *Proc. 9th Int. Conf. Learn. Represent.*, 2021.

[31] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. 31st Conf. Neural Inf. Process. Syst. (NIPS)*, Long Beach, CA, USA, 2017, pp. 6000–6010.

[32] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[33] V. Mazzia, S. Angarano, F. Salvetti, F. Angelini and M. Chiaberge. (2021). *MPOSE2021, A Dataset for Short-Time Human Action Recognition*. [Online]. Available: https://github.com/PIC4SeR/MPOSE2021_Dataset

[34] J. Eraso, E. Munoz, M. Munoz, and J. Pinto. (Jul. 2022). *Dataset CAUCAFall*. Mendeley Data. [Online]. Available: https://data.mendeley.com/datasets/7w7fccy7ky/4

[35] N. Kour, S. Gupta, and S. Arora. (Dec. 2020). *Gait Dataset for Knee Osteoarthritis and Parkinson's Disease Analysis With Severity Levels*. Mendeley Data. [Online]. Available: https://data.mendeley.com/datasets/44pfnysy89/1

[36] S. Yan, Y. Xiong, and D. Lin, "Spatial–temporal graph convolutional networks for skeleton-based action recognition," in *Proc. 32nd AAAI Conf. Artif. Intell. (AAAI)*, 2018, pp. 7444–7452.

[37] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Two-stream adaptive graph convolutional networks for skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 12018–12027.

[38] H. Duan, Y. Zhao, K. Chen, D. Lin, and B. Dai, "Revisiting skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, New Orleans, LA, USA, Jun. 2022, pp. 2959–2968.

[39] Y. Gao, O. Beijbom, N. Zhang, and T. Darrell, "Compact bilinear pooling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 317–326.

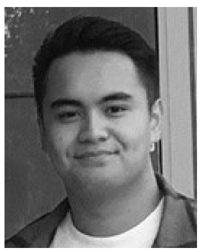
[40] L. Yuan, L. Sun, P. S. Yu, and Z. Wang, "Decentralized federated learning: A survey and perspective," 2023, *arXiv:2306.01603*.

[41] H. Jeon, Y. Yoon, and D. Kim, "Lightweight 2D human pose estimation for fitness coaching system," in *Proc. 36th Int. Tech. Conf. Circuits/Syst., Comput. Commun. (ITC-CSCC)*, Jun. 2021, pp. 1–4.

[42] J. Liu, Y. Xia, and Z. Tang, "Privacy-preserving video fall detection using visual shielding information," *Vis. Comput.*, vol. 37, no. 2, pp. 359–370, Feb. 2021.

[43] L. Gong, S. Mccall, and M. Yu, "Enhancing privacy with optical element design for fall detection," *Electron. Lett.*, vol. 59, no. 20, Oct. 2023, Art. no. e12995.

[44] A. Satapathy and J. Livingston, "A comprehensive survey on SSL/ TLS and their vulnerabilities," *Int. J. Comput. Appl.*, vol. 153, no. 5, pp. 31–38, Nov. 2016.



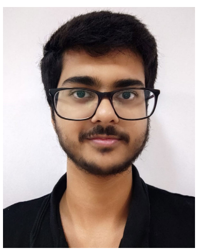
SHELDON MCCALL was born in Bedfordshire, U.K., in 1999. He received the B.Sc. and M.Sc. degrees in computer science from the University of Lincoln, in 2020 and 2021, respectively. He is currently pursuing the Ph.D. degree in computer science, specializing in machine learning and computer vision. Since 2022, he has been a Research Assistant collaborating on several research projects with the School of Computer Science, University of Lincoln. He centered around the development of software artifacts used in machine learning/computer vision projects.

SHINA SAMUEL KOLAWOLE received the degree in mechanical engineering from Manchester Metropolitan University, in 2015, the M.Sc. degree in engineering management from Nottingham Trent University, in 2019, the second master's degree from the University of Lincoln, with a focus on data science and applied analytics, in 2022, and the Ph.D. degree (Hons.), in 2023. As a Research Assistant with the University of Lincoln, he led a groundbreaking project on "Developing Machine Learning Models for Fall Risk Prediction with Open-Pose," utilizing video recordings of gaits. He is a highly accomplished professional with a strong academic background. His research interests include data science, including deep learning, machine learning, statistical analysis, statistical modeling, and computer vision.

AFREEN NAZ received the M.Sc. degree in computer science and the M.S. degree in applied mathematics. She is currently a Research Assistant with the School of Chemistry, University of Lincoln. Her current research involves studying the biology of obesity for the prevention and treatment of age-related diseases.

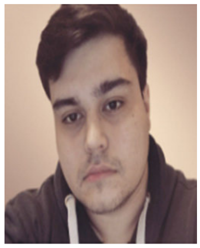
LIYUN GONG received the bachelor's degree in electronic and communication engineering from Liverpool University, Liverpool, U.K., in 2010, where she is currently pursuing the Ph.D. degree with the Department of Electrical Engineering and Electronics. She is a Postdoctoral Research Associate in data science and machine learning with the University of Lincoln. Her research interests include pattern recognition and data mining.

SYED WAQAR AHMED received the B.S. degree in physics specializing in spectroscopy and the M.S. degree in a low-field magnetic resonance spectroscopy. He is currently a Lecturer with the Department of Sciences and Humanities, FAST-National University of Computer and Engineering Sciences, where he teaches applied physics and digital logic design. He is interested in the research field at the junction of physics and computer science. His research interests include physics, machine learning artificial intelligence, quantum computing, and data science.



PANDEY SHOURYA PRASAD is currently pursuing the bachelor's degree in electronics and communication with a specialization in AI/ML with the International Institute of Information Technology Bangalore. With a robust academic background in machine learning, he is particularly enthusiastic about the intricacies of computer vision within the ML domain. His main research interests include computer vision, NLP, and signal processing.

MIAO YU received the Ph.D. degree from the School of Electrical and Electronic Engineering, Loughborough University, with a focus on computer vision based techniques for fall detection with application toward assisted living. From 2013 to 2017, he was a Research Associate for the research project "Signal Processing Solutions in a Networked Battlespace" (funded by the EPSRC) with the Department of Aeronautical and Automotive Engineering, Loughborough University. He is currently a Senior Lecturer with the School of Computer Science, University of Lincoln. His research interests include developing algorithms in statistical signal processing, image/video processing, machine learning, and data/knowledge modeling, with applications in object detection and tracking, behavior recognition, and abnormal detection for healthcare.



JAMES WINGATE is currently a Lecturer with the School of Computer Science and a member of the MLearn Research Group, University of Lincoln. His research interests include machine learning and artificial intelligence, especially looking at latent space analysis or healthcare applications.

SAEID POURROOSTAEI ARDAKANI received the Ph.D. degree in computer science from the University of Bath, with a focus on data aggregation in sensor networks. He was an Assistant Professor in computer science with the University of Nottingham (UNNC) and a member of the Next Generation Internet of Everything Laboratory (NGIoE) and the Artificial Intelligence and Optimisation Research Group. He is currently a Senior Lecturer in computer science with the University of Lincoln, U.K. He is also an Academic Member of the MLearn Research Group and an Associate Member of the Lincoln Centre for Autonomous Systems (L-CAS). To date, he has published two books and more than 100 journal articles, conference papers, book chapters, and technical reports. His research and teaching expertise centers on smart and adaptive computing and/or communication solutions to build collaborative/federated (sensory/feedback) systems in the Internet of Things (IoT) applications and cloud environments. He is also interested in (ML-enabled) big data processing and analysis applications.