Human Fall Detection for Smart Home Caring using Yolo Networks

Bo LUO[1]
Engineering Training Center, Chongqing Kechuang Vocational College
Yongchuan 402160, Chongqing, China

***Abstract*—In order to help the elderly and limit the incidence of falls that result in injuries, effective fall detection in smart home applications is a challenging topic. Many techniques have been created employing both vision and non-vision-based technologies. Many researchers have been drawn to the vision-based technique amongst them because of its viability and application. However, there is still room for improvement in the effectiveness of fall detection given the poor accuracy rate and high computational cost issues with current vision-based techniques. This study introduces a new dataset for posture and fall detection, whose photo images were gathered from Internet resources and data augmentation. It employs YOLO networks for fall detection purpose. Furthermore, different YOLO networks are implemented on our dataset to address the most accurate and effective model. Based on assessment parameters including accuracy, F1 score, recall, and mAP, the performance of the various YOLOv5n, s and YOLOv6s versions are compared. As experimental results showed, the YOLOv5s performed better than other.**

Keywords*—YOLO; computer vision; fall detection; smart home; caring

I. INTRODUCTION

In recent years, improvements in information and communication technologies (ICTs) and the innovations that followed them have significantly altered people's lives and given rise to elegant surroundings, cities, and societies. By observing our environments and making decisions to produce desired results, technologies like artificial intelligence (AI) and the Internet of Things (IoT) improve our quality of life. As the foundation of cities and societies, houses play a crucial role in developing smart living. They are anticipated to be major enablers of smart cities and societies [1]. Life expectancy is rising, and fertility is significantly declining in the modern world due to several socioeconomic causes [2]. An automated home-based solution to lessen the pressure on staffed health services and give frequent insight into fall risk would be an alluring alternative strategy [3] as more senior people struggle to preserve their independence and live in their own homes. At the same time, healthcare difficulties are becoming more and more significant due to the rise in the number of senior people worldwide. Elderly individuals who live alone must use human motion capture technology to address these problems. Also, by observing an elder's posture, it is possible to track how healthy they are, and if high-risk postures, such as falling over, are noticed, a warning may be sent. In addition to increasing the effectiveness of posture recognition, these systems will lighten the workload on human resources [4]. Due to considerations including shifting viewing angles, body occlusion, and significant variations in human posture, figuring out how to recognize human posture is extremely difficult [5].

The three basic categories are sensor-based, vision-based, and radio-based HAR [5]. Sensor-based types rely on information gathered by sensors to identify human activity. For instance, a light sensor's activation can signal movement in a certain region or activity. Vision-based types rely on picture and video data formats to identify human activity. As an illustration, motions in smart home recordings imply doing specific tasks like cooking or strolling. Radio-based technology relies on the information and characteristics of signals to detect human activity. Wearable motion sensors that track the movement of body parts might detect particular behaviors like sitting or walking [6].

Systems for recognizing gestures and actions based on video analysis have been thoroughly investigated. Since vision-based data are less expensive and simpler to gather than sensor-based data, the most recent research has been on vision-based HAR. Thus, this study only partially covers the vision-based HAR investigations; it only includes a limited and representative sample. HAR is used in many applications, including surveillance systems, behavior analysis, gesture recognition, patient monitoring systems, and ambient assisted living (AAL). Recent research has focused on fall detection rather than fall risk prediction [7].

Falls are a common occurrence in the course of human growth. Falls happen when kids learn to stand, walk, climb, run, and engage in other activities. In a similar vein, falls also grow more common as we age. Though most falls are minor mishaps, as people age, falls become far more common and serious. It frequently underestimates falls' effect on people and society since they are just a natural part of life [8] [9]. Worldwide, falls are a significant health and financial problem. While falls inflict a high cost on healthcare systems in terms of in-patient and long-term care, they can also have indirect psychological impacts, such as reducing or avoiding physical activity out of fear of falling, in addition to the immediate direct medical repercussions. From an economic standpoint, it is obvious that any additional demand on the health system has a direct financial cost to society; nevertheless, lost productivity also has a hidden cost that is frequently disregarded. It should be no surprise that fall detection research and innovation have been sparked by the frequency and effects of falls, the risk of mortality, healthcare expenditures, and lost social and economic output [8].

The HAR problem has long been addressed using machine learning (ML) techniques, including random forest (RF), Bayesian networks, Markov models, and support vector machines (SVM). Traditional ML algorithms have demonstrated impressive performance in tightly regulated settings with little input data. However, they could be more efficient and take a lot of time to produce since they need several pre-processing stages and appropriate hand-crafted characteristics. External feature use also results in subpar incremental learning or unsupervised learning results. Due to its remarkable performance in several study domains, including object detection and identification, picture classification, and natural language processing (NLP), deep learning has attracted much community attention in recent years. Deep learning significantly decreases the effort required to select the best features compared to typical machine learning algorithms. The deep learning framework has also been effective with unsupervised learning and reinforcement learning. As a result, more and more newly released HAR frameworks are through deep learning [7] .

Deep learning, particularly the convolutional neural network (CNN), which is modeled after the hierarchical processing of the human visual cortex, has had great success in the last few years when it comes to classifying images. CNN is a useful feature extraction and classification technique because it can automatically learn discriminative features from training data [10]. One or more of CNN's exciting application areas include speech recognition, object detection, video processing, object classification and segmentation, and natural language processing [11].

The two primary kinds of target detection algorithms based on CNN are two-stage detection algorithms, which divide target detection into two steps: finding and recognizing. The traditional technique, known as the Region-Convolutional Neural Network (R-CNN), performs poorly and cannot keep up with real-time demands. Fast regions with CNN (Fast R-CNN) and faster regions with CNN (Faster R-CNN) are offered because of further advancements based on R-CNN, although they still need to satisfy people's demands for real-time performance. The other is a one-stage detection technique, which combines target placement and recognition into a single phase for optimization. Single-shot multi-box detector (SSD) and YOLO series models are classic examples of this method [12].

In this paper, different versions of YOLO networks are investigated for fall detection which is utilized in smart home application for elderly caring. This investigation intends to address the most of less memory space usage and highest accuracy detection rate in fall detection.

The format of this essay is as follows: The background is offered in Section II, the technique is described in Section III, along with the training and testing procedure, the discussion and analysis of the findings are found in Section IV, and the conclusion and future work are found in Section V.

II. Related Works

Several recent deep learning-based approaches have been reported to improve the accuracy of fall detection recognition. In this section, we will have a brief overview of several methods of this type.

The suggested approach establishes a framework for fall protection research. In contrast to providing raw data to the cloud for real-time prediction of fall occurrences, [13] offered a framework for fall detection through LSTM networks that utilized edge devices like a laptop for computation. The suggested system used the open-source Apache Flink streaming engine, a low-cost MetaMotionR sensor from MbientLab, and three-axis accelerometer raw data. The architecture has been trained and tested using a portion of the public MobiAct dataset. The created system found that the waist was the ideal location for placing sensors. With a 95.8% accuracy rate, the suggested framework can forecast autumn occurrences from current fall data. Performance can be improved by using many sensors and data streams.

The study [14] has presented a fall detection system design with health monitoring features. Utilizing low-power enabled low-power wide-area network (LPWAN) technology, the system combined Edge computing, Fog computing, and a compression method to transport data. This reduced the latency of the system. LSTM and RNN networks have been developed on the edge computer to identify falls from the incoming data. Raw data is delivered to the cloud via these edge gateways for online analysis, along with real-time notifications and alarms. The suggested architecture extends battery life and allows operation in regions with weak network access. Using the suggested approach and the MobiAct dataset, fall event predictions have been made with an average accuracy of 95% and a precision of 90%. The system performance may be increased by making a few adjustments and combining various techniques.

To more reliably and speedily identify fall behavior, [15] provides a fall detection approach based on a video in a complicated environment. The following is the paper's primary contribution: First, a YOLOv3 network model for the detection method is suggested. Second, the Pascal VOC data set format creates the human fall detection data set. Next, a self-built data set is used to train and test the network model on a GPU server. The experimental data demonstrates that the algorithm's mAP is 0.83 and its AP of down is 0.97, which are better than other conventional algorithms and have a strong resilience and detection impact.

In [16], a noise-tolerant FDS is shown operating with missing values in the data. This study seeks to use DL techniques for wearable sensor-based fall detection when faced with difficulty identifying missing values in data. On the SisFall and UP-Fall datasets, two public accessible datasets, the suggested method applies RNN with an underlying stack of BiLSTM blocks. When a value in the sensor data disappears, BiLSTMs can quickly obtain the long-range context thanks to their innate capacity to store long-term dependencies from both the past and the future. This explains why BiLSTM is a good fit for our noise-tolerant fall detection system that uses sequential sensor data. On the benchmark datasets for SisFall and UP-Fall Detection, the system outperforms the current state of the art with the accuracy, sensitivity, and specificity values of 97.21%, 97.41%, 96.38%, and 91.45%, respectively.

Due to its ability to maintain long-term dependencies from the past and future, the results show that BiLSTM is an appropriate model option to manage missing variables for wearable fall detection systems.

Based on the Fast Pose Estimation approach, [17] suggests a revolutionary human fall detection solution. The method employs 1Dimentional Convolutional Neural Network (1D-CNN) and Time-Distributed Convolutional Long Short-Term Memory (TD-CNN-LSTM) models to categorize retrieved data from picture frames accurately. As a result, the suggested approach effectively contributes to accurate human fall detection by employing the Fast Pose Estimation technique. The original URFD films were subjected to rotation, brightness, horizontal flip, and gamma correction augmentation procedures for dataset preparation. The result was an improved version of the URFD dataset. This resulted in a total of 560 movies, comprising 240 videos of falls and 320 videos of daily activities, which were then used to evaluate the produced models.

III. METHODOLOGY

A. YOLO based Networks

Basically, YOLO is a pre-trained object detector that is trained to recognize everyday objects like tables, chairs, cars, phones [18]. To create a model that can detect human falls, we used different versions of yolo, among which the yolov5s model obtained better results.

1) YOLO v5 network: The YOLO v5 target detection model, which avoids the recomputation of candidate areas in the two-stage series and has high identification precision and quick inference, is the most representative target detection model in the one-stage series. The four primary model structures of the YOLO v5 architecture are YOLO v5l, YOLO v5x, YOLO v5m, and YOLO v5s [19], with decreasing order of network complexity. The YOLO v5n model, which has just 1.9 MB parameters and the same model depth as the YOLO v5s model but a network width half that of the YOLO v5s, was later developed to adapt the solution to mobile devices.

The backbone, neck, and head are the three fundamental parts of the YOLO v5 basic architecture. Fig. 1(a) to 1(d) show the module composition for the basic design. A CNN, which creates visual characteristics by combining many fine-grained pictures, is one of the Backbone structures. Convolution operations such as 2D convolution, 2D regularization, and SiLU activation are carried out by the conv module, which serves as the architecture's fundamental convolution unit [20].

2) YOLO v6 Network: The second autonomous detector used in this investigation is the single-stage object detection framework for industrial applications, YOLOv6, the most recent member of the YOLO family. Although this model does not belong to the official YOLO series, it reportedly exceeds YOLOv5 in terms of detection precision and inference speed, making it more effective for industrial applications. YOLOv6 includes many improvements in the Backbone, Neck, Head blocks and also training strategies. For instance, the Neck and Backbone in YOLOv6 have been redesigned by using Rep-PAN and EfficientRep structures, respectively, based on the idea of hardware-aware neural network. EfficientRep Backbone can make use of hardware computing power, such as GPU, and it also has strong representation capabilities. Rep-Pan Neck, is more accurate and faster than PANet and SPP. Yolov6 Head is decoupled by adding a layer between the network and the final Head, which in turn improves the performance during the training process [21]. The architecture of YOLOv6 is shown in Fig. 2.

B. Dataset

A bespoke fall detection dataset containing two directory pictures and labels was made using photographs gathered from various sources. The study's classifications refer to sitting, walking, and falling. Two subdirectories—Training (333 photos) and Val (111 images)—are included in the images directory and are utilized for different purposes. Here, we have text files with labels for that specific image in this directory. The Labels directory has two subdirectories, train and Val. Our dataset is shown in various cases in Fig. 3.

The 333 photos from the dataset were increased to 1092 images utilizing Roboflow to enhance our model. A maximum of three enhanced versions of each image were created by randomly applying blur (up to 1 px), noise (up to 5% px), brightness (between -40% and +40%), exposure (between -35% and +35%), rotation hue (between -50° and +50°), and exposure (between -35% and +35%).

The dataset related to validation consists of three sets, including one main dataset and two other sets obtained by applying preprocessing and augmentations. In one of our preprocessing suites: resizing (stretched to 416x640), automatic contrast adjustment (using adaptive equalization), grayscale (applied) and increments: brightness (between -40% and +) 40%, exposure (between -21% and +21%) and in another set of pre-processing: resizing (stretched to 640x480), automatic contrast adjustment (using adaptive equalization), and gain: brightness (between -40% and +40%), exposure (between -21% and +21%) , 90° rotate (clockwise, counterclockwise) we used. Examples of augmented images are shown in Fig. 3. Lastly, the labeled pictures are divided into a training (70%) and validation set (30%).

C. Google Colab

We used Google Colab, which provides free access to powerful GPUs. All training and testing tasks are performed using a 12GB NVIDIA Tesla T4 GPU. Our model was trained for 20 epochs with batch size of 16, image size of 640 and with YOLOv5 default adjustment for other hyper parameters. Fig. 4 shows the Google Collab's details for our model implementation.

D. Training and Testing

It is usually a good idea to start with a model that has already been trained on very big datasets and then use its weights to train an object detector. Even if the trained weights do not have the items needed for this experiment, this is OK. Transfer learning is the name for this process. In order to speed up network learning, beginning weights from a pre-trained model are utilized, which comprises weights from the COCO

dataset. Additionally advantageous is the fact that fewer data will be needed. [18].

Our total dataset consists of 1425 images, 75% of which are used for training, 25% for validation. 75% of training includes 1092 images, 25% of validation includes 333 images.

IV. Experimental Results and Performance Analysis

In this section, we introduce the experimental results and model performance analysis, and show the training outcomes with using pretraining weights and compare the three models of YOLO.

A. Experimental Results

We trained our model with YOLOv5n, YOLOv5s, YOLOv6 which are different versions of YOLO model, YOLOv5s performed better. Some examples of model predictions for new and unseen images are shown in Fig. 5, and label 0 for falling, label 1 for walking, and label 2 for sitting.

B. Model Evaluation

The gathered experimental data were compared in this work using various assessment criteria, including accuracy, recall, and mean average precision (mAP). The true positive rate, or TPR, is a statistic used to assess the likelihood that items from the real world would be correctly identified. When a model produces no false negatives, which indicates that there are no bounding boxes that are not recognized but ought to be detected, it has a high recall. Eq. (1) provides the mathematical form for the recall.

$$R = \frac{TP}{TP+FN} = \frac{TP}{\text{Total Ground Truths}} \quad (1)$$

The real positive and false negative are denoted in the equation above by the letters TP and TN, respectively. Eq. (2) defines *precision* as the percentage of correctly anticipated positives, often known as the positive predictive value. The accurate model creates no false positives (FP) and only detects important things.

$$P = \frac{TP}{TP+FP} = \frac{TP}{\text{Total Predictions}} \quad (2)$$

According to Eq. (3), AP is the area under the PR curve, and mAP is the average of all AP values across various classes and categories (3).

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \quad (3)$$

where n is the number of classes [22].

We YOLOv5n, YOLOv5s, and we taught YOLOv6s. In Fig. 6 and 7, we have shown the confusion matrix, F1 confidence and Precision-Recall curves, respectively. In addition, we have included the parameters of each model for better evaluation.

Table I show the performance results for different Yolo based models. These models are YOLOv5n, YOLOv5s and YOLOv6s. As shown in the Table, YOLOv5s presents better results compared to other models.

YOLOv5s model obtained better results.

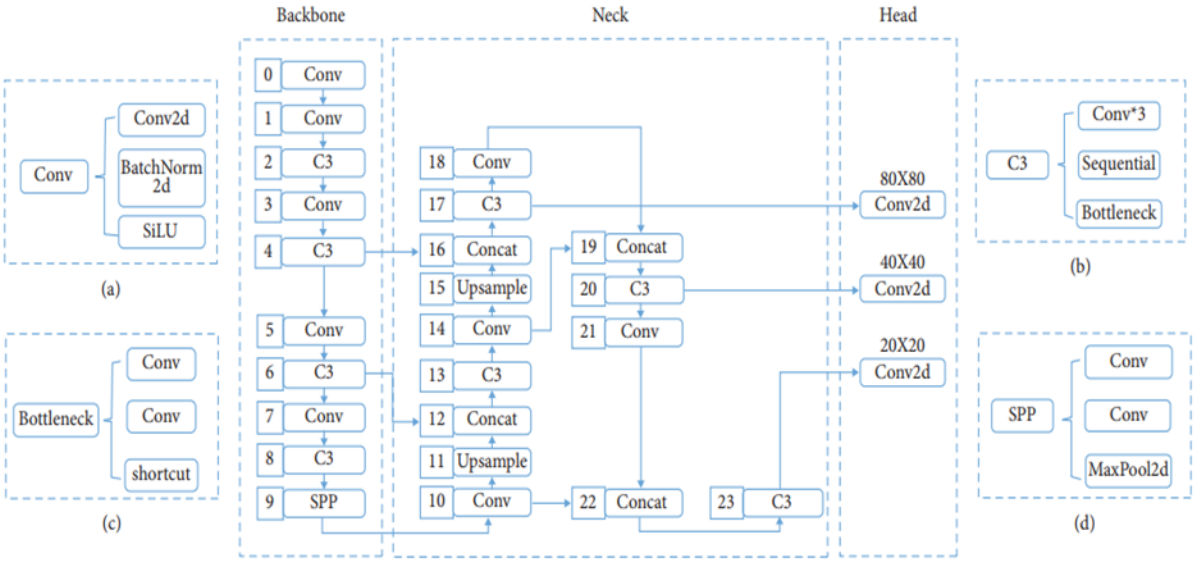


Fig. 1. Yolov5 architecture [20].

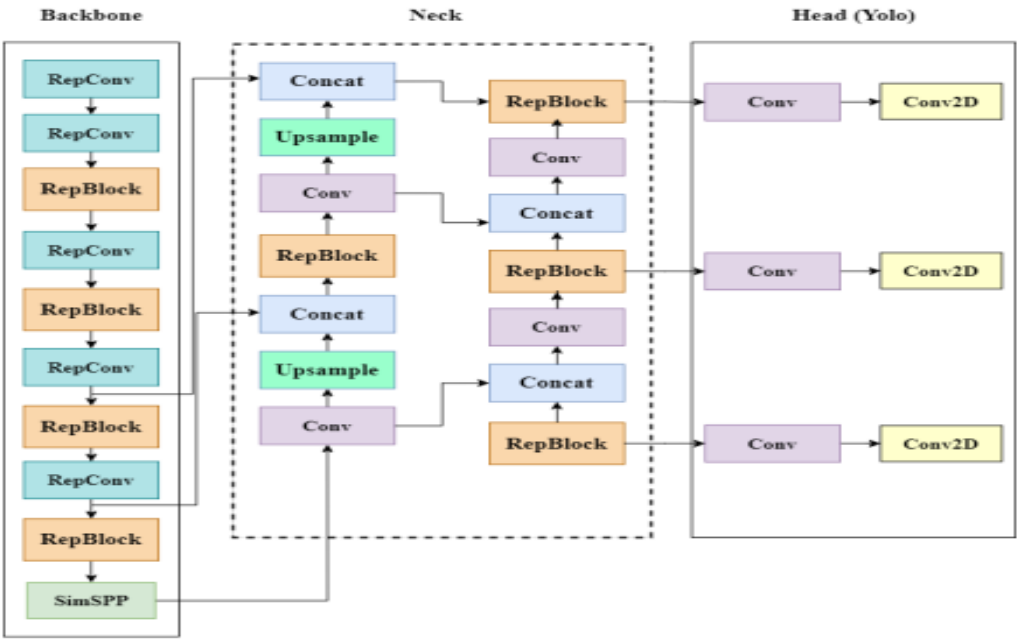


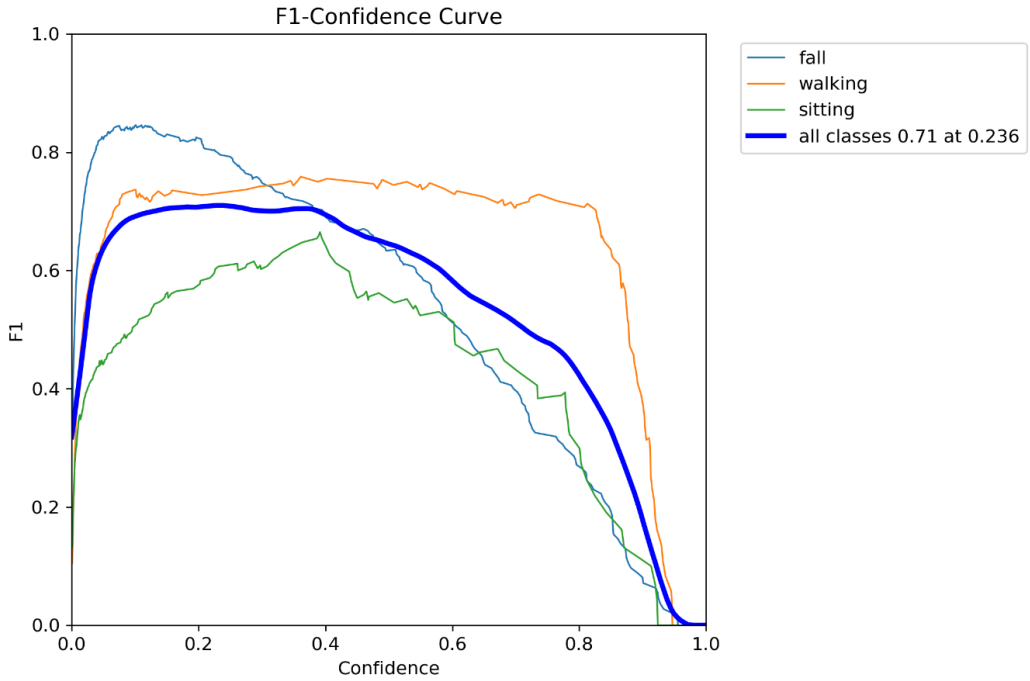
Fig. 2. Network structure of YOLOv6 [21].

Fig. 3. Sample images from dataset.

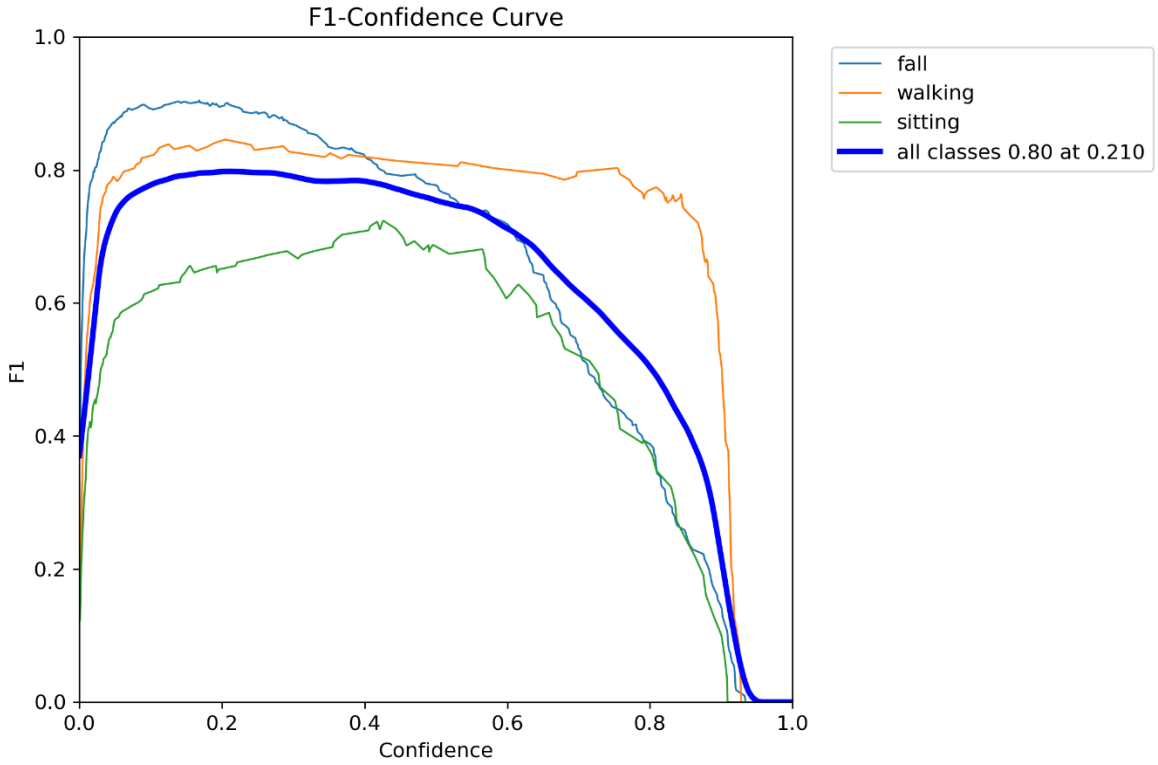
```
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 460.32.03    Driver Version: 460.32.03    CUDA Version: 11.2     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  Tesla T4            Off  | 00000000:00:04.0 Off |                    0 |
| N/A   63C    P8    11W /  70W |      0MiB / 15109MiB |      0%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+
```

Fig. 4. Details of google colab' GPU.

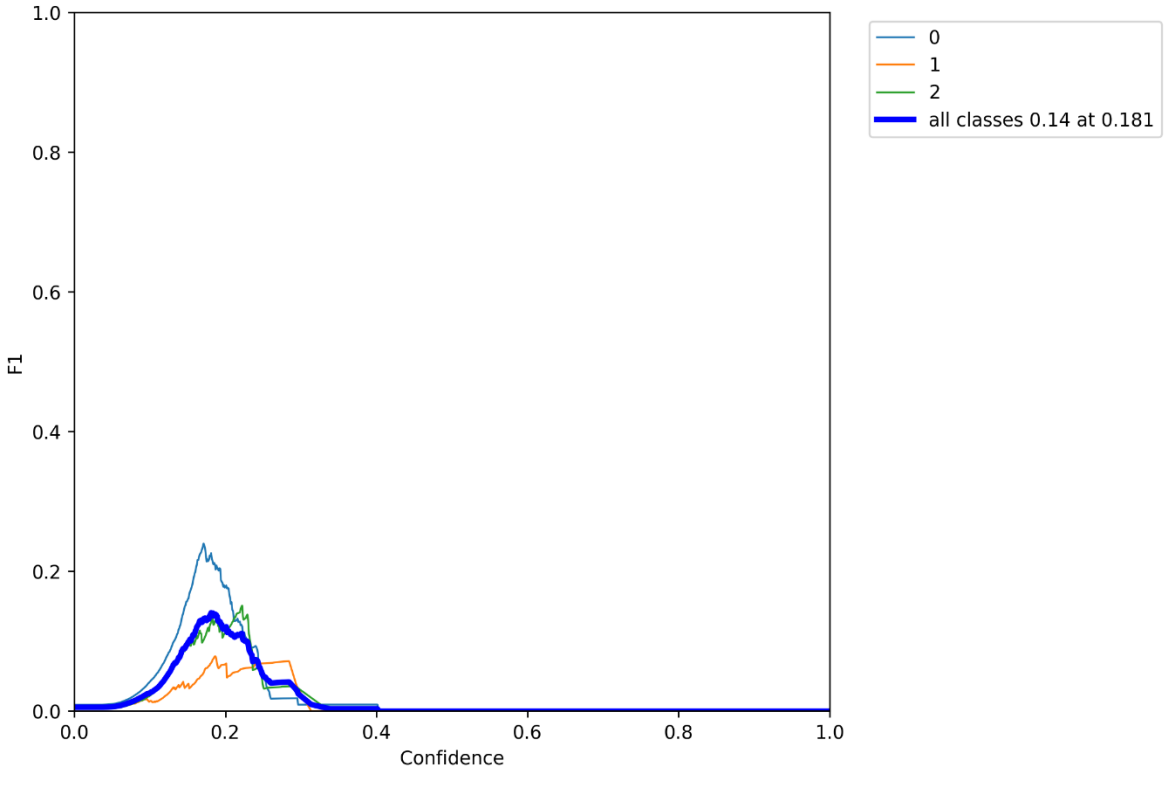
Fig. 5. Experimental results.



Yolov5n

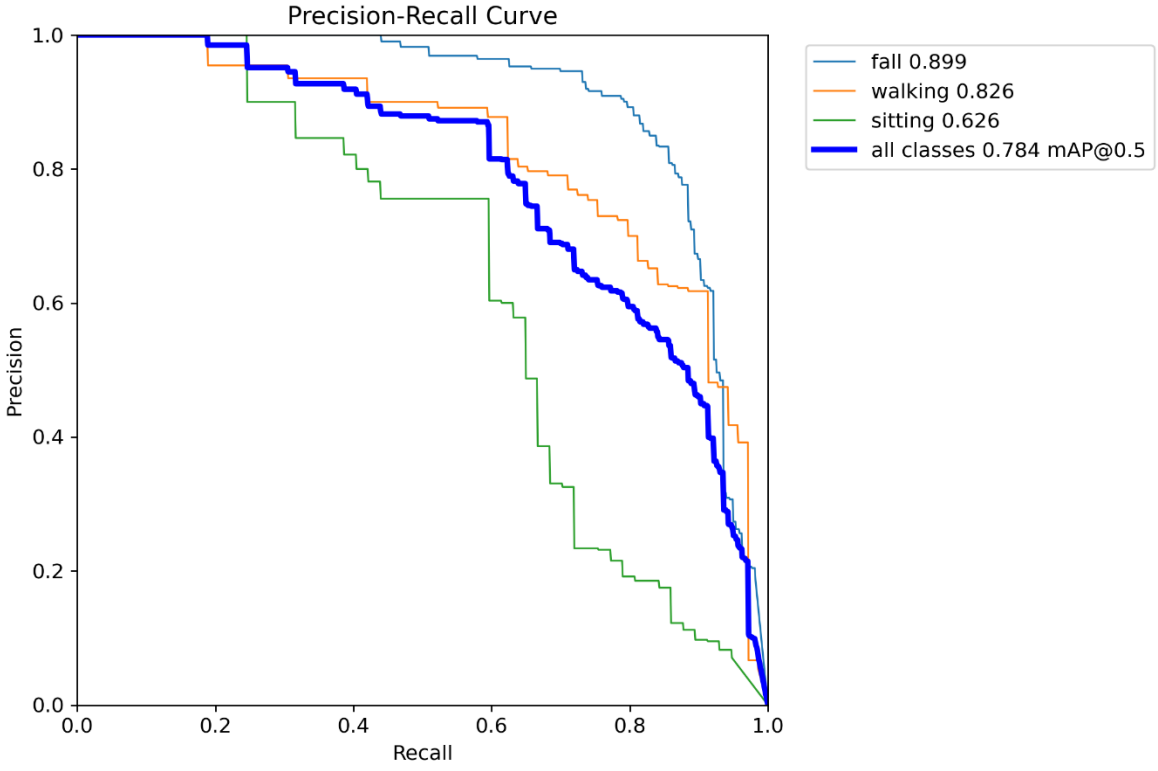


Yolov5s

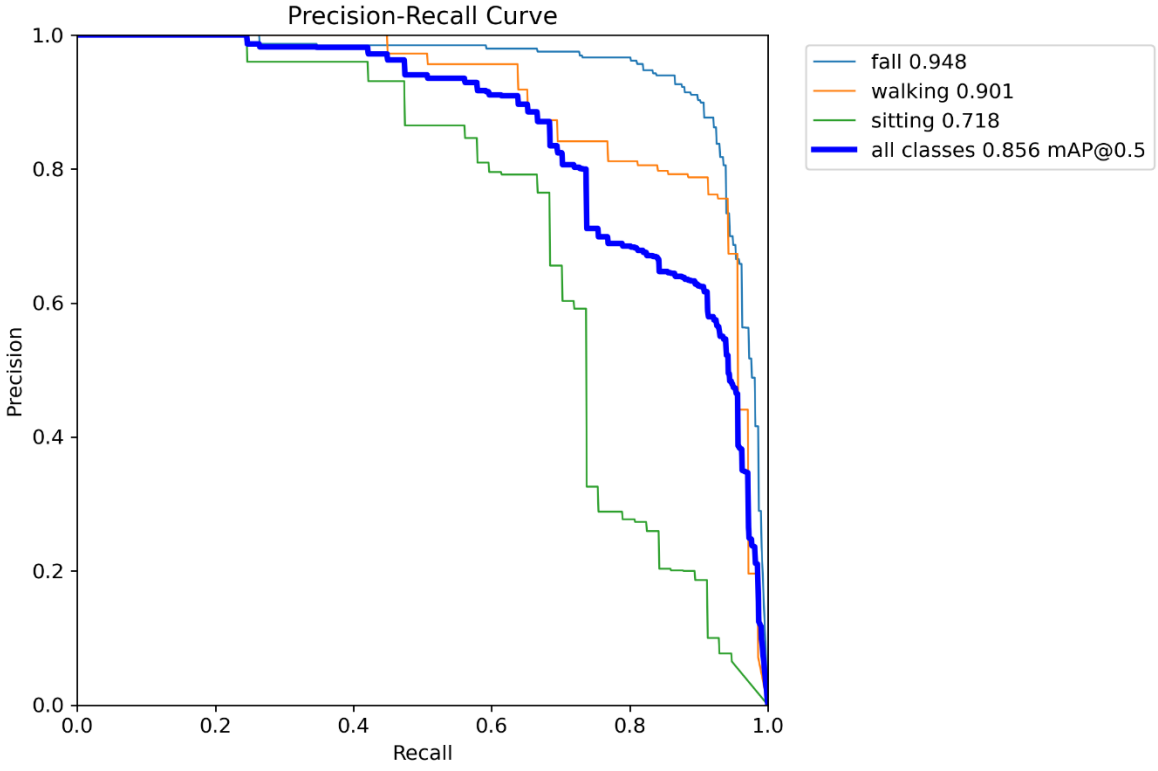


Yolov6s

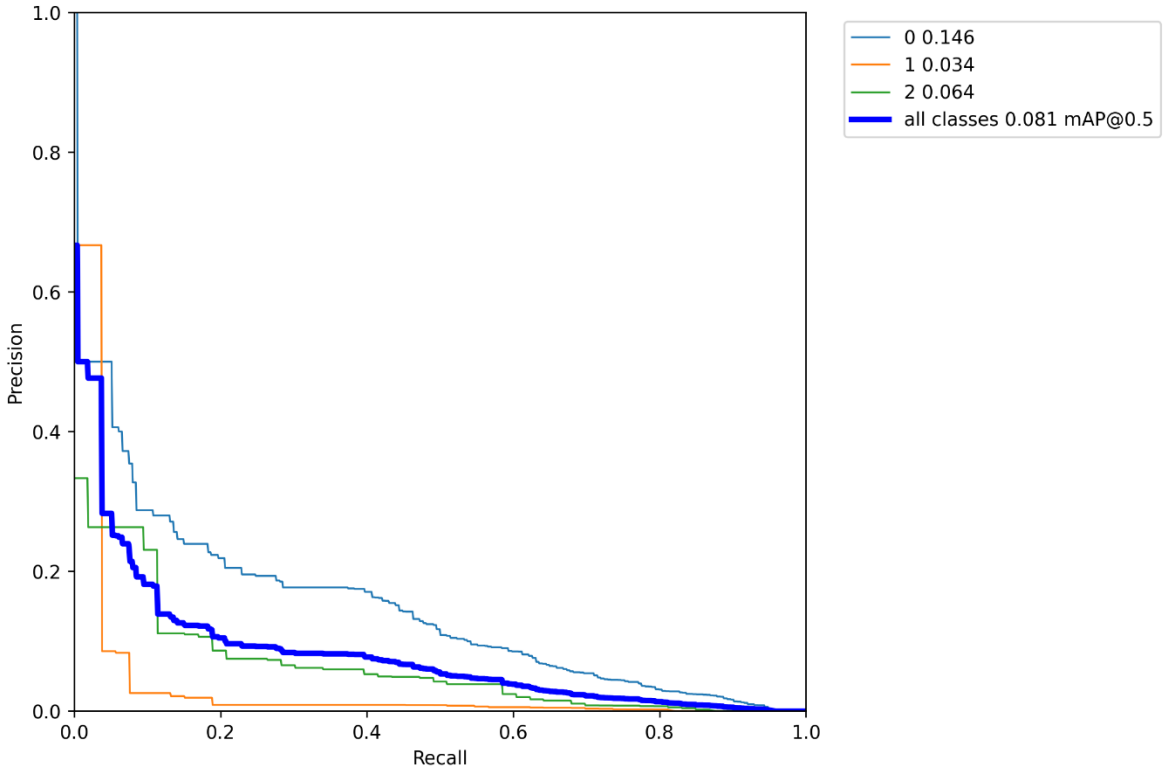
Fig. 6. F1-Confidence curves.



Yolov5n



Yolov5s



Yolov6

Fig. 7. Precision-recall curves.

TABLE I. RESULT OF MAP, RECALL, PRECISION

Model	*MAP*@0.5	*MAP*@0.5: 0.95	*Precision*	*Recall*
YOLOv5n	0.845	0.45	0.844	0.7
YOLOv5s	0.948	0.588	0.938	0.84
YOLOv6s	0.741	0.215	0.95	0.884

V. PERFORMANCE COMPARISON

This section presents performance comparison of different human fall detection methods. These methods involve OpenPose + LSTM [23], OpenPose + CNN [24], OpenPose + three thresholds [25] and the proposed method. In order to conduct fair comparison, these methods are experimented in current dataset and evaluated the performance using the procedure stated in [26]. Table II shows the performance comparison for existing methods.

TABLE II. PERFORMANCE COMPARISON BETWEEN METHODS

Model	*OpenPose + LSTM*	*OpenPose + CNN*	*OpenPose + three thresholds*	*our proposed method*
Accuracy	0.936	0.917	0.924	0.948

As shown in Table II, our proposed method presents better results compared to other existing methods in terms of the accuracy rate.

VI. CONCLUSION

This study investigated the advantages of data augmentation for the human posture dataset, and a dataset consisting of 1092 images for training and 374 images for testing was created from three classes, i.e. falling, walking and sitting. The performance of different YOLOv5 and YOLOv6 kinds was compared in the second stage based on accuracy, recall, and mAP. Based on the results, YOLOv5s outperformed other fall detection algorithms and had the greatest mAP @ 0.5, 0.948 and mAP @ 0.5: 0.95, which was 0.588. Finally, the proposed method is compared to other existing methods to demonstrate the outperformed method. The proposed method in this study is sensitive to the camera field of view for elderly fall detection that can be effectively influenced on accuracy rate in real applications. For future works, may employ other sensors combine with vision-based sensors to improve accuracy rate detection.

REFERENCES

[1] Alqahtani, E., et al., Smart homes and families to enable sustainable societies: A data-driven approach for multi-perspective parameter discovery using bert modelling. Sustainability, 2022. 14(20), pp. 13534.

[2] Zolfaghari, S., E. Khodabandehloo, and D. Riboni, TraMiner: Vision-based analysis of locomotion traces for cognitive assessment in smart-homes. Cognitive Computation, 2022. 14(5) , pp. 1549-1570.

[3] Forbes, G., S. Massie, and S. Craw, Fall prediction using behavioural modelling from sensor data in smart homes. Artificial Intelligence Review, 2020. 53(2) , pp. 1071-1091.

[4] Quan, W., et al., Human posture recognition for estimation of human body condition. Journal of Advanced Computational Intelligence and Intelligent Informatics, 2019. 23(3), pp. 519-527.

[5] Yu, N. and J. Lv, Human body posture recognition algorithm for still images. The Journal of Engineering, 2020. 2020(13) , pp. 322-325.

[6] Mohamed, M., A. El-Kilany, and N. El-Tazi, Future Activities Prediction Framework in Smart Homes Environment. IEEE Access, 2022. 10: p. 85154-85169.

[7] Dang, L.M., et al., Sensor-based and vision-based human activity recognition: A comprehensive survey. Pattern Recognition, 2020. 108: p. 107561.

[8] Santos, G.L., et al., Accelerometer-based human fall detection using convolutional neural networks. Sensors, 2019. 19(7) , pp. 1644.

[9] Jokanović, B. and M. Amin, Fall detection using deep learning in range-Doppler radars. IEEE Transactions on Aerospace and Electronic Systems, 2017. 54(1) , pp. 180-189.

[10] Kamel, A., et al., Deep convolutional neural networks for human action recognition using depth maps and postures. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2018. 49(9) , pp. 1806-1819.

[11] Khan, A., et al., A survey of the recent architectures of deep convolutional neural networks. Artificial intelligence review, 2020. 53(8) , pp. 5455-5516.

[12] Chen, T., Z. Ding, and B. Li, Elderly Fall Detection Based on Improved YOLOv5s Network. IEEE Access, 2022. 10, pp. 91273-91282.

[13] Ajerla, D., S. Mahfuz, and F. Zulkernine, A real-time patient monitoring framework for fall detection. Wireless Communications and Mobile Computing, 2019. 2019.

[14] Queralta, J.P., et al. Edge-AI in LoRa-based health monitoring: Fall detection system with fog computing and LSTM recurrent neural networks. in 2019 42nd international conference on telecommunications and signal processing (TSP). 2019. IEEE.

[15] Wang, X. and K. Jia. Human fall detection algorithm based on YOLOv3. in 2020 IEEE 5th International Conference on Image, Vision and Computing (ICIVC). 2020. IEEE.

[16] Waheed, M., H. Afzal, and K. Mehmood, NT-FDS—A Noise Tolerant Fall Detection System Using Deep Learning on Wearable Devices. Sensors, 2021. 21(6) , pp. 2006.

[17] Salimi, M., J.J. Machado, and J.M.R. Tavares, Using deep neural networks for human fall detection based on pose estimation. Sensors, 2022. 22(12): p. 4544.

[18] Hatab, M., H. Malekmohamadi, and A. Amira. Surface defect detection using YOLO network. in Proceedings of SAI Intelligent Systems Conference. 2020. Springer.

[19] Zhou, F., H. Zhao, and Z. Nie. Safety helmet detection based on YOLOv5. in 2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA). 2021. IEEE.

[20] Dai, G., L. Hu, and J. Fan, DA-ActNN-YOLOV5: Hybrid YOLO v5 Model with Data Augmentation and Activation of Compression Mechanism for Potato Disease Identification. Computational Intelligence and Neuroscience, 2022. 2022.

[21] Aburaed, N., et al. A Study on the Autonomous Detection of Impact Craters. in IAPR Workshop on Artificial Neural Networks in Pattern Recognition. 2023. Springer.

[22] Ali, L., et al., Development of YOLOv5-Based Real-Time Smart Monitoring System for Increasing Lab Safety Awareness in Educational Institutions. Sensors, 2022. 22(22) , pp. 8820.

[23] Jeong, S., Kang, S. and Chun, I., 2019, June. 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) (pp. 1-4). IEEE.

[24] Xu, Q., Huang, G., Yu, M. and Guo, Y., 2020. Fall prediction based on key points of human bones. Physica A: Statistical Mechanics and its Applications, 540, p.123205.

[25] Chen, W., Jiang, Z., Guo, H. and Ni, X., 2020. Fall detection based on key points of human-skeleton using openpose. Symmetry, 12(5), p.744.

[26] Ali, M.A., Hussain, A.J. and Sadiq, A.T., 2022. Human Fall Down Recognition Using Coordinates Key Points Skeleton. International journal of online and biomedical engineering, 18(2), pp.88-104.