Dean&Francis

ISSN 2959-6157

A Novel Fall Detection Scheme Using Yolo-Pose with Efficient Attention and Lightweight Convolution

Jianqi Wang

International Curriculum Centre,
High School Affiliated to the
Renmin University of China;
Beijing;100080; China.

ericwangjianqi@hotmail.com

Abstract:

Falling is a prominent external cause of severe injuries and death among the elderly. To this end, fall detection serves to mitigate these hazards. This paper outlines the need for fall detection systems and surveys the proposed methodologies, citing several disadvantages and advantages. In these respects, we propose a fall detection system based on a threshold classification approach and the improved YOLO-pose model, the latter involving attention mechanism-related enhancements and increased convolutions to enhance the accuracy and speed of pose estimation. This paper wants to evaluate this proposed addition to the original one. The paper tests the proposed system under multiple scenarios to demonstrate efficacy in practical applications within elder care environments. Finally, we evaluate the performance of our model and go through plans regarding this fall-detection system.

Keywords: Fall Detection; Pose Estimation; Elderly; YOLO-pose.

1 Introduction

The proportion of older people (over 60) in the global population is rising rapidly. According to estimates by the World Health Organization, the number of older people worldwide will reach 1 billion in 2019 and is expected to increase by 400 million by 2023 and rise to 1 billion by 2025[1]. In addition, the proportion of the elderly population will almost double from 12% in 2015 to 22% in 2050[2]. Statistics show that the number of older adults is growing rapidly and gradually becoming an essential group in today's society.

Death is the most frequently discussed topic among the many challenges older people face. According to a report released by the Centers for Disease Control and Prevention in the United States, among the top 10 causes of death in the elderly, the only non-disease-related death is due to accidents[3]. Falls are the second leading cause of death among all accidents, with an estimated 684,000 deaths due to falls each year, of which 37.3 million falls are severe enough to require medical intervention[4]. The problem is particularly acute for older people, who account for the highest percentage of deaths from falls. In addition, studies have shown that more than 400,000 older adults die from falls every year[5]. Even without death, falls can cause significant damage to older people. The National Center for Biotechnology Information states that common injuries in falls include fractures as well as "hematoma, joint dislocation, severe lacerations, sprains, and other disabling soft tissue injuries."[6]

However, research shows that timely detection and treatment of elderly people who have fallen can effectively reduce 80% of the risk of mortality and 26% of the risk of long-term treatment[7]. Therefore, this article will suggest a method of detecting the falling of the elderly using deep learning to reduce the harm of falling on the elderly. We introduced a vision-based fall detection method using an improved YOLO-pose structure for pose estimation and a threshold-based classification method. We aim to apply the system to nursing homes since nursing homes are where the elderly gather. Since there are many elderly people in nursing homes, fall detection methods based on wearable sensors are not effective. Therefore, we selected the vision-based method based on its ability to simultaneously detect falls for multiple people. One main difficulty in applying a vision-based fall detection system is to ensure that elderly people's privacy is not infringed. However, the proposed methods for privacy protection in fall detection systems depend on special cameras to get in-depth images or track the movement of people using them. Those methods are expensive for non-profit nursing homes. Therefore, we are using the technique of mosaicing the face of the person in the picture, which is more efficient. Section 2 describes some of the existing fall detection techniques, and Section 3 is dedicated to our proposed method. The result of the experiment is presented in section 4. Section 5 explores and analyzes the proposed fall detection system's limitations and application. At last, we will draw some conclusions in Section 6.

2 Related Works

2.1 Silhouette-Based Method

Adrián Núñez-Marcos et al. proposed a method of fall detection developed via the use of CNN for feature extraction and FC-NN for classification algorithm[8]. Their approach further simplified the system›s independence from environmental factors, lowering the complexity of hand-crafted image processing. Liu Chien-Liang et al. proposed a fall detection method based on human silhouette detection[9]. This technique monitored the movement of objects appearing in the background of video footage, extracted the silhouette of the human, and classified it as falling, standing, or temporarily resting using KNN classification. This technique shows completive effectiveness only when human postures are seen to change. A system detecting fall behaviour using depth videos is proposed by Sase Priyanka S and Smriti H. Bhandari[10]. This system utilized a combination of background subtraction, "filtration, binarization, and connected component analysis" to obtain the ROI. While this system was promising for classification, it could not detect lying down. Rougier Caroline et al. proposed a fall detection method by storing and analyzing the shape of the human body's motion history to identify falling[11]. This process detected unusual falls experienced in day-to-day operations while respecting the subject's privacy. The method of Foroughi Homa et al. also applied the shape motion feature; however, it could classify various types of falls, such as forward or sideways [12].

2.2 Key-point-based method

They suggested a fall detection system based on skeletal data and ST-GCN[13], which acquires the skeletal graph of human bodies with a Kinect v2 camera and uses the ST-GCN algorithm to classify falling. The author also combined transfer learning into this so that it could efficiently deal with the lack of data. Wang, Chien Yao, et al. proposed a fall detection technique using improved YOLO-v7[14]. This technique is based on the body detection by YOLO-v7 in image or video and the comparison of shoulder height difference between consecutive frames against a defined threshold [15]. The issue with this technique is that since it is threshold-based, fall detection may only be accurate sometimes. Maji, Debapriya, et al.[16] used the improved YOLO model. The authors employ a pose estimation algorithm to enhance the model, helping it predict messages for multiple subjects. Chen et al. presented a feature extraction method using GlobalNet and RefineNet. This method is more accurate than the other feature extraction methods. They offered a fall detection system using OpenPose for feature extraction and MobileNetv2 for the classifier[17]. They modified MobileNetv2 with pooling and key points figures to gain higher accuracy. The accuracy of this method is 98.6% on the Le2i dataset and 99.75% on the UR fall dataset.

3 Methodology

The basic algorithm of the fall detection system proposed by this paper is shown in Figure 1. In this system, the images collected by cameras will be inputs, and the improved YOLO-pose model will read the image and analyze the key points of human bodies in the images. After that, the classification algorithm will classify the human in the images as falling down or not, and if people fall in the images, the alarm will be triggered and output warning information.

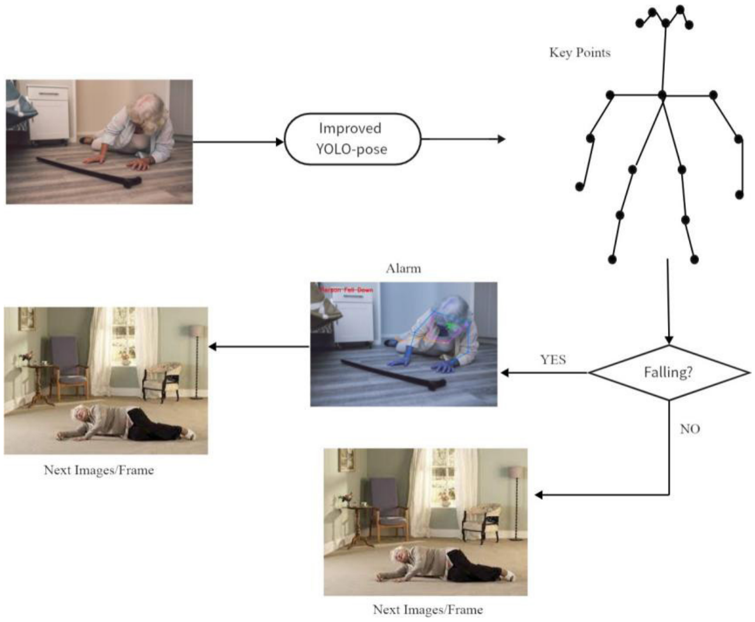


Fig. 1 Fall detection system workflow

3.1 Data Analysis

We have trained and validated the upgraded YOLO-pose pose estimation model using COCO 2017. MS Common Object in Context (COCO) has become a central asset towards enriching subsequent research into object identification, segmentation, and captioning[18]. COCO, developed by Microsoft, is annotated with specific object categories and their corresponding vital points. It has approximately 164k images: training, validation, and testing. The training set has 118k images, while validation and test have 5k and 41k images, respectively. We use the training set to train and test the model with the validation set. We are also using the Fall Detection Dataset to assess the effectiveness of our model in a fall detection system[19]. This dataset has 374 images as training data and 111 images as validation data. The images in this dataset are gathered from several daily real-life scenarios and are divided into fall and not-fallen.

3.2 Data Processing

Here, we need to say summer by summer. We don't count animals or object-centred photos, even though the COCO dataset for pose detection would assist in determining whether someone has just fallen. Thus, we select 2k for the value set from 5k images; 50k could be chosen for training out of about 180k in total.

3.3 Improving YOLO-pose

3.3.1 YOLO-pose Versions

In this article, we will compare two YOLO-pose versions, v7 and v8, which are the latest two YOLO-pose versions. Since the YOLOv7-pose is the older version, many models are derived from the YOLOv7-pose. We are using one of those derived models named YOLOv7-w6-pose. This model lightweights the original model and makes it more efficient. For YOLOv8-pose, we are using YOLOv8x-pose-p6, the highest accuracy on the official release report of YOLOv8-pose offered by Ultralytics. We will use the version that performs better on pose estimation.

3.3.2 Attention Mechanisms

In this article, we incorporate attention mechanisms into the YOLO-pose model to enhance its emphasis on human subjects. The first attention mechanism we introduce is the Convolutional Block Attention Module (CBAM), which consists of a Channel Attention Module and a Spatial Attention Module[20]. The Channel Attention Module identifies the important elements in the input images, while the Spatial Attention Module determines the locations of these significant features. By combining these two modules, CBAM enhances YOLO-pose's ability to focus on humans.

Additionally, we are integrating Efficient Channel Attention (ECA) and Coordinate Attention (CA) into the YOLO-pose framework. These three attention mechanisms have different advantages. CBAM adopts a dual attention mechanism, which enables CBAM to capture features more comprehensively when processing images, improving the model's expressive power and accuracy. ECA has the advantages of no dimensionality reduction and fewer parameters, making it suitable for application scenarios that require performance improvement without increasing computational complexity. CA is sensitive to direction perception and position, has flexibility, and has a larger range of information capture. Adopting these three types of attention mechanisms can improve model accuracy while exploring which type of attention mechanism YOLO pose is more suitable for making similar changes and improvements to the newly proposed model. The architecture of those three attention mechanisms is shown in Figure 2, Figure 3, and Figure 4.

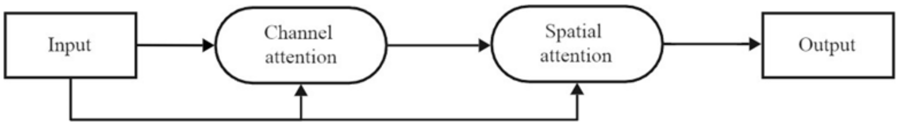


Fig. 2 CBAM Structure

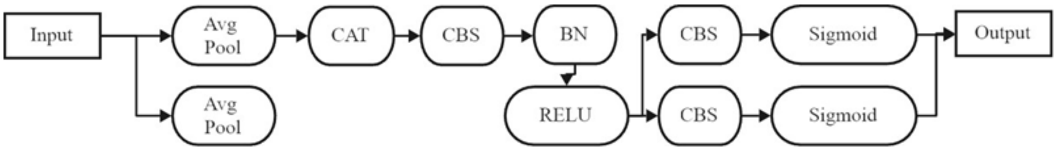


Fig. 3 CA Structure

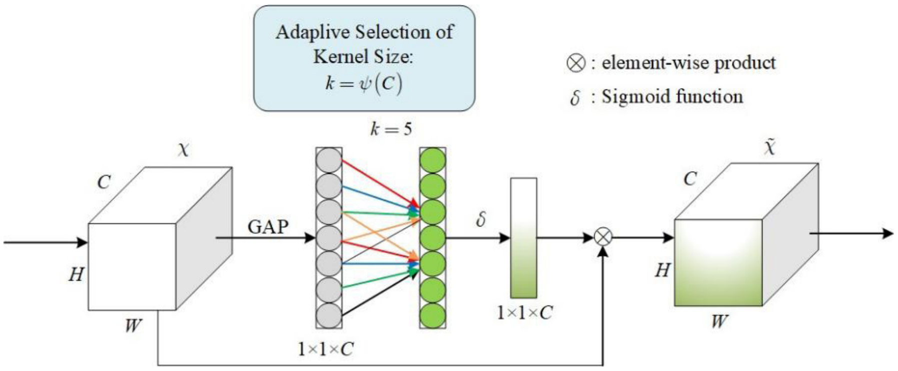


Fig. 4 ECA Structure

3.3.3 Convolutions

The article involves replacing the proposed model's convolution with three other different types of convolution, characterized as the Deformable Convolution Module(DC), the Depthwise Separable Convolution Module(DSC), and the Ghost Convolution Module(GC). Such convolutions are intelligent enough to compress multiple layers' work into one layer, according to Zhao, Dewei, et al., so that the model becomes lightweight[21]. Moreover, the Depthwise Separable Convolution Module leads to fewer model parameters, producing a less complex and more effective model. In addition, according to Angshuman Thakuria and Chyngyz Erkinbaev, the Ghost Convolution Module can generate nearly the same feature map with fewer filters compared to the original convolution used in the YOLO model and make the model more efficient. By applying those convolutions, we aim to improve the detection and processing speed of the model.

$$y(p_0)=\sum_{pn\in R} w(p_n)\bullet x(p_0+p_n+?p_n) \tag{1}$$

Equation 1 shows the mathematical equation form of the Deformable Convolution. *p0* is the location of the centre of the input image, and *pn* is the relative location between the convolution kernel and *P0*. Δ*pn* is the offset between the position of points in the convolution kernel and the position in the original images.

$$y_i{}^c=\sum_{K_h=1}^{K_h}\sum_{K_w=1}^{K_w} x^c_{i+K_h-?\frac{K_h}{2}?,\,j+k_h-?\frac{K_w}{2}?}\bullet w_i{}^{c,K_h,K_w} \tag{2}$$

$$y_{i,j}{}^{c'}=\sum_{c=1}^{C} w^c_{c'}\bullet\sum_{k_h=1}^{K_h}\sum_{k_w=1}^{K_w} x^c_{i+k_h-?\frac{K_h}{2}?,\,j+k_w-?\frac{K_w}{2}?} \tag{3}$$

The Depthwise Separable Convolution consists of two components: Depthwise Convolution and Pointwise Convolution. The mathematical expressions for Depthwise Convolution and Pointwise Convolution are presented in Equations 2 and 3, respectively. The variables *i* and *j* denote the position within the feature map. K_h and K_w are the height and width of the convolution kernel. C represents the number of channels in the input feature map. The variables *w*, *x*, and *y* correspond to the weights, input, and output at specific positions.

$$yij=\Phi i,j(yi')+Conv\ (X) \tag{4}$$

Equation 4 is the mathematics equation for the Ghost Convolution where $\Phi_{i,j}$ is a linear operation for generating a ghost feature map, and y_{ij} is the *j*th ghost feature map generated from the *i*th feature map.

3.4 Model structure and techniques

3.4.1 YOLOv7-w6-pose

Figure 5 shows the structure of YOLOv7-w6-pose. The structure can be divided into two main parts: backbones and backbone and head. The main function of the Backbone is to do feature extraction and set up the structure for the whole model. The Head analyzer analyzes the outputs of the Backbone, and based on those outputs, it can perform final classification and detection. The YOLOv7-w6-pose's Backbone is formed with ReOrg, CBA, and ELAN, while CBS, SPPCPSC, Upsample, CAT, E-ELAN, Detect, and Keypoint form the Head.

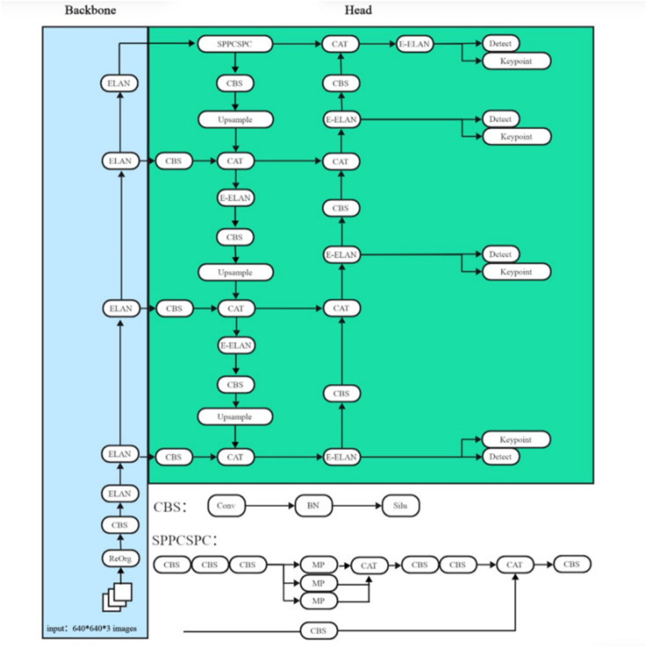


Fig. 5 YOLOv7-w6-pose Structure

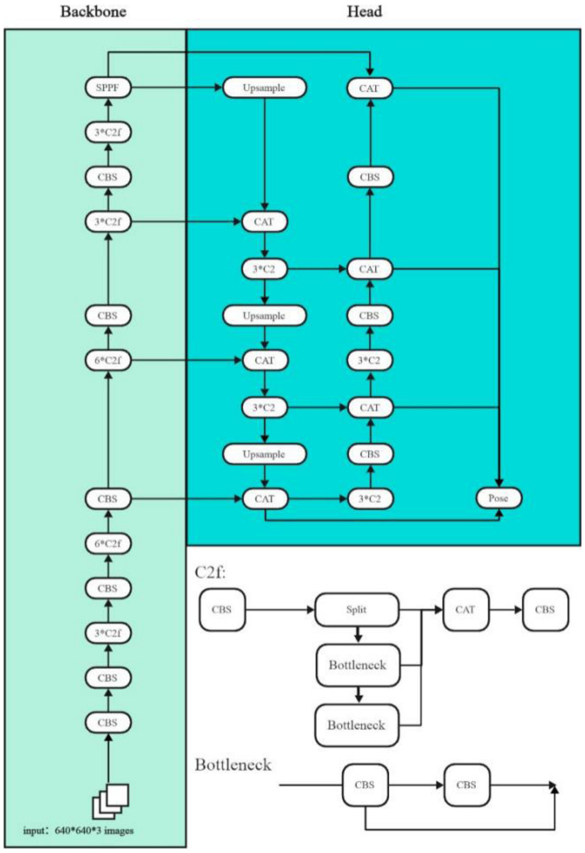


Fig. 6 YOLOV8-pose-p6 Structure

3.4.2 YOLOv8-pose-p6

Figure 6 shows the structure of YOLOv8-pose-p6. The Backbone and Head in YOLOv8-pose- p6 serve similar functions as in YOLOv7-w6-pose but leverage different modules. Unlike in YOLOv7-w6-pose, in this model, the Backbone is formed by CBS, C2f, and SPPF, and Upsample, CAT, C2, and Pose from the Head.

4 Experiment

4.1 Metrics

For the evaluation of the model, we have three main metrics, namely Precision (P), Recall (R) and mean average precision (mAP). Precision is the ratio of true positives among true positives and false positives, showing the model's ability to avoid false positives and provide reliable results. The recall is the ratio of true positives among true positives and false negatives, evaluating the model's capability to recognize all objects of interest. Precision and Recall can be calculated using specific equations (Equations 5 and 6). The mAP can assess the model's detection ability under different scenarios, and a high mAP score suggests the model can work effectively in various situations. Two types of mAP are used in this article: mAP_{50} and $mAP_{50:95}$. mAP_{50} focuses more on easy conditions, while $mAP_{50:95}$ evaluates the model's ability under different difficulty levels. Additionally, the time used during testing is considered to evaluate the addition of convolutions since the aim of changing convolutions is to enhance the model's detection speed.

$$P= \frac{\text{TruePositive}}{\text{TruePositive+FalsePositive}} \tag{5}$$

$$P= \frac{\text{TruePositive}}{\text{TruePositive+FalseNegative}} \tag{6}$$

4.2 Models with single-edition

4.2.1 Two YOLO-pose versions

YOLOv7-pose and YOLOv8-pose are the two latest YOLO versions that achieve pose estimation. Therefore, we are comparing those two versions to get a better version as our baseline model.

According to the official Ultralytics report on the release of YOLOv8-pose, the YOLOv8x-pose-p6 has a mAP_{50} of 0.921 and a $mAP_{50:95}$ of 0.716 in the COCO dataset[22]. However, the test results of YOLOv7-w6-pose on the COCO dataset are 0.94 for mAP50 and 0.74 for mAP50:95. The data shows that YOLOv7-w6-pose performs better than YOLOv8x-pose-p6, so we are going to use YOLOv7-w6-pose as our baseline model for the following experiments. The baseline model performance is shown in Table 1.

Table 1 Baseline Model Performance

Model	P	R	mAP_{50}	$mAP_{50:95}$	Time
YOLOv7-w6-pose	0.877	0.882	0.94	0.74	2minutes 26seconds

4.2.2 Models with attention mechanisms

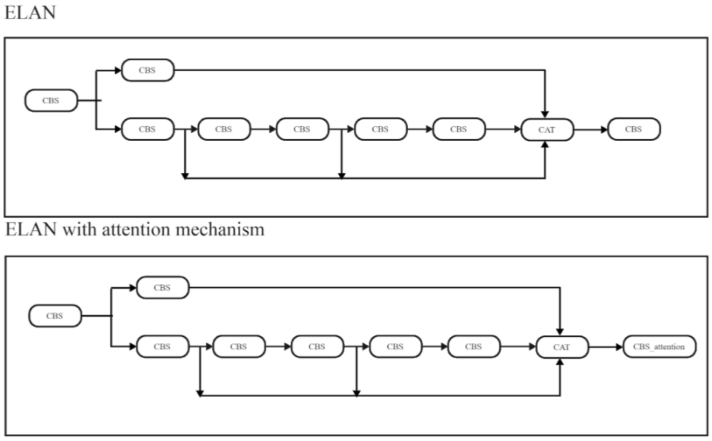


Fig. 7 Comparison Before and After Adding the Attention Mechanism

Figure 7 shows the change before and after we add attention mechanisms. The method we are using to add the attention mechanism is to change the last module used in ELAN from CBS to CBS with the attention mechanism. The main purpose of using this method is that the number of layers of the model will not be changed, and it is less likely to cause errors during training than changing the number of layers. The data for this edition is shown in Table 2.

Table 2 Performance of Models With Different Attention Mechanisms

Model	P	R	mAP_{50}	$mAP_{50:95}$
+CBAM	0.866	0.849	0.921	0.706
+ECA	0.905	0.836	0.938	0.736
+CA	0.871	0.879	0.936	0.735

In Table 2, the data marked as red and blue are the best and second-best performances for each metric. Based on the data we get, the attention mechanism that performs best on pose estimation is ECA since it has the highest precision. As we mentioned in the previous section, the main purpose of adding an attention mechanism is to enhance the model's performance.

4.2.3 Model with convolutions

The main change difference after the change of convolutions is that four CBS modules in ELAN and E-ELAN are changed into other convolutions mentioned in the previous part of this article. Figure 8 and Figure 9 compare the structure of ELAN and E-ELAN before and after the edition.

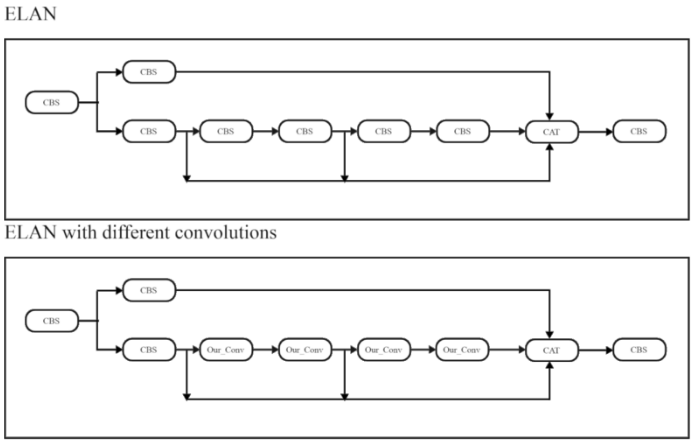


Fig. 8 Comparison Before and After Changing Convolutions in ELAN

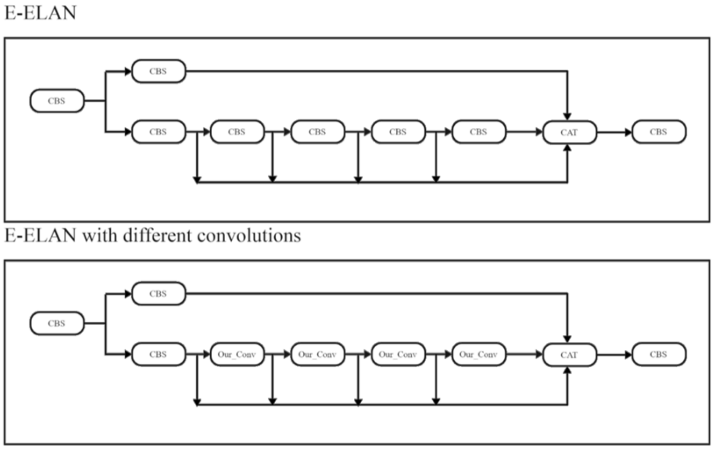


Fig. 9 Comparison Before and After Changing Convolutions in E-ELAN

Table 3 presents each convolution's performance, with the best performance highlighted in red and the second best highlighted in blue.

Table 3 Performances of Models with Different Convolutions

Model	P	R	mAP_{50}	Time
+GC	0.836	0.759	0.84	2minutes 48seconds
+DC	0.357	0.334	0.275	2minutes 55seconds
+DSC	0.881	0.857	0.93	1minute 41seconds

According to the data in Table 3, the best version among the three convolutions we selected is the Depthwise Separable convolution, which has the highest data for all four metrics. In addition, the Depthwise Separable Convolution model has higher accuracy than the baseline model. Most importantly, the new model has also significantly increased the speed of the model. That accomplishes our main purpose of editing convolutions. Also, the data in the table indicates that the model with Deformable Convolution has the worst performance and a significant gap compared to the other two models. Even though Deformable Convolution can make the work done in multiple layers in the original convolution in one layer, the architecture of Deformable Convolution is much more complex than the original convolution, Depthwise Separable Convolution, and the Ghost Convolution. It may result in low accuracy and lower speed.

4.3 Models with combined edition

From the results we get with the single edition, the optimized combination is YOLOv7-w6-pose, ECA, and depthwise separable convolution. In this section, we will test our last model by combining those three factors to prove that our edition of the model is effective. The performance of our final model is shown in Table 4.

Table 4 Final Model Performance

Model	P	R	mAP_{50}	$MAP_{50:95}$	Time
Base	0.877	0.882	0.94	0.74	2minutes 26seconds
Our Model	0.886	0.842	0.925	0.685	1minutes 17 seconds

Based on the data in Table 4, our model has improved the accuracy of the baseline model. The most significant improvement we made was on the speed of the model. It will make our model work effectively in the whole fall detection system. However, our model still needs some improvement in recalling the correct answers and facing more general conditions.

We also assess our model's performance in comparison to other YOLO models. Research by Mohd et al. evaluated various versions of YOLOv5, YOLOv6, and YOLOv7, revealing that the highest mAP_{50} achieved by these models is 0.783, which is lower than our model's performance[23]. Additionally, an analysis covering all iterations of the YOLO series, from the original YOLO to YOLOv8, found that both mAP_{50} and $mAP_{50:95}$ scores for these models were also below those of our model[24]. This evaluation utilized our dataset, indicating that our modified model outperforms many other YOLO series models. That further supports the effectiveness of our modifications to the model.

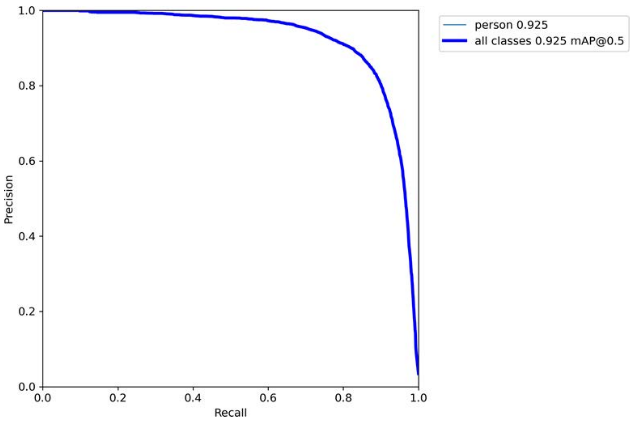
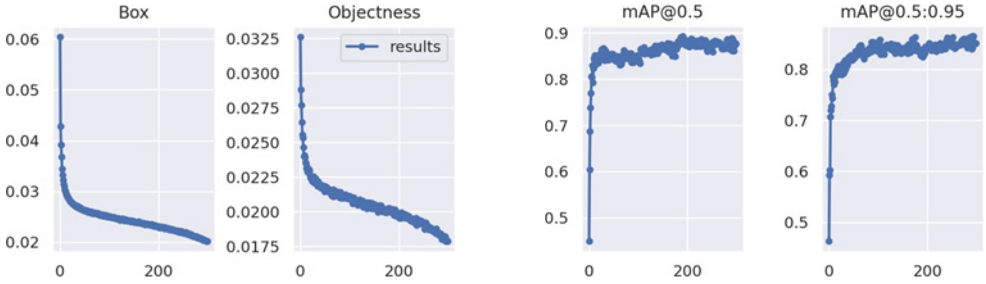


Fig. 10 PR Curve



(a) Loss (b) mAP

Fig. 11 The Changes in Loss and Accuracy

Figure 10 and Figure 11 show the changes in accuracy and loss of our model during the training. It indicates that the training process is smooth and correct. The curve for loss in Figure 11a keeps going down for both the boxes surrounding human bodies and the objectness. In addition, both curves for mAP_{50} and $mAP_{50:95}$ in Figure 11b are increasing during the training. Figure 10 can also further prove that the data we show in Table 4 is the best result of the training since the mAP_{50} corresponds to the mAP_{50} shown on the curve.

4.4 Results with threshold-based classification

4.4.1 Algorithm

The classification method used in this paper is based on the key point coordinates identified by the previously proposed model. If the difference in coordinates between the shoulders and feet is too small, it will be recognized as a fall. This method primarily considers that the height of the torso coordinates significantly decreases during a fall. Additionally, to avoid misidentifying a sitting position as a fall, we use the relative height of the feet and shoulders, making the alerts given by this fall detection system more accurate.

4.4.2 Results

Table 5 Fall Detection System Performance

Model	True Positive	False Positive	True Negative	False Negative
Base	204[1]	41	166	74
Our Model	218	28	179	60

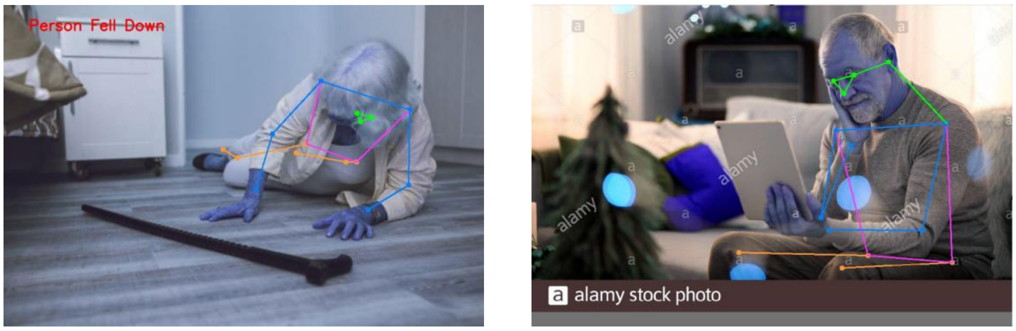
Table 5 presents the performance of our fall detection system. From the data in the table above, we can determine the values of our system's P and R.

$$PrecisionOur = \frac{218}{218+28} = 0.886 \tag{7}$$

$$RecallOur = \frac{218}{218+60} = 0.78 \tag{8}$$

$$PrecisionBase = \frac{204}{204+41} = 0.832 \tag{9}$$

$$RecallBase = \frac{204}{204+74} = 0.733 \tag{10}$$



(a) Images With Falling

(b) Images Without Falling

Fig. 12 Processed Images by the Fall Detection System

Figure 12 shows the image processed by the system using our model. The data and Figure 12a show that the fall detection system using our model can accurately identify and provide warnings for fall behaviour. Meanwhile, Figure 12b demonstrates that the system does not mistakenly recognize daily behaviours such as sitting posture as falls. Therefore, our system effectively detects falls, and applying our model to this system is feasible and successful. By comparing the results of the baseline model and our model, we can conclude that our model has three advantages.

To begin with, our model has better accuracy since the P and R of the fall detection system leverage our model, which is higher than the precision and recall of the system with the baseline model. Furthermore, our system is much faster than the system with the baseline model. The average time our system takes to classify one image is 4 seconds. The baseline model systems spend 10 seconds on average to classify one image. In addition, the baseline model system's classifying speed is decreasing. The time the baseline model system spends to classify the 20th image is 20 seconds, and our system has a constant speed. We tested the two systems on devices with the same storage spaces. Our fall detection system can classify all the images quickly, while the baseline model system needs larger storage, so we cannot get the overall baseline model's precision and recall rate. The higher speed and lighter weight prove that our edition makes the model more applicable since, as mentioned in the previous sections of this article, on-time detection will reduce the probability of death of the elderly and long-term harm.

5 Conclusion

We plan to apply the fall detection system discussed in this article to the bedrooms of nursing homes. Our system can detect falls for multiple individuals simultaneously and operates at a fast detection speed, making it highly beneficial in an environment like a nursing home. Furthermore, our system has implemented corresponding measures. In addition, we found camera devices in public places through visits to nursing homes. It makes using our fall detection system in nursing homes cost-effective while proving that elderly people are receptive to recording devices that do not involve privacy. Therefore, our system is both implementable and effective in nursing home bedrooms.

This method may incorrectly identify everyday actions such as sitting or lying down as falls. Additionally, trying out various classification methods could optimize the model's overall performance. This paper is more concerned with fall detection for older adults. In the future, we will seek more relevant datasets focused on falls among the elderly or collect the necessary data ourselves. In addition, the fall detection data set we are using is not big. In addition, the technology of mosaic on the face mentioned in the article to protect the privacy of the elderly has not been specifically implemented. We will integrate this technology into the fall detection system, as elaborated in this article, to improve the entire system in the future. Despite a thorough examination and discussion concerning falls among older adults, falls turn out to be a much more persistent, unavoidable social malaise requiring attention. They are very dominant concerning their onslaught on older adults and their heavy consequences. However, the risk can be eliminated by timely detection.

References

[1] WHO. Ageing, 2021a. URL https://www.who.int/health-topics/ageing#tab= tab_1.

[2] WHO. Ageing and health, 2022. URL https://www.who.int/news-room/ fact-sheets/detail/ageing-and-health.

[3] Yelena G, Hoyert D, Lentzner H, et al. Two Trends in Health and Aging Trends in Causes of Death among Older Persons in the United States. 2006.

[4] WHO. Falls, Apr 2021b. URL https://www.who.int/news-room/fact-sheets/ detail/falls.

[5] Lei Ren. Research and implementation of fall detection method based on smartphones. Master's thesis, Zhejiang University, 2019.

[6] NIH, 2015. URL https://www.ncbi.nlm.nih.gov/books/NBK235613/.

[7] Takamasa Iio, Masahiro Shiomi, Koji Kamei, et al. Social acceptance by senior citizens and caregivers of a fall detection system using range sensors in a nursing home. Advanced Robotics, 30(3):190–205, Feb 2016.

[8] Adrián, Núñez-Marcos,, Gorka Azkune, et al. Vision-based fall detection with convolutional neural networks. Wireless Communications and Mobile Computing, 2017: 1–16, 2017.

[9] Chien-Liang Liu, Chia-Hoang Lee, Ping-Min Lin. A fall detection system using the k-nearest neighbour classifier. Expert Systems with Applications, 37(10):7174–7181, Oct 2010.

[10] Priyanka S Sase and Smriti H Bhandari. Human fall detection using depth videos. IEEE Access, 16(12), Feb 2018.

[11] Caroline Rougier, Jean Meunier, Alain St-Arnaud, et al. Fall detection from human shape and motion history using video surveillance. 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07), 5(6), 2007.

[12] Homa Foroughi, Baharak Shakeri Aski, Hamidreza Pourreza. Intelligent video surveillance for monitoring fall detection of elderly in home environments, Dec 2008.

[13] Oussema Keskes, Rita Noumeir. Vision-based fall detection using st-gcn. IEEE Access, 9: 28224–28236, 2021.

[14] Chien-Yao Wang, Alexey Bochkovskiy, Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies set new state-of-the-art for real-time object detectors. arXiv: 2207.02696 [CS], Jul 2022.

[15] Debapriya Maji, Soyeb Nagori, Manu Mathew, et al. Yolo-pose: Enhancing yolo for multi-person pose estimation using object keypoint similarity loss. arXiv:2204.06806 [cs], Apr 2022.

[16] Yilun Chen, Zhicheng Wang, Yuxiang Peng, et al. Cascaded pyramid network for multi-person pose estimation, Apr 2018.

[17] Mengqi Gao, Jiangjiao Li, Dazheng Zhou, et al. Fall detection is based on openpose and mobilenetv2 network. IET image processing, 17(3):722–732, Oct 2022.

[18] Microsoft.Coco dataset dataset, 2024.

[19] Uttej Kandagatla. Fall detection dataset, 2021.

[20] Sanghyun Woo, Jongchan Park, Joon-Young Lee, et al. Cbam: Convolutional block attention module. Computer Vision – ECCV, pp. 3–19, 2018.

[21] Dewei Zhao, Faming Shao, Li Yang, et al. Object detection based on an improved yolov7 model for unmanned aerial-vehicle patrol tasks in controlled areas. Electronics, 12(23):4887, Jan 2023.

[22] Ultralytics. Pose, 2023. URL https://docs.ultralytics.com/tasks/pose/#models.

[23] Izzaty Mohd, Ali Sophian, Hasan Zaki, et al. Assessing the performance of yolov5, yolov6, and yolov7 in road defect detection and classification: a comparative study. Bulletin of Electrical Engineering and Informatics, 13(1):350–360, 2024.

[24] Muhammad Hussain. Yolo-v1 to yolo-v8, the rise of yolo and its complementary nature toward digital manufacturing and industrial defect detection. Machines, 11(7):677, Jul 2023.