AIFDS - AI based Fall Detection System using YOLOv5-s and DeepSparse

Joel D'silva
Dept. of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi Mumbai, India
joeldsilva4901@gmail.com

Aryan Koul
Dept. of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi Mumbai, India
aryankoul290301@gmail.com

Madhur Thakkar
Dept. of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi Mumbai, India
madhurthakkar247@gmail.com

Prof. Archana Shirke
Dept. of Information Technology
Fr. C. Rodrigues Institute of Technology
Navi Mumbai, India
archana.shirke@fcrit.ac.in

***Abstract*—A fall in this study is defined as an event in which a person suddenly and inadvertently collapses from an upright position and the person's legs can no longer support oneself. These incidents of falling can have serious impact such as injury or in the worst-case scenario can cause even death if they do not get help immediately. Aid can be given more quickly if the occurrence of falls can be immediately detected. We propose a fall detection algorithm using YOLOv5-s. The people in the image are treated as individual objects by the algorithm and bounding box is drawn around them with high accuracy and speed. The proposed technique measures the length and breadth of the bounding box around the person. Fall is detected when the width of the box exceeds the height by 5%. In order to improve the performance of YOLOv5-s model the system proposes a CPU runtime engine which provides GPU-class performance on CPU using Neuralmagic's DeepSparse.**

***Index Terms*—YOLO, fall detection, camera, DeepSparse, Artificial Intelligence**

I. Introduction

A fall is defined as an event which results in a person coming to rest inadvertently on the ground or floor or other lower level. Worldwide people are facing injuries caused by falling and sometimes even people are dying due to a fall. According to WHO's survey, falls are a leading cause of unintentional injuries in adults older than 65 years old, with 37.3 million falls requiring medical attention and 646,000 resulting in deaths annually. This is especially common among senior citizens living alone[1]. The current birth rate is lower than the death rate which implies that the population is ageing and hence the fatalities caused due to falls are also increasing. In such a dire situation the creation and improvement of fall detection and prevention technologies is the need of the hour. Currently the fall detection technologies can be broadly divided into three classes :

- Wearable devices
- Environmental sensors
- Image detectors

The wearable devices use technologies like accelerometers, gyroscopes, etc to detect falls. A major problem which hinders their use is the changes in temperature, humidity and frequent recharging. Along with these hardware requirements they also cost a lot as they need continuous maintenance and subscription fees. Sometimes the wearer may not be comfortable using them.

Environmental sensor technology uses various types of sensors to detect the sounds of falls, sudden changes in pressure, heat signatures and ultrasonic sounds. The major issue with these technologies are differentiating animals and objects from humans and detection of falls in crowded places.

Image detectors use machine learning to detect changes in posture. But the issue faced while using this technology is the computational charge and the requirement of high quality data-sets for training. Our project aims to reduce the resources used in image detectors by integrating DeepSparse [2] engine alongwith YOLOv5-s [3] model with the system's Fall Detection algorithm to accurately detect falls using a single camera.

II. Literature Survey

A. Human fall detection

[4] In this project they are applying an algorithm for fall detection consists of video acquisition from an external source such as a surveillance camera, or a saved video or clip. Firstly, a video is acquired as an input. The video is divided into frames, usually being 30 frames per second. There were marks placed on the head of the human in the video. A frame could be regarded as a single image. The human in the image is identified using various methods. Finally the features are extracted and the occurrence of fall is detected on the basis of patterns in these features. The fall is detected on the basis of these features:

- Foreground Extraction
- Centroid Acceleration
- Centroid

- Head Position
- Number of Ones
- Head speed
- Motion Vector
- Centre Speed
- Aspect Ratio
- Fall Angle

B. Fall Detection Systems for Elderly Care: A Survey

[5] In this paper, they have proposed a fall detection system based on machine learning. Their system detects falls by classifying different activities into fall and non-fall actions and alert the registered user in case of emergency. They have used SisFall dataset with variety of activities of multiple participants calculate features. Machine learning algorithms SVM and decision tree are used to detect the falls on the basis of calculated features. Their system was able to acquire an accuracy up to 96% by using decision tree algorithm.

C. IoT Based Fall Detection Monitoring and Alarm System For Elderly

[6] This paper presents an IoT based fall detection monitoring and alarm system for the elderly using 3-axis Accelerometer. In the proposed system for detection of falling, the elderly patient's acceleration data are continuously acquired by using a wearable sensor and stored on a cloud server, using an IoT board. To access the stored data, an android application is designed for the medical expert to examine the fall in the patient and provide the needed assistance, if needed. A threshold-based approach for the fall detection has been used to get the sensor data and set the threshold on accelerometer readings. A complete algorithm has been designed for the detection of genuine fall.

D. Automatic Body Fall Detection System for Elderly People using Accelerometer and Vision Based Technique

[7] In this project generally wearable sensor and vision based technique are used that is automatically detecting body fall as early as possible. Accelerometer is used for measuring or maintaining orientation and angular velocity. In vision based procedure first procure casings or video arrangements from the camera is done. The division module separates the body outline from the foundation. For Feature Extraction GLCM method is used. SVM method is used for classification. By using these methods human body fall can surely be detected and preventive measures can be taken.

E. Falling Detection System Based on Machine Learning

[8] In this proposed system, a dataset of videos containing falling actions has been utilized via dividing each video into many shots that are consequently being converted into gray-level images. Then, for detecting the moving objects in videos, the foreground is firstly detected, then noise and shadow are deleted to detect the moving object. Finally, a number of features, including aspect ratio and falling angle, are extracted and a number of classifiers are being applied in order to detect the occurrence of falling. Experimental results, using 10-fold cross validation, shown that the proposed falling detection approach based on Linear Discriminant Analysis (LDA) classification algorithm has outperformed both support vector machines (SVMs) and Knearest neighbor (KNN) classification algorithms via achieving falling detection with accuracy of 96.59 %.

III. Proposed System

The main purpose of this system is to detect human fall and report to the user via sending an Email to the registered Email ID. The user can interact with the system via a web-based application developed using Python-Flask.

The Fig 1 depicts the flow of the project and displays the modules and processes included. When the website is loaded the fall detection section is displayed. The fall detection module captures the frames from the camera and sends those frames in First-In-First-Out batches (FIFO) to the flask server.

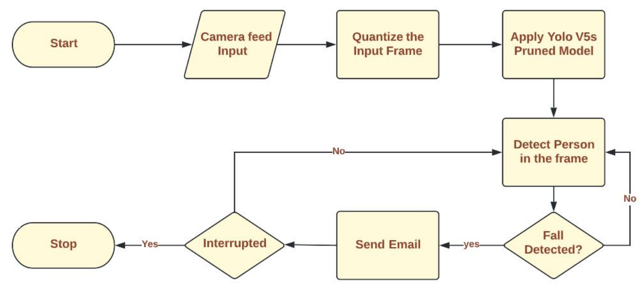

Fig. 1. Flow Chart

In Figure above we have briefly described the overall architecture of fall detection system . A fall detection system receives an input in the form of video which contains any object. The object can be a person or anything. When our model detects a person using Yolo V5s Pruned Model, it starts applying the Fall Detection Algorithm on it. If the fall is detected it sends an alert message to the respective authorities via email. And, if the fall is not detected it keeps on taking the camera feed as an input and applying the respective algorithms.

- **Camera Feed Input**: Input video for the system might be acquire by webcam or any other camera. That camera footage is known as Camera Feed Input.
- **Quantize the Input Frame**: It is the process of converting the video frame into contiguous numpy array. It helps in consuming less storage and increasing the resource optimization. It also ends up being more efficient as an input to the Yolo Model.
- **Yolo V5s Pruned Model**: YOLO is a state-of-the-art, real-time object detection system. The algorithm applies a single neural network to the full frame. The main advantage of YOLO is that it looks at the whole frame at test time and makes predictions with a single network evaluation which makes it extremely fast. It helps us in detecting whether the object in the frame is a person or anything else.

- **Fall Detection**: For Fall Detection we are using a purely hit and trial based model. In which we have set the measurements of the bounding box of the person that if width of the bounding box of the person is 5 percent i.e. 1.05 times greater than the height of the bounding box of the person, then fall is detected or else keep the fall detecting.
- **Alert**: When the fall is detected send an alert mail to the logged in mail id along with the frames of the fallen person.
- **Stop Camera Feed**: If the user wants to stop giving the input and stop the detection. He can simply do it by just clicking on the button or else he can simply keep it going and going.
- **Logout**: When the client is done using our application he/she can logout from our system and stop the camera live feed.

IV. Results and Discussion

TABLE I
COMPARISON OF DIFFERENT DEEPSPARSE YOLOV5 BASED MODELS

Model Name	*Description*	*FPS*	*Confidence*
yolov5s-pruned_quant	Sparse INT8 quantized YOLOv5s model that recovers 94% of its baseline mAP	35 - 45	80 - 87%
yolov5s-pruned	Sparse YOLOv5l model trained with full FP32 precision that recovers 96% of its baseline mAP	27 - 35	85 - 90%
yolov5s-base	Dense full precision YOLOv5s model	17 - 25	85 - 95%
yolov5l-pruned_quant	Sparse INT8 quantized YOLOv5l model that recovers 95% of its baseline mAP	30 – 35	80 - 90%
yolov5l-pruned	Sparse YOLOv5l model trained with full FP32 precision that recovers 98% of its baseline mAP	25 – 30	85 - 90%
yolov5l-base	Dense full precision YOLOv5l model	20 - 25	87 - 97%

Six different YOLOv5 models were tested consisting of three variants of YOLOv5-s models and three variants of YOLOv5-l models. The pruned and quant YOLOv5 models are sparsified models and they accepts quantized input images, due to which they performed the best with and average of 30-40 fps (frames per second) with only a minor effect on confidence compared to other models/variants. The YOLOv5-baseline models performed the worst in terms of average fps but gave the best confidence compared to other models. The YOLOv5-pruned models performed moderately in terms of both average fps and confidence. It was observed that YOLOv5-s models had better fps than YOLOv5-l models in all cases. Since, the main focus of this paper is to get the best performance on CPU the yolov5s-pruned_quant model was selected for inference as it performed superior to yolov5l-pruned_quant model. Hence, from the Table I we can infer that Pruning of the model and Quantization of inputs have a big impact on performance, especially on CPU.

TABLE II
YOLOV5 MODEL SPARSIFICATION AND VALIDATION RESULTS

Model Type	*Sparsity*	*mAP@50*	*File Size(MB)*
YOLOv5l Base	0%	65.4	147.3
YOLOv5l Pruned	86.3%	64.3	30.7
YOLOv5l Pruned Quantized	79.2%	62.3	11.7
YOLOv5s Base	0%	55.6	23.7
YOLOv5s Pruned	75.6%	53.4	7.8
YOLOv5s Pruned Quantized	68.2%	52.5	3.1

The measure of sparsity, mean average precision (mAP) and file size in MBs of six different models based on YOLOv5 is given in Table II. It can be understood from the table that on pruning and sparsification of yolov5 models their file sizes reduce drastically. This helps in saving memory bandwidth of the machine and hence saving overall computation power and cost, while providing two-fold performance.

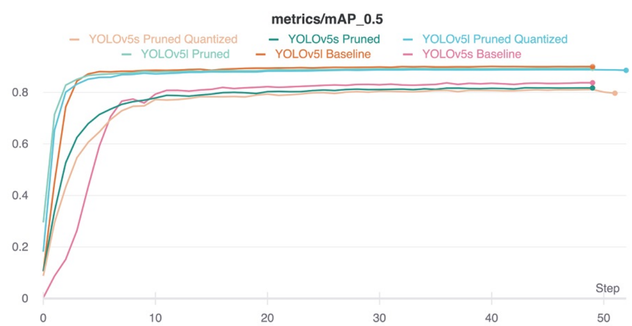

Fig. 2. Graphs of different deepsparse yolov5 models

V. Conclusion

Efficient methods to decrease the risks of people falling are needed to be devised. In the past several attempts have been made to carry out this task using conventional wearable technology or highly demanding computer vision methods. In order to gain high performance without compromising much on accuracy, an approach with resource optimization in mind was explored by AIFDS. The YOLOv5-s model based on Neuralmagic's Deepsparse proposed in this paper has successfully gained much higher performance on CPU with minimal affect on accuracy in the task of Fall Detection as compared to base YOLOv5 models . The system is able to achieve an accuracy of about 80%. The application detects the falls and notify the concerned people via email for immediate help. Hence, the dangers of such falls can be reduced as necessary help can be provided on time.

Acknowledgment

We would like to express our gratitude towards everyone who helped and supported us in making of this project.

We thank Dr. S. M. Khot, Principal of Fr. C Rodrigues Institute of Technology and Prof. Vaishali Bodade, Head of the Information Technology Department for giving us the opportunity to work on this project. We also thank our guide Prof. Archana Shirke for guiding us in this endeavor.

REFERENCES

[1] World Health Organisation, “Falls”, https://www.who.int/en/news-room/fact-sheets/detail/falls, 2018.

[2] Kurtz, Mark and Kopinsky, Justin and Gelashvili, Rati and Matveev, Alexander and Carr, John and Goin, Michael and Leiserson, William and Moore, Sage and Nell, Bill and Shavit, Nir and Alistarh, Dan, ”Inducing and Exploiting Activation Sparsity for Fast Inference on Deep Neural Networks”, 2020 Proceedings of the 37th International Conference on Machine Learning

[3] Y. Yin, L. Lei, M. Liang, X. Li, Y. He and L. Qin, ”Research on Fall Detection Algorithm for the Elderly Living Alone Based on YOLO,” 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT), 2021, pp. 403-408, doi: 10.1109/ICESIT53460.2021.9696459.

[4] S. F. Ali, M. Muaz, A. Fatima, F. Idrees and N. Nazar, ”Human fall detection,” INMIC, 2013, pp. 101-105, doi: 10.1109/INMIC.2013.6731332.

[5] S. Badgujar and A. S. Pillai, ”Fall Detection for Elderly People using Machine Learning,” 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), 2020, pp. 1-4, doi: 10.1109/ICCCNT49239.2020.9225494.

[6] A. Gupta, R. Srivastava, H. Gupta and B. Kumar, ”IoT Based Fall Detection Monitoring and Alarm System For Elderly,” 2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), 2020, pp. 1-5, doi: 10.1109/UPCON50219.2020.9376569.

[7] S. M. Turkane, Swapnil J. Vikhe, C. B. Kadu, P. S. Vikhe, “Automatic Body Fall Detection System for Elderly People using Accelerometer and Vision Based Technique”,vol 8 issue 4 ,2019.

[8] M. Nadi, N. El-Bendary, A. E. Hassanien and T. -h. Kim, ”Falling Detection System Based on Machine Learning,” 2015 4th International Conference on Advanced Information Technology and Sensor Application (AITS), 2015, pp. 71-75, doi: 10.1109/AITS.2015.27. [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].