

Human Fall Detection from Sequences of Skeleton Features using Vision Transformer

Ali Raza^{1,2}^a, Muhammad Haroon Yousaf^{1,2}^b, Sergio A. Velastin^{3,4}^c and Serestina Viriri⁵^d

¹Department of Computer Engineering, University of Engineering and Technology, Taxila, Pakistan

²Swarm Robotics Lab, National Centre of Robotics and Automation (NCRA), Pakistan

³School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K.

⁴Department of Computer Engineering, University Carlos III, 28911 Leganés, Spain

⁵School of Mathematics, Statistics & Computer Science, University of KwaZulu-Natal, Durban, 4041, South Africa


Keywords: Computer Vision, Fall Detection, Vision Transformers, Event Recognition.


Abstract: Detecting human falls is an exciting topic that can be approached in a number of ways. In recent years, several approaches have been suggested. These methods aim at determining whether a person is walking normally, standing, or falling, among other activities. The detection of falls in the elderly population is essential for preventing major medical consequences and early intervention mitigates the effects of such accidents. However, the medical team must be very vigilant, monitoring people constantly, something that is time consuming, expensive, intrusive and not always accurate. In this paper, we propose an approach to automatically identify human fall activity using visual data to timely warn the appropriate caregivers and authorities. The proposed approach detects human falls using a vision transformer. A Multi-headed transformer encoder model learns typical human behaviour based on skeletonized human data. The proposed method has been evaluated on the UR-Fall and UP-Fall datasets, with an accuracy of 96.12%, 97.36% respectively using RP normalization and linear interpolation comparable to state-of-the-art methods.


1 INTRODUCTION


Human fall detection is a challenging problem that may be tackled in several ways. To tackle the problem, several approaches have been suggested in recent years. The majority of the existing methods intend to recognize human actions or activities but not limited to walking, standing or falling etc. Elderly fall detection is vital among these activities. According to the World Health Organization, around 37.3 million people worldwide suffer fall-related injuries that need medical care each year (Organization et al., 2008). Falls are consistently among leading causes of death all around the world. Although elderly people are most in danger of terrible and deadly falls, children and younger people are also at high risk when they sustain fall injuries. A human fall can be interpreted as a sudden and significant change in the human pose.

Human pose estimation is currently an area of active interest for researchers. There are many recent studies in the literature in this field, especially on the use of machine learning and computer vision methods. Various sensor-based approaches can be used to detect different actions, such as walking, running, standing, sitting, jumping, and falling. The use of computer vision algorithms to detect actual falls has caught interest due to its application in the healthcare sector. Most of the increases in healthcare costs may be associated to the increases in the aging populations in developed countries. This kind of application has grown in popularity due to the implementation of safety measures at high-risk workplaces, shopping malls, hospitals, nursing homes, and other places. Falls are one of the most prevalent causes of injuries, and the elderly are more susceptible to them. Wearable sensors have been previously used for fall detection, and barometers, inertial sensors, and gyroscopes have been used in some systems to identify falls. However, the hardware components are expensive, intrusive and not user-friendly to be worn by people. Computer vision can be considered a more

^a <https://orcid.org/0000-0000-0000-0001>

^b <https://orcid.org/0000-0001-8255-1145>

^c <https://orcid.org/0000-0001-6775-7137>

^d <https://orcid.org/0000-0002-2850-8645>

flexible approach for fall detection and researchers have reported many possible techniques and datasets. Certain studies have demonstrated that videos can be used to detect human activity in general. Vision-based on skeletonized data of human joints, derived from conventional images, are proving popular because they can protect privacy and be more robust to varying environmental factors. These methods also detect falling, walking, running, jumping, jogging, etc. More people can now be recognized in a single image. Probabilistic models are used in computer vision systems to detect falls. A computer vision-based fall detection system uses deep learning or machine learning to recognize and predict human falls based on the human skeleton joint points rather than images used as training data, reducing training time.

A summary of relevant existing research work in the domain of human fall detection is given in Table 1. Article references are given in the first column. The second column shows whether a method can detect activities other than falls. The third column indicates whether a pose detection technique with skeleton detection process is employed. The fourth column indicates whether the algorithm has been implemented to identify activities using image sequences. The fifth column specifies if the detection technique includes depth sensors. The sixth column shows the model/technique applied in each work. The last column indicates which dataset was used in the respective research work. It can be observed that 60% of the surveyed literature uses skeleton data for activity detection, with OpenPose being the most frequently pose estimator. Only 20% of the work can detect multiple activities. The most often chosen public datasets for testing are UR-Fall 30%, UP-Fall 16%, and personal datasets 5%, while 70% of recent publications describe the use of sequences for activity detection.

Considering the complex nature of the human fall detection problem, this paper presents a vision transformer technique to solve the human fall detection problem in a non-intrusive way. For this purpose, BlazePose is used as a pose extractor and backbone (although any other similar pose estimation method may be used). After that, a vision transformer is used as the head of the network. The proposed solution eliminates the need to wear any physical sensors. The approach takes less training time, helps preserving users' privacy, and increases machine learning algorithm efficiency and accuracy. The proposed

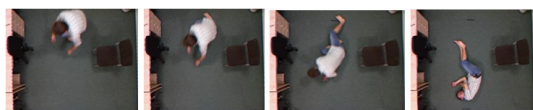


Figure 1: Fall sequence frames.

methodology uses linear interpolation to compensate for missing joint points. A sequence of frames is utilised to identify pose estimations that a single frame cannot identify. Figure 1 shows a fall sequence. The proposed approach uses UR-Fall and UP-Fall and gives improved results that are compared with state-of-the-art methods.

Table 1 show the existing literature of the human fall detection on visual data. Some of the existing work done using pose estimation and use LSTMs and/or CNNs, but to the best of our knowledge vision transformers have not been reported to solve human fall detection using pose estimation.

The key contributions in this work are:

- A vision transformer is proposed for human fall detection. Specifically, we use a multi-headed transformer encoder network for the fall detection.
- BlazePose is used for pose estimation. RGB videos are used to get skeleton features, which are then fed into a multi-headed transformer encoder. This encoder takes a sequence as input, uses a way to pay attention, and can predict where a person will fall.
- When it comes to position normalization and linear interpolation, missing keypoints in the skeletons can be fixed with preprocessing techniques.
- The proposed approach is evaluated on the publicly available UR-FALL and UP-FALL datasets, and results are compared with state-of-the-art methods.

2 PROPOSED APPROACH

The process flow of the proposed technique is given in Figure 2. As input, the network gets a series of video frames, which are then processed by subsequent blocks. Each of these blocks is explained in more detail below.

2.1 Human Pose Estimation

The first step in determining how people fall is to locate a person in video frames and determine the location of their skeleton joints. Human skeletons behave differently in the fall compared to conventional actions. The ability to locate these skeletons can aid the detection of falls in videos. This work uses BlazePose as a method for estimating human poses or skeletons. BlazePose is a machine learning (ML) technique that uses RGB video frames to figure out 33 landmarks and a background segmentation mask. This gives a

Table 1: Summary of Human Fall Detection Works using Visual Data.

Ref	Multi Activity	Skeleton	Sequence	Cam RGB	Model	Dataset
(Chhetri et al., 2021a)	No	No	Yes	Yes	CNN	UR-Fall
(Chen and Duan, 2021)	No	No	Yes	Yes	NanoDet-Lite	UR-Fall
(Cai et al., 2021)	No	No	Yes	Yes	MCCF	UR-Fall
(Sultana et al., 2021)	No	No	Yes	Depth	2D CNN-GRU	UR-Fall
(Leite et al., 2021)	No	No	Yes	Depth	CNN-SVM	UR-Fall
(Keskes and Noumeir, 2021)	No	No	Yes	Depth	ST-GCN	TST-Fall
(Tran et al., 2021)	No	No	No	Yes	CNN-YOLO	CMD-Fall
(Nguyen et al., 2021)	No	No	No	Yes	KNN-SVM	BOMNI
(Berlin and John, 2021)	No	own	Yes	Yes	LSTM	UP-Fall
(Galvão et al., 2021)	No	own	Yes	Yes	AutoEncoder	UP-Fall
(Guan et al., 2021)	No	OpenPose	Yes	Yes	LSTM	UP-Fall
(Kang et al., 2021b)	No	PoseNet	Yes	Yes	GRU	UP-Fall
(Lin et al., 2020)	No	OpenPose	Yes	Yes	LSTM-GRU	UP-Fall
(Chhetri et al., 2021b)	No	OpenPose	Yes	Yes	SVM	UP-Fall
(Dentamaro et al., 2021)	No	OpenPose	Yes	Yes	SVM	UP-Fall
(Chang et al., 2021)	No	OpenPose	Yes	Yes	LSTM	CMU
(Liu et al., 2021)	No	OpenPose	Yes	Depth	RF	SDU-Fall
(Kang et al., 2021a)	No	own	Yes	Yes	GRU	SDU-Fall
(Ramirez et al., 2021a)	No	AlphaPose	No	Yes	KNN	UP-Fall
(Wang et al., 2021)	Yes(7)	Yolov3	Yes	Yes	3D CNN	PKU-MMD
(Zhu et al., 2021)	Yes(4)	OpenPose	Yes	Depth	DNN	FDD
(Yin et al., 2021a)	Yes(8)	own	Yes	Depth	MC-LSTM	TST-Fall
(Ramirez et al., 2021b)	Yes(12)	AlphaPose	No	Yes	RF-SVM MLP-KNN	UP-Fall

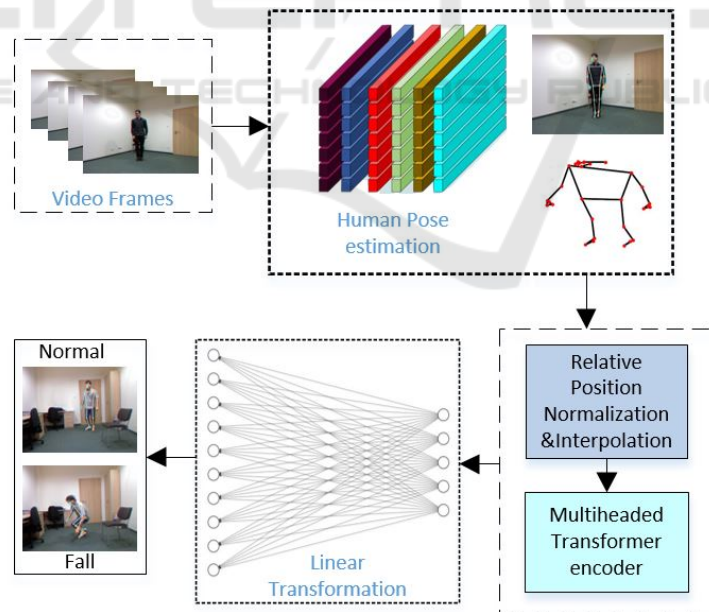


Figure 2: Proposed Approach Pipeline.

good estimate of the body’s pose. Figure 3 illustrates the 33 landmarks.

Blazepose follows a two-step mechanism. First, it detects the person. After the person is detected, it

calculates pose. Figure 4 shows a skeleton detected from a video frame.

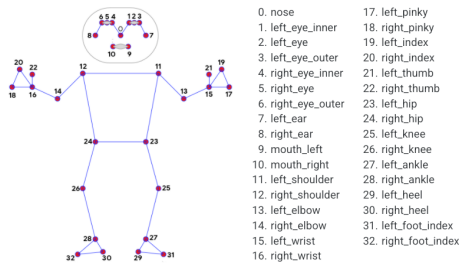


Figure 3: 33 full body landmarks.

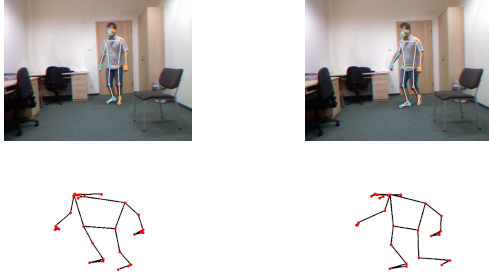


Figure 4: Skeleton detected from a video frame.

2.2 Data Pre-Processing

Pre-processing is useful because data may not be collected in the same unit or format. As a result, the format must be changed, outliers must be removed, missing values must be added, and features must be scaled. Here are the steps that are taken to correct the data format, normalize the data, and locate the missing values.

2.2.1 Relative Position Normalization

To improve the model's accuracy, we use min-max normalization to scale the data to the interval $[0, 1]$ to accelerate the model's convergence. The new X_{nom} scale data are defined as.

$$X_{nom} = \frac{X - X_{min}}{X_{max} - X_{min}} \in [0, 1] \quad (1)$$

The original data is represented by X , the lowest value is represented by X_{min} and the highest value is represented by X_{max} .

However, the dataset contains missing points. Even though not all groups have missing data, the value 0 indicates that X_{min} will equal 0. As a result, the original data distribution will be changed, resulting in ambiguous attributes. Consequently, we suggest the following normalization approach to convert the initial coordinate position to the relative coordinate positions in the f_h frame.

$$x_c^f = \frac{W}{2} \quad (2)$$

and

$$y_c^f = \frac{H}{2} \quad (3)$$

where W and H are the width and height of the images, respectively. When x_n^f or y_n^f point equals 0, it is considered to be a missing point and is not calculated. The calculated center point will replace the missing center point if it is a missing point. The distances (x_{dis}, y_{dis}) that need to be displaced are defined as

$$x_{dis}^f = x_8^f - x_c^f \quad (4)$$

and

$$y_{dis}^f = y_8^f - y_c^f \quad (5)$$

The displacement is to the center point where the eighth joint point is (x_8^f, y_8^f) and the central hip circumference is used as the reference point (x_c^f, y_c^f) . When the displacement value is less than zero, the object moves to the right. When the displacement value is more than zero, the object goes to the left.

The new joint point coordinates (rx_n^f, ry_n^f) for computing the displacement to the relative location are then given as

$$rx_n^f = (x_n^f - x_{dis}^f) \quad (6)$$

and

$$ry_n^f = (y_n^f - y_{dis}^f) \quad (7)$$

Two advantages can be gained from this method. First, the original data distribution is preserved, and secondly the missing points do not need to be computed simultaneously. In addition, any unnecessary features should be eliminated. Aspects of the human body displacement process, such as left-right or far-near walking, fix the human skeleton in the same position and give continuous motions to facilitate the training of the model.

As shown in Figure 5, the image is resized to 640 by 480 pixels and its center is located. The human skeleton will be moved to the image's center after displacement, as shown in Figure 5, and the central hip-joint point will be similar to the position of the center point. Some joints in the skeleton were considered irrelevant for distinguishing fall and non-fall events. Thus, they were eliminated because an excessive number of feature parameters may confuse the process and diminish the recognition rate. So, the only significant points that are retained for training are: the nose, shoulders, elbows, wrists, neck, hips, knees, and ankles.

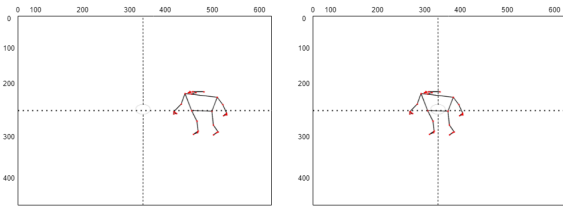


Figure 5: After displacement, the location and number of joint points.

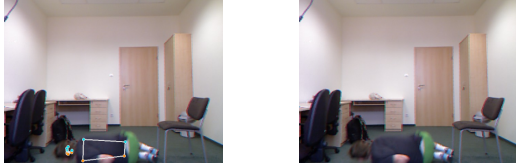


Figure 6: Missing skeleton detected from a video frame.

2.2.2 Settlement for Missing Keyoints

BlazePose uses 2D image features to identify the skeleton. Therefore, instability and incorrect prediction errors might arise when these points are absent. As seen in Figure 6, if the human body is obscured or concealed, its contour is unclear, or if the upper and lower limbs of the human body are occluded, it will result in loss and errors during skeleton construction. The proposed approach uses linear interpolation to adjust for the missing data. To ensure adequate data, we designate 100 images as a group and use the initial non-missing value data to interpolate missing values in succeeding images. The interpolation method may estimate the following data value from the original data. If the initial number of joint points is inadequate, the predicted missing value and the real value will differ significantly. Linear interpolation calculates the slope between two known data points using two known data points. The approximate values for Y are given by,

$$Y = Y_0 + \frac{Y_1 - Y_0}{X_1 - X_0} (X - X_0) \quad (8)$$

Figure 6 shows that the discrepancy between the estimated and real value is not statistically significant when the turning point is not missing. Before RP normalization, the total dataset's missing key points accounted for 22.38%, and after RP-Normalization, missing key points accounted for 13.4%. Moreover, after interpolation + RP-Normalization, the missing key points are only 1.8% of the total.

2.3 Multi-Headed Transformer Encoder

The multi-head self-attention mechanism is the major component of the transformer encoder. In the

proposed technique, the transformer perceives the encoded representation of the input in the form of a sequence vector. Skeleton features are used to encode the input.

$$\text{Shape} = [S, N, D] \quad (9)$$

In this case, S represents the sequence dimension number of each frame in the video, N represents the stack of multiple videos into a batch, and D represents the vector dimension, 512 in this case. The multi-

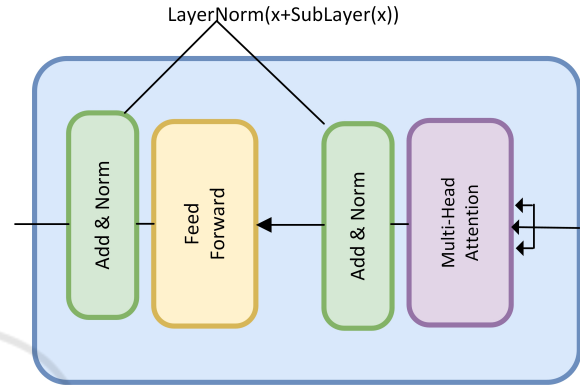


Figure 7: Multi-Head Transformer encoder.

head mechanism processes scaled dot-product attention numerous times in parallel rather than just once. The encoder generates an attention-based representation, which can locate a single piece of data within an infinitely large context by concatenating and linearly converting the separate attention outputs.

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = [\text{head}_1; \dots; \text{head}_h] \mathbf{W}^O \quad (10)$$

$$\text{where head}_i = \text{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \quad (11)$$

$$\text{where } \mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V, \text{ and } \mathbf{W}^O \quad (12)$$

2.4 Linear Transformation Layer

In addition to the multiheaded transformer encoder, a linear transformation layer is applied to the skeleton regression to predict the output.

3 EXPERIMENTATION AND RESULTS

3.1 Datasets

The UR-FALL (Alarifi and Alwadain, 2021) and UP-FALL (Martínez-Villaseñor et al., 2019a) datasets are

used in this research. Due to varying lighting conditions, occlusion and the size of the dataset, the UR-Fall dataset is challenging. This dataset is not of real events but comprises actors performing everyday activities and falling. It contains 70 sequences (30 falls and 40 daily life activities). The dataset was captured using two Microsoft Kinect cameras and an accelerometer. The UP-FALL dataset is also publicly available and contains 11 activities and different stages in falls on three of those activities. Both of the datasets have a frame rate of 25 frames per second, and the resolution is 320x240 pixels.

3.2 Experimental Setup

The proposed approach is trained and tested on a Core i7 machine equipped with an NVIDIA Quadro P5000 GPU. The extracted skeletons have 33 joints per human body and the Pytorch framework is used for the implementation.

3.3 Results and Discussion

In the proposed approach, skeleton poses are applied to detect and distinguish between falling and not falling activities. RP normalization with interpolation is used to detect missing key points, and it provides better results than those from the method without normalization and interpolation. Table 3 shows accuracy, precision, recall, and F1 score without normalization, min-max normalization, and RP normalization with interpolation for UR-FALL. In this case, linear interpolation + RP normalization increased accuracy, precision, and F1 score to 96.12%, 96.05%, and 96.04% respectively.

The proposed approach is also used on the UP-Fall dataset which gives slightly better results as compared to the UR-Fall dataset because UR-Fall dataset include two type of fall sitting from chair and standing other activities are sitting crouching down, sitting. On the other hand UP-Fall dataset has more fall types which include forward using knees forward, backwards sitting forward using hands other activities are sitting, picking up an object, jumping laying, walking, standing and kneel down. Table 4 shows accuracy, precision, recall, and F1 score without normalization, min-max normalization, and RP normalization with interpolation for UP-FALL. It shows improvements in accuracy, recall and F1-score to 97.36%, 97.15% and 96.47% respectively.

Figure 9 shows the confusion matrices of all falls and everyday normal events using different processes. For UR-Fall without normalization, accuracy is 94.87% and after min-max normalization accuracy

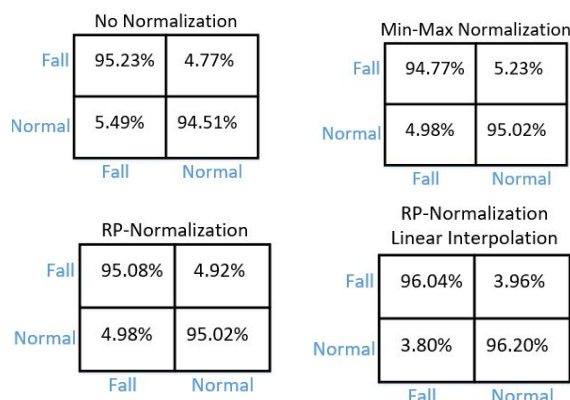


Figure 8: Confusion Matrices UR-FALL.

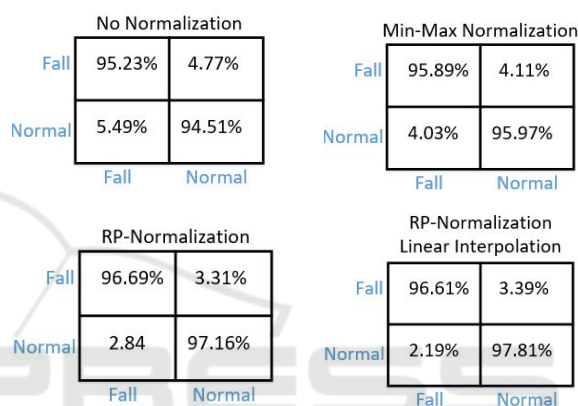


Figure 9: Confusion Matrices UP-FALL.

increases to 95.13% and after RP normalization there is another small increase to 95.51% and after linear interpolation and RP normalization accuracy is higher, 96.12%, as compared to the other methods. Figure 10 illustrates a prediction of a fall and a daily life normal activities where Figure 10a shows an example of the detection of a human fall and Figure 10b shows an example of a detection of a normal daily life activity such as walking.

3.4 Comparison with Existing Work

In Table 4, fall detection approaches are compared to other reports in the literature. Chhetri et al. (Chhetri et al., 2021a) use a CNN method to detect human falls. This method was reported to be 95% accurate on the UR-Fall dataset, which is less accurate than our method. A multiheaded transformer encoder with RP normalization and interpolation is used in our proposed approach to estimate human poses from skeleton sequences. In comparison with the state of the art method, this method produces better accuracy results of 96.12 percent in the UR-Fall dataset and 97.36 percent in the UP-FALL dataset.

Table 2: Evaluation results for the proposed method on UR-FALL Dataset.

Evaluation Measures %	No Normalization %	Min-Max Normalization %	RP Normalization %	Linear Interpolation +RP Normalization %
Accuracy	94.87	95.13	95.51	96.12
Precision	95.24	94.76	95.83	96.05
Recall	94.40	95.04	96.08	95.09
F1-score	94.82	95.01	95.57	96.04

Table 3: Evaluation results for the proposed method on UP-FALL Dataset.

Evaluation Measures %	No Normalization %	Min-Max Normalization %	RP Normalization %	Linear Interpolation +RP Normalization %
Accuracy	95.89	96.52	96.98	97.36
Precision	95.54	96.30	96.70	96.61
Recall	95.38	96.00	96.70	97.15
F1-score	95.47	96.15	96.26	96.47

Table 4: Comparison with other methods.

Methods	Dataset	Methodology	Skeleton Sequence	Accuracy %
(Chhetri et al., 2021a)	UR-Fall	CNN	Yes	95.00
(Chen and Duan, 2021)	UR-Fall	NanoDet-Lite	X	91.2
(Sultana et al., 2021)	UR-Fall	CNN	X	94.00
(Leite et al., 2021)	UR-Fall	CNN-SVM	X	94.96
(Martínez-Villaseñor et al., 2019a).	UP-Fall	RF,SVM MLP,KNN	Yes	32.33,34.40 27.08,34.03
(Yin et al., 2021b)	UR-Fall	SVM	Yes	91.07
(Martínez-Villaseñor et al., 2019b)	UP-Fall	MLP,CNN	X	95.0 ,95.1
Our Proposed	UR-Fall	Multiheaded Vision Transformer	Yes	96.12
Our Proposed	UP-Fall	Multiheaded Vision Transformer	Yes	97.36

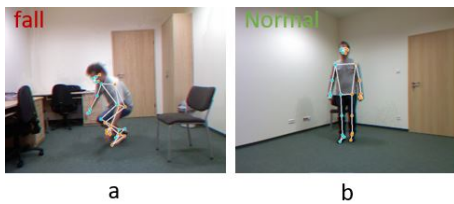


Figure 10: Prediction results (a) corresponds to a human fall sequence and (b) a normal sequence.

4 CONCLUSIONS

This paper has presented a computer vision-based fall detection method using a vision transformer. A multi-headed transformer uses skeletons to provide the desired prediction. A multiheaded transformer encoder network is used to model how a person falls based on a human skeleton feature vector. Missing joint points are addressed using min-max normalization,

RP normalization, and instate-of-the-artpared to the work that has been done so far for the detection of human falls, we achieved state-of-the-art results. Future work is planned to use Le2i for training and testing the proposed network. We also plan to extend the work to realistic (not acted out) environments including multiple people, like shopping malls, public places, homes, etc. Implementing an edge platform for real-time processing is also in the future pipeline.

REFERENCES

- Alarifi, A. and Alwadain, A. (2021). Killer heuristic optimized convolution neural network-based fall detection with wearable iot sensor devices. *Measurement*, 167:108258.
- Berlin, S. J. and John, M. (2021). Vision based human fall detection with siamese convolutional neural networks. *Journal of Ambient Intelligence and Humanized Computing*, pages 1–12.

- Cai, X., Liu, X., An, M., and Han, G. (2021). Vision-based fall detection using dense block with multi-channel convolutional fusion strategy. *IEEE Access*, 9:18318–18325.
- Chang, W.-J., Hsu, C.-H., and Chen, L.-B. (2021). A pose estimation-based fall detection methodology using artificial intelligence edge computing. *IEEE Access*, 9:129965–129976.
- Chen, G. and Duan, X. (2021). Vision-based elderly fall detection algorithm for mobile robot. In *2021 IEEE 4th International Conference on Electronics Technology (ICET)*, pages 1197–1202. IEEE.
- Chhetri, S., Alsadoon, A., Al-Dala'in, T., Prasad, P., Rashid, T. A., and Maag, A. (2021a). Deep learning for vision-based fall detection system: Enhanced optical dynamic flow. *Computational Intelligence*, 37(1):578–595.
- Chhetri, S., Alsadoon, A., Al-Dala'in, T., Prasad, P., Rashid, T. A., and Maag, A. (2021b). Deep learning for vision-based fall detection system: Enhanced optical dynamic flow. *Computational Intelligence*, 37(1):578–595.
- Dentamaro, V., Impedovo, D., and Pirlo, G. (2021). Fall detection by human pose estimation and kinematic theory. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2328–2335. IEEE.
- Galvão, Y. M., Portela, L., Ferreira, J., Barros, P., Fagundes, O. A. D. A., and Fernandes, B. J. (2021). A framework for anomaly identification applied on fall detection. *IEEE Access*, 9:77264–77274.
- Guan, Z., Li, S., Cheng, Y., Man, C., Mao, W., Wong, N., and Yu, H. (2021). A video-based fall detection network by spatio-temporal joint-point model on edge devices. In *2021 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 422–427. IEEE.
- Kang, Y., Kang, H., and Kim, J. (2021a). Fall detection method based on pose estimation using gru. In *International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing*, pages 169–179. Springer.
- Kang, Y. K., Kang, H. Y., and Weon, D. S. (2021b). Human skeleton keypoints based fall detection using gru. *Journal of the Korea Academia-Industrial Cooperation Society*, 22(2):127–133.
- Keskes, O. and Noumeir, R. (2021). Vision-based fall detection using st-gcn. *IEEE Access*, 9:28224–28236.
- Leite, G. V., da Silva, G. P., and Pedrini, H. (2021). Three-stream convolutional neural network for human fall detection. In *Deep Learning Applications, Volume 2*, pages 49–80. Springer.
- Lin, C.-B., Dong, Z., Kuan, W.-K., and Huang, Y.-F. (2020). A framework for fall detection based on openpose skeleton and lstm/gru models. *Applied Sciences*, 11(1):329.
- Liu, C., Lv, J., Zhao, X., Li, Z., Yan, Z., and Shi, X. (2021). A novel key point trajectory model for fall detection from rgb-d videos. In *2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pages 1021–1026. IEEE.
- Martínez-Villaseñor, L., Ponce, H., Brieva, J., Moya-Albor, E., Núñez-Martínez, J., and Peñafort-Asturiano, C. (2019a). Up-fall detection dataset: A multimodal approach. *Sensors*, 19(9):1988.
- Martínez-Villaseñor, L., Ponce, H., Brieva, J., Moya-Albor, E., Núñez-Martínez, J., and Peñafort-Asturiano, C. (2019b). Up-fall detection dataset: A multimodal approach. *Sensors*, 19(9):1988.
- Nguyen, V. D., Pham, P. N., Nguyen, X. B., Tran, T. M., and Nguyen, M. Q. (2021). Incorporation of panoramic view in fall detection using omnidirectional camera. In *The International Conference on Intelligent Systems & Networks*, pages 313–318. Springer.
- Organization, W. H., Ageing, W. H. O., and Unit, L. C. (2008). *WHO global report on falls prevention in older age*. World Health Organization.
- Ramirez, H., Velastin, S. A., Fabregas, E., Meza, I., Makris, D., and Farias, G. (2021a). Fall detection using human skeleton features.
- Ramirez, H., Velastin, S. A., Meza, I., Fabregas, E., Makris, D., and Farias, G. (2021b). Fall detection and activity recognition using human skeleton features. *IEEE Access*, 9:33532–33542.
- Sultana, A., Deb, K., Dhar, P. K., and Koshiba, T. (2021). Classification of indoor human fall events using deep learning. *Entropy*, 23(3):328.
- Tran, T.-H., Nguyen, D. T., and Nguyen, T. P. (2021). Human posture classification from multiple viewpoints and application for fall detection. In *2020 IEEE Eighth International Conference on Communications and Electronics (ICCE)*, pages 262–267. IEEE.
- Wang, K., Li, X., Yang, J., Wu, J., and Li, R. (2021). Temporal action detection based on two-stream you only look once network for elderly care service robot. *International Journal of Advanced Robotic Systems*, 18(4):17298814211038342.
- Yin, J., Han, J., Xie, R., Wang, C., Duan, X., Rong, Y., Zeng, X., and Tao, J. (2021a). Mc-lstm: Real-time 3d human action detection system for intelligent health-care applications. *IEEE Transactions on Biomedical Circuits and Systems*, 15(2):259–269.
- Yin, J., Han, J., Xie, R., Wang, C., Duan, X., Rong, Y., Zeng, X., and Tao, J. (2021b). Mc-lstm: Real-time 3d human action detection system for intelligent health-care applications. *IEEE Transactions on Biomedical Circuits and Systems*, 15(2):259–269.
- Zhu, N., Zhao, G., Zhang, X., and Jin, Z. (2021). Falling motion detection algorithm based on deep learning. *IET Image Processing*.