International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-2, December 2019

Vision based Patient Fall Detection using Deep Learning in Smart Hospitals

Komal Singh, Akshay Rajput, Sachin Sharma

***Abstract:** With the emergence of new concepts like smart hospitals, video surveillance cameras should be introduced in each room of the hospital for the purpose of safety and security. These surveillance cameras can also be used to provide assistance to patients and hospital staff. In particular, a real-time fall of a patient can be detected with the help of these cameras and accordingly, assistance can be provided to them. Different models have already been developed by researchers to detect a human fall using a camera. This paper proposes a vision based deep learning model to detect a human fall. Along with this model, two mathematical based models have also been proposed which uses pre-trained YOLO FCNN and Faster R-CNN architecture to detect the human fall. At the end of this paper, a comparison study has been done on these models to specify which method provides the most accurate results.*

***Keywords:** Deep learning, FCNN, R-CNN, Smart Hospitals.*

I. INTRODUCTION

Human fall can be dangerous as they can cause severe injuries or even lead to death. Compared to healthy people, sick people and elder are more prone to falls because of the weakness caused by their sickness as well as a biological change in their body due to their age [1]. It is seen that patient falls are more common in hospitals. The situation becomes severe for the hospitals since at least one hospital staff should be present near the patient in order to make sure that the patient does not fall. But in reality, it is not possible to have such manpower. Also, with the emergence of new concepts like smart hospitals [2], humans will interact with an increased number of cameras in their daily life. Hence, to assist hospital staff as well as to give patients some alone time, an automatic human fall detection system can be designed with the help of pre-installed cameras to alarm the hospital staff as soon as a fall is detected.

Enormous research has been done by researchers to design an efficient Automatic Human Fall Detection System (AHFDS). All of the existing work can be mainly categorized based on three methods as mentioned in named wearable device based, context-aware based and camera/vision based approaches. In case of wearable device based method, people often forget to carry such devices with them and in the case and if the system is button operated they forget to press the button after fall to generate an alarm. While children often use This button just to create inconvenience for their parents. While context aware based methods use audio, pressure and vibration data generated during a fall which is prone to environmental noise. In comparison with vision-based approaches, they can overcome these drawbacks. Also, surveillance cameras have been installed everywhere in the environment such as hospitals, residence, shelter homes for children and elder, etc. They gather data which has rich information content that can be utilized for fulfilling several functions. Therefore, a reliable vision-based human fall detection system will play an important role in the development of future health care systems. By keeping in mind all these points, we proposed a human fall detection model which uses vision-based method.

In recent years, with the introduction of deep learning (DL) in computer vision, results of many problems such as object detection, object recognition, image classification, and segmentation have been significantly improved. In this paper, our proposed vision-based approach takes advantage of Convolution Neural Network and pre-trained object detection architectures to detect a human fall from standing position.

This work provides three different models. Two of the models are based on mathematical approach applied on two deep learning methods YOLO [3] [Fig. 1] and Faster R-CNN [4] [Fig. 2]. While in the third approach, we provide our own deep learning neural network model to detect a human fall.

The rest of this paper is composed as pursues. Section II provides a summary of fall detection models designed using deep learning approach in vision-based systems. Section III proposes two mathematical models which use predefined deep learning models (YOLO and Faster R-CNN). Section IV provides a deep learning model in order to detect human fall intelligently. Section V compares all the three proposed fall detection algorithms with the algorithms which have already been developed using deep learning. Finally, Section VI concludes the paper.

II. DEEP LEARNING APPROACHES IN FALL DETECTION

Recently, researchers turned their research towards using deep learning techniques for designing an intelligent human fall detection system. In [6], authors used very deep two stream ConvNet [9] and modified VGG16 architecture [5] to detect falls from optical flow of video frames.

Revised Manuscript Received on December 30, 2019.

***** Correspondence Author
Komal Singh*, Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun, Uttarakhand, India. Email: komalsingh.0492@gmail.com
Akshay Rajput, Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun, Uttarakhand, India.
Sachin Sharma, Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun, Uttarakhand, India.

Authors in [2] proposed a deep learning model using long short-term memory (LSTM) neural network. They trained and tested their model on the NTU RGB+D Action Recognition Dataset [10]. Work done by authors in [7] is based on motion history images to capture temporal and spatial features in a video sequence and then passed these features to a depth wise convolutional neural network for classification. They trained their CNN on public Fall Detection Dataset (FDD). [8] used faster R-CNN and modified it to detect fall of human on furniture. They used different features such as centroid of human presence in the scene, human shape aspect ratio, and motion speed. [9] used range-Doppler radar in order to capture time-frequency domain and range domain values which are then passed to neural network for classification.

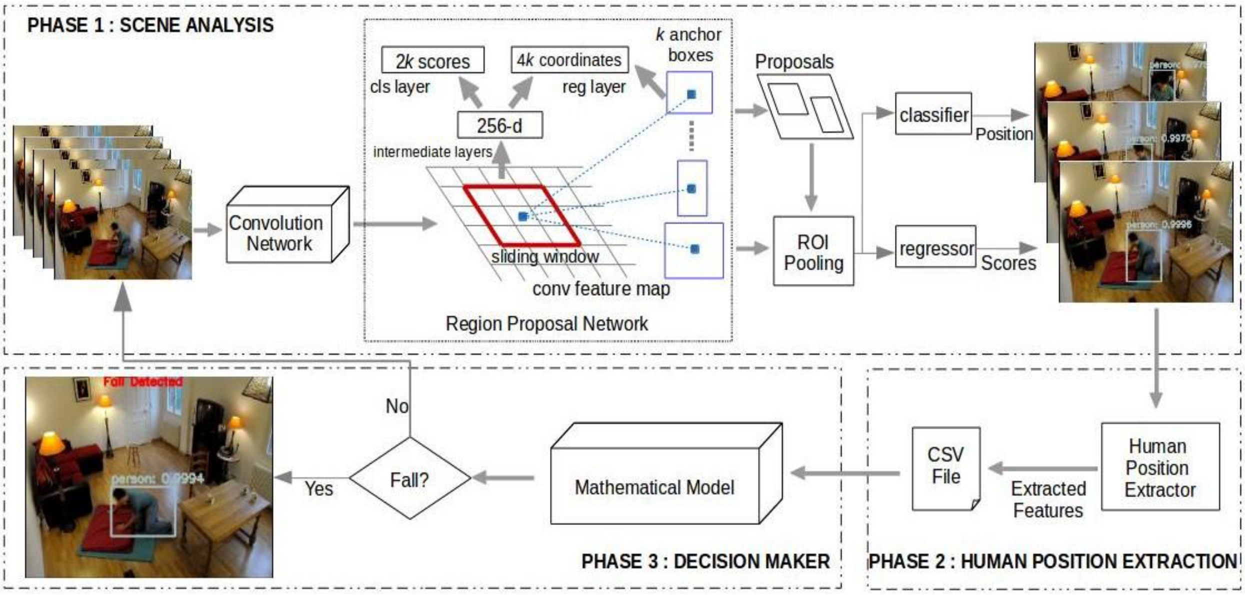


Fig. 1. Model 1 using YOLO

Wang et al. [11] proposed a model which uses PCAnet for extracting features from scene images, then used SVM classifier to classify between a fall and an ADL. Author in [12] combined Histograms of Oriented Gradients (HOG), Local Binary Pattern (LBP) and features extracted from a Caffe [13] neural network to recognize a silhouette and then applied a SVM classifier. Authors in [14] modified Harcascade in order to find hybrid features ranging from Haar-like features to motion boundary histogram (MBH). When working on vision based systems, the first task is to detect a human in a scene. Many authors designed their own human detection neural network model while others used predefined neural network architectures for human detection. In our work, we developed three models based on vision based method. First two models use a pre-trained model while in the third model we developed our own neural network in order to detect a fall. A comparative analysis of all three models is provided in section V.

III. FALL DETECTION USING MATHEMATICAL APPROACH

This paper proposes three different models for vision-based human fall detection method. First two models are based on mathematical approach combined with the deep learning approach. The working of these two models is divided into three phases as shown in Fig. 2 and 3.

A.Phase 1: Scene Analysis

The main purpose of this phase is to detect and recognize human present in the scene/frames of video. Each of the proposed mathematical models uses different pre-defined deep learning based object detection algorithms in order to detect human and find its accurate location in the scene, i.e., YOLO object detection architecture for Model 1 and Faster R-CNN architecture for Model 2.

- **Model 1:** Model 1 uses YOLO (You Only Look Once) [3] object detection architecture to find a human position in the scene. YOLO a single deep neural network comprised of 24 layers which include 9 convolutional layers, 9 layers using leaky rectifier activation function and 6 max-pooling layers followed by 2 fully connected layers as shown in Fig. 2. Input video frames are divided into an M x M grid. Each of the grid cells are individually responsible for predicting B bounding boxes for a single object. Each bounding box consists of 5 prediction values: (x, y) coordinates of the center of predicted bounding box, width (w) and height (h) of predicted bounding box and confidence score for that box. If the center of an object lies inside a grid cell, then that grid cell is responsible for detecting that particular object. The confidence score will be high in this case. But if a cell does not contain that object, the confidence score should be zero. Frames containing predicted bounding box are then provided as an input to the second phase.
- **Model 2:** For scene analysis, Faster R-CNN [4] architecture is used by Model 2. It is a region based neural network which uses two subnetworks. As shown in Fig. 3, video frames are provided as an input to the first convolutional network which in result generates convolutional feature map.

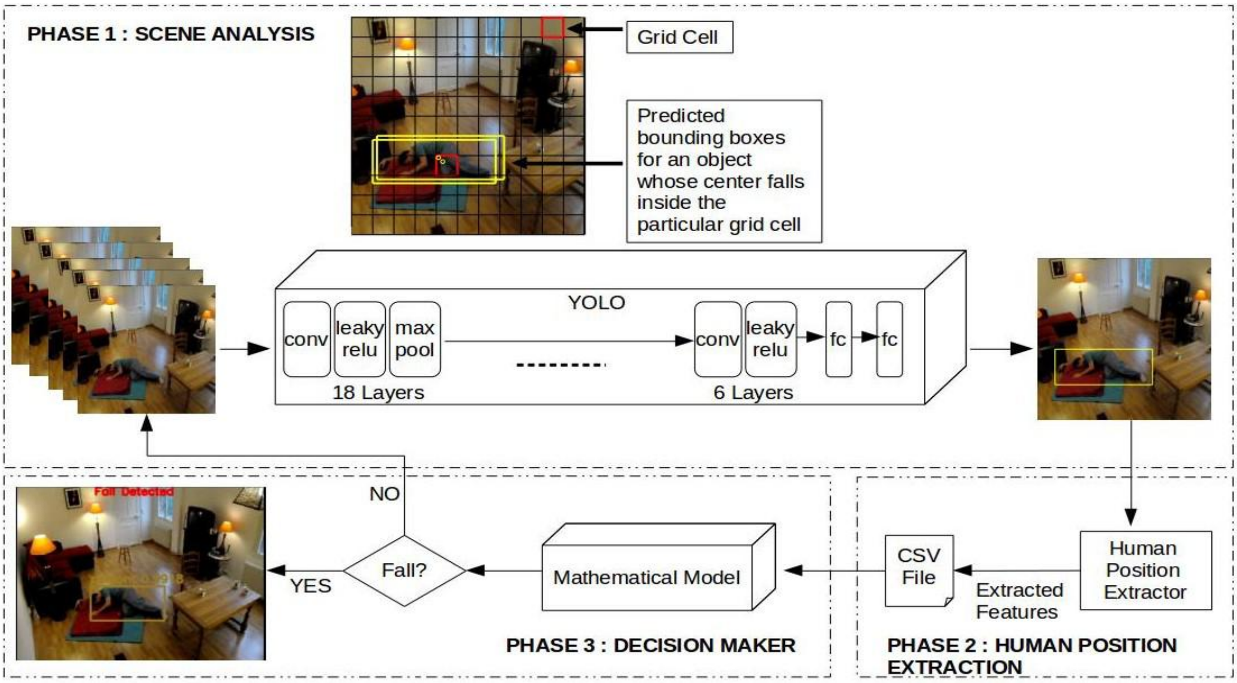


Fig. 2. Model 2 using Faster R-CNN

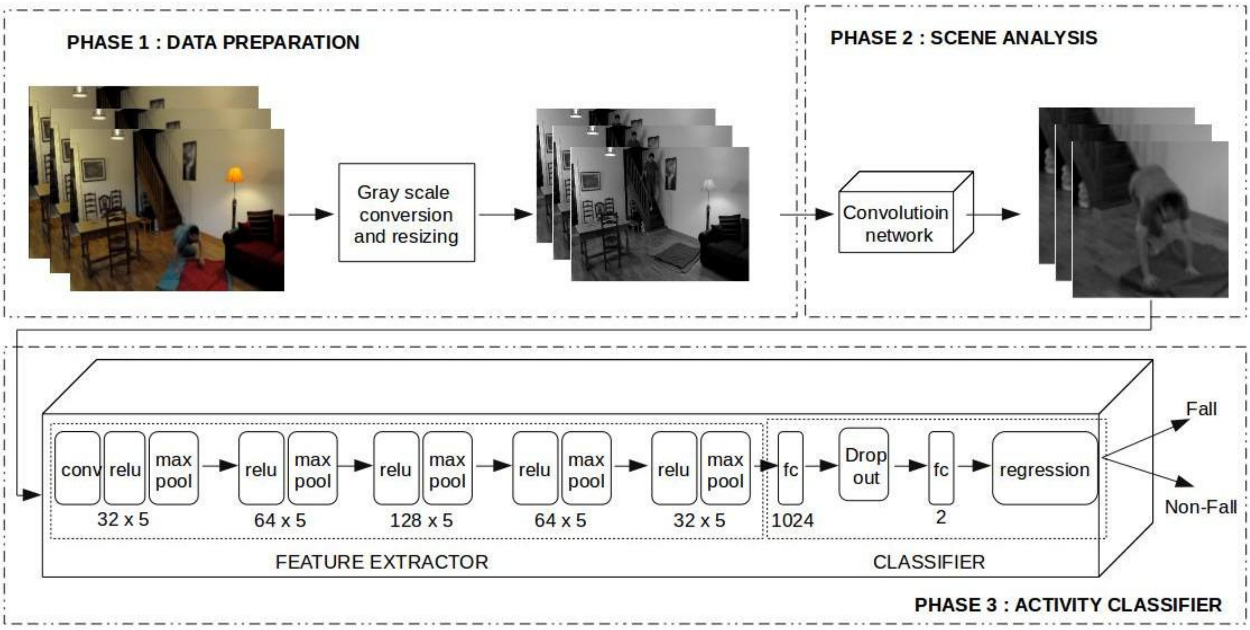


Fig. 3. Model 3 using Deep Neural Network

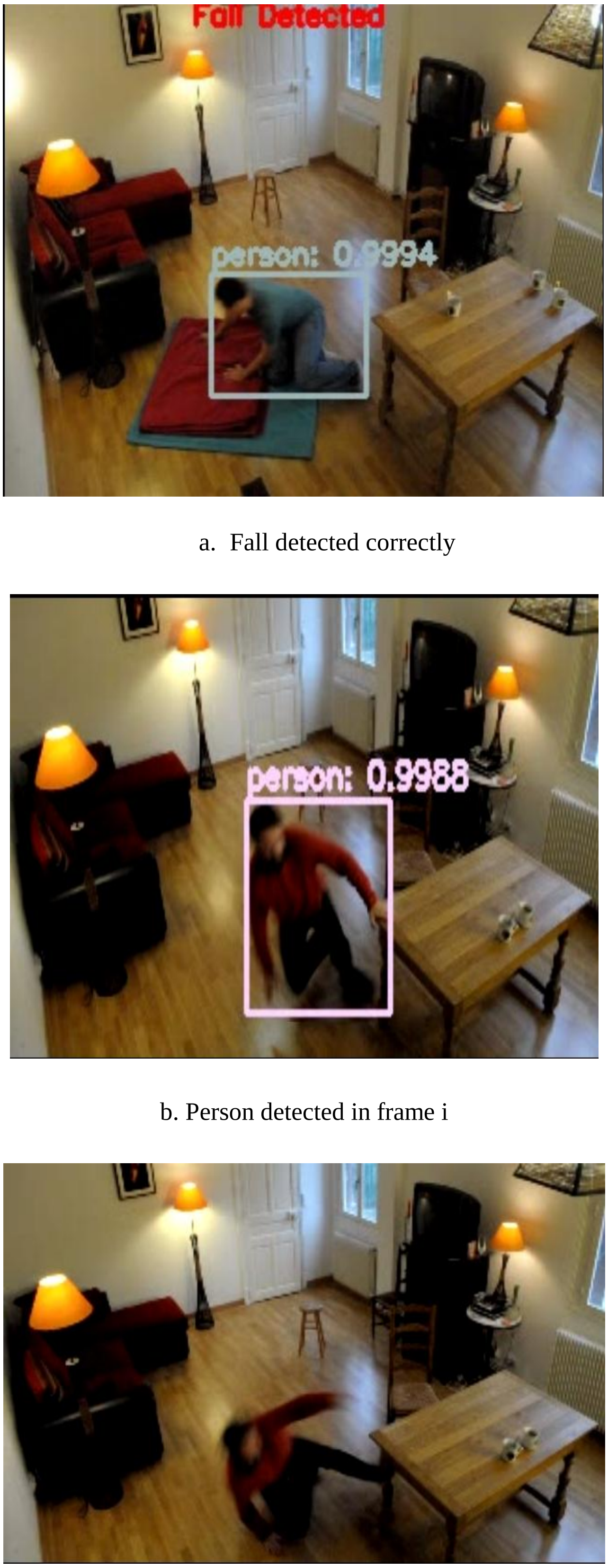
This feature map is then passed as an input to the second network known as region proposal network (RPN). RPN slides a small network over the feature map and picks a small sliding window.

This window is then passed to intermediate layers where each sliding window is decreased to lower-dimensional vector (256-dimension features). This vector is forwarded to two fully connected layers, i.e. a classification layer (cls layer) and a regression layer (reg layer). Multiple region proposals (denoted as k) are predicted for each sliding window location. The k proposals are denoted as k reference boxes, which are also defined as anchors. Therefore, for k predicted boxes, the reg layer generates 4k outputs to encode the coordinates of k boxes, and the cls layer generates 2k outputs that score the estimated probability of object for each proposal. Each anchor is centered at the sliding window in question and is connected with a scale and aspect ratio in the RPN. Regions identified (bounding boxes) are then reshaped and resized using an ROI pooling layer and are then passed to a classifier which classifies the image within the proposed region and to a regressor which predicts the offset value for the bounding box. Frames containing region proposals are then passed to second phase.

B. Phase 2: Human Position Extraction

The second phase i.e. Human Position Extraction will extract the features from all the frames of a video.

For each video V, the total number of frames in V is represented as N. Frame number i of V is represented as F_i. In order to analyze posture and the actual location of a human presence in each frame, features like center coordinates (X_C; Y_C), width (W) and height (H) of predicted bounding box drawn around the human is extracted from each frame.

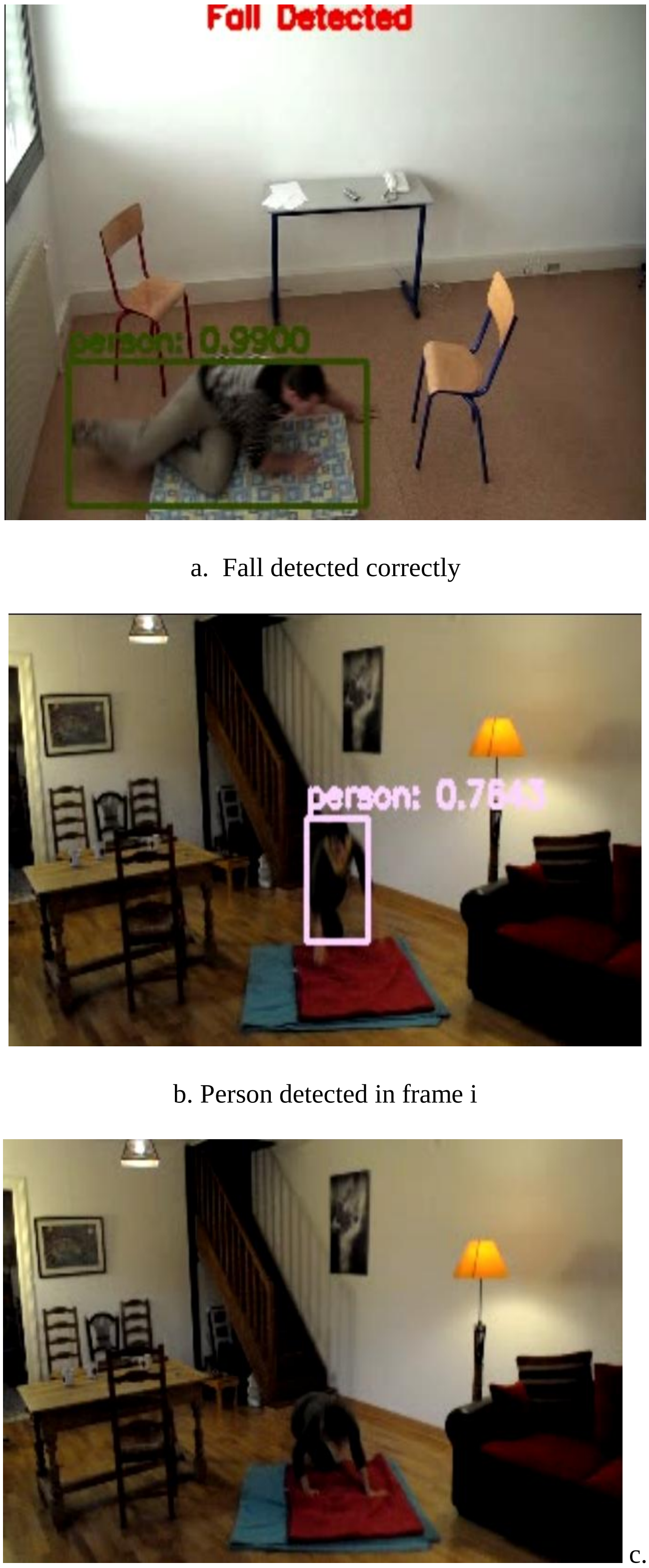


a. Fall detected correctly

b. Person detected in frame i

c. Person not detected in frame i+1

Fig. 4. Detection result of Model 1



a. Fall detected correctly

b. Person detected in frame i

c. Person not detected in frame i+1

Fig. 5. Detection result of Model 2

For every video frame F_i where i ϵ N, an entry for these extracted features is then added to a CSV file. Hence, the location information of each person PM present in each frame Fi is represented as PM = $\{(X_C; Y_C)_i; W_i; H_i | i \in N\}$. At the end of the CSV file, a new column is added which will indicate whether the fall has taken or not in that particular frame. Initially, this column is set to zero and will get updated at the end of the third phase.

C.Phase 3: Decision Maker

The third phase is an automatic decision maker which will take the CSV file as an input and use the mathematical approach to find the change in human position with respect to the optical flow of video frames. To find the accurate relation between a fall human position and a non-fall human position, the last k number of frames need to be referred in the form of optical flow. The model detects a human fall if:

$$H_i <= H_{i-k} * 0.4 \text{ and } W_i >= W_{i-k} * 1.6 \text{ and } H_i < W_i \qquad (1)$$

Where, H_i is the height of the box around human in frame number i, W_i is the width of the box around human in frame i, k is the reference frame number. With the help of above mentioned relative change of human position in frames, our model classifies the human video frames in two categories, i.e. fall and non-fall.

IV. FALL DETECTION USING DEEP LEARNING

In this section, we propose a Deep Neural Network (CNN) that automatically learns the features of a human being and learns how to detect falls by keeping in track change in human position in the optical flow of consecutive video frames. This model is formed of following components: grayscale conversion, scene analysis, CNN classifier, as shown in Fig. 4.

A.Phase 1: Data Preparation

To increase the efficiency of our network, we converted color video frames to grayscale before providing them as an input to our neural network. These converted frames are then resized before providing as an input to the next phase.

B.Phase 2: Scene Analysis

This phase takes the output generated by phase 1 as an input to a four-layer CNN whose job is to find an area of interest in each frame. Area of interest is the area where human is present and its surrounding in a video frame. The objective behind this phase is to remove unwanted information from the scene so as to increase the efficiency of our model.

C.Phase 3: Activity Classifier

Once the area of interest is calculated, it is passed to a deep neural network comprised of two sub networks. The first network is a feature extractor whose job is to extract features from the area of interest frames. It is a five-layer convolution neural network which will take frame by frame as input and extract features from it. Extracted features are then passed to the second neural network for classification. This second network is a fully connected neural network containing two fully connected layers. The first fully connected layer has a dropout chance of 0.8 respectively to prevent overfitting. In the end, regression layer is used to predict the probability of fall and non-fall. For the fine tuning the network and performance optimization of the model, the ADAM optimization algorithm is used during regression layer.

V. EXPERIMENTAL RESULTS AND ANALYSIS

A. System Configuration and Data set

Our proposed models are implemented on a PC having Tensorflow 1.12.0, tflearn 0.3.2, Keras 2.2.4, Opencv 3.4.3 libraries installed in Python 3.5.2 using an Intel Xeon 3.4GHz processor and 32 GB RAM. For measuring the performance of our deep neural network, we used the Fall Detection Dataset (FDD). The dataset consists of 60 videos recorded in a home environment and 64 videos recorded in an office environment. The environment at both the locations was simulated, i.e. everything was staged and persons in the videos were falling on purpose.

B. Experiment Results

The results of our detection models on above mentioned datasets are shown in Fig. 4, Fig. 5 and Fig. 6. Fig. 4 represents the output generated by our model 1. Similarly, Fig. 5 for model 2 and Fig. 6 for model 3. According to fig 4(a), our model predicts a correct fall since YOLO easily detects human being in the frame. But in Fig. 5(b) and (c), we found that YOLO was not able to detect human being in all the frames because of which our mathematical model was not able to differentiate between a fall and a non-fall posture and hence detects a fall as non-fall.

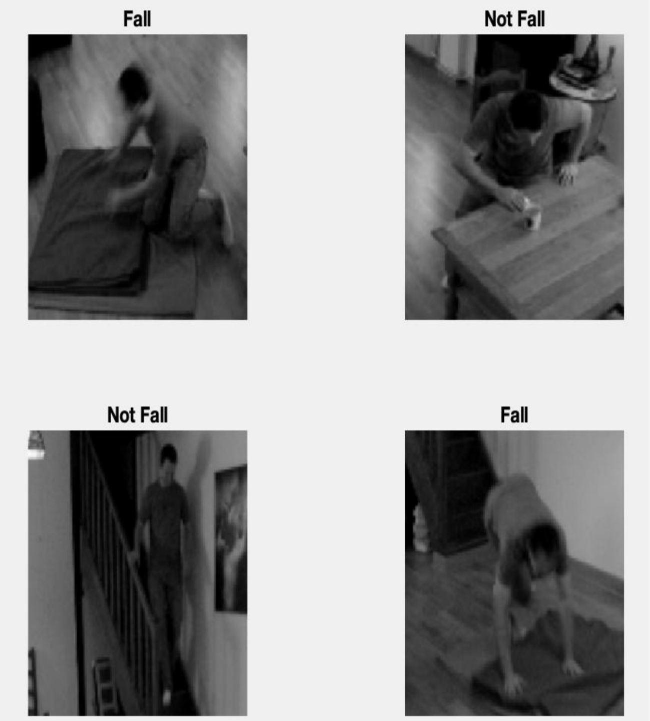


Fig. 6. Detection result of Model 3

Table-I: Performance matrix of model 3

No. of frames referred	Real result	Prediction result	
		Positive	Negative
K = 1	True	65	4314
	False	15	19
K=10	True	79	4320
	False	9	5

By testing model 1 on dataset, we concluded that in most of the videos YOLO was not able to detect a human in video frames in which fall was taking place. Because of this limitation we used another pre-trained object architecture, i.e. Faster R-CNN. Fig. 5(a) provides the output of our model 2 where a fall is correctly detected as a fall. After comparing the output of our model 1 and 2, we found that Faster R-CNN predicts the human being more efficient. But it has also its limitations. According to Fig. 5(b), model 2 was not able to detect falls in the forward direction. As for Fig. 6, it shows the output generated by our model 3. It can be seen that some of the falls which are not detected by YOLO model, are detected by Faster R-CNN model. And also, falls which are not detected by both mathematical models are detected by model 3.

C. Performance Measurement

We used three parameters to evaluate the performance of the models; sensitivity, specificity, and accuracy. Here sensitivity tells how well a model detects a fall as a fall, and specificity tells how well an ADL is classified as non-fall.

Accuracy shows the overall ratio of correctly detected events. All these parameters are defined as follows:

$$\text{Sensitivity} = TP / (TP + FP) \qquad (2)$$

$$\text{Specificity} = TN / (TN + FN) \qquad (3)$$

$$\text{Accuracy} = (TP + TN) / \text{ Total Events} \qquad (4)$$

Here, TP stands for True Positive meaning the fall events are correctly classified as a fall event. TN stands for True Negative meaning the non-fall events are correctly classified as a non-fall event. FP stands for False Positive meaning the non-fall events are classified as a fall event. FN stands for False Negative meaning the fall events are classified as a non-fall event. We evaluated the performance of our network based on above mentioned parameters by considering the number of frames referred or viewed at a time (let suppose k) for fall detection.

Table I shows demonstrate the effectiveness of our neural network model based on the value of k.

Table II proves our result by comparing the performance of our proposed models with existing models based on below mentioned metrics. The robustness of our proposed deep neural network model is compared with our two threshold based models based on above provided quantitative analysis techniques. Fig. 7 provides the validation curve of our proposed deep neural network model. It proposes that our trained deep neural network model provides validation more than 98.6. This means that the model 3 is expected to provide ~98.6 % accuracy on new data.

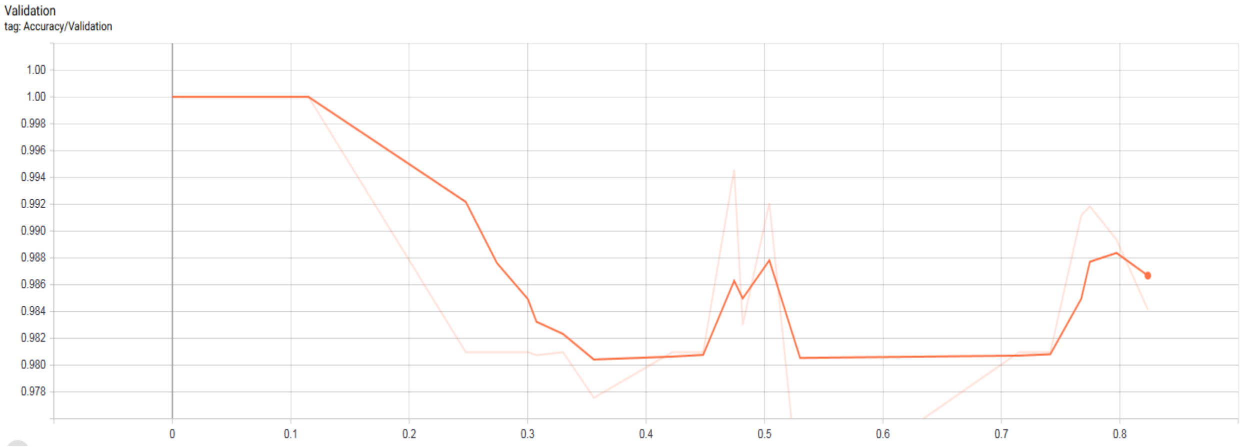


Fig. 7. Validation curve of Model 3.

Table-II: Comparative analysis of different models

Methods	Accuracy (%)	Specificity (%)	Sensitivity (%)
Nu nez-Marcos [53]	97.00	97.00	99.00
Charif et al. [66]	99.61	99.69	98.00
Zerrouki et al. [47]	97.02	-	-
Truls [54]	92.83	93.00	91.90
Hsu et al. [69]	86.30 - 94.10	72.20 - 86.40	96.60 - 100.00
Yun, Gu et al. [70]	-	95.84 - 97.25	98.55 - 100.00
Nguyen et al. [71]	79.60 - 85.40	-	-
Chaccour et al. [72]	94.00	73.00	71.0 - 100.0
Nguyen et al. [21]	96.90	97.60	96.00
Proposed Model 3 with k = 1	99.22	99.65	77.38
Proposed Model 3 with k = 10	99.68	99.79	94.04

VI. CONCLUSION

Fall of a human being can become an important health issue for sick and elder people. Hence, a real-time video surveillance support system needs to design which can make their life in hospitals more comfortable. In this work, we proposed three deep learning models for human fall detection. The first and second model used YOLO and Faster R-CNN deep neural architecture for human detection while in model 3 we proposed our own neural network. All three models take real time surveillance videos as input and print frames in which fall has taken place. During testing of our models, we found that model 3 outperforms model 1 and 2 and also previously existing models in terms of accuracy and specificity on fall detection dataset (FDD).

REFERENCES

1. O. Mohamed, Ho-Jin Choi, and Y. Iraqi, Fall detection systems for elderly care: a survey, 6th International Conference on New Technologies, Mobility and Security (NTMS), pp. 1-4. IEEE, 2014.
2. A. Shojaei-Hashemi, P. Nasiopoulos, J. J. Little, M. T. Pourazad, Videobased Human Fall Detection in Smart Homes Using Deep Learning, In 2018 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1-5. IEEE, 2018.
3. J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You Only Look Once:Unified, Real-Time Object Detection, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779-788, 2016.

4. S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: Towards realtimeobject detection with region proposal networks, IEEE Transactions on
Pattern Analysis Machine Intelligence ,Issue:6, vol. 39, pp. 11371149.
5. K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556, 2014.
6. A. Nu nez-Marcos, G. Azkune, I. Arganda-Carreras, Vision-Based Fall Detection with Convolutional Neural Networks, Wireless Communications and Mobile Computing, 2017.
7. T. Haraldsson, Real-time Vision-based Fall Detection with Motion History Images and Convolutional Neural Networks, Lule University of Technology, Department of Computer Science, Electrical and Space Engineering, 2018.
8. W. Min, H. Cui, H, Rao, Z.n Li, L. Yao, Detection of Human Falls on Furniture Using Scene Analysis Based on Deep Learning and Activity Characteristics, IEEE Access, 6, pp.9324-9335, 2018.
9. B. Jokanovi and M. Amin, Fall Detection Using Deep Learning in Range-Doppler Radars, IEEE Transactions on Aerospace and Electronic Systems, Issue:1, vol. 54, pp. 180-189, 2018.
10. A. Shahroudy, J. Liu, T.-T. Ng, G. Wang, NTU RGB+D: A Large Scale Dataset for 3D Human Activity Analysis, In proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1010-1019, 2016.
11. S. Wang, L. Chen, Z. Zhou, X. Sun, J. Dong, Human fall detection insurveillance video based on PCANet, Multimedia Tools and Applications, Issue:19, vol. 25, pp. 11603-11613, 2016.
12. K. Wang, G. Cao, D. Meng, W. Chen, W. Cao, Automatic fall detection of human in video using combination of features, In proceedings of IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 1228-1233, 2016.
13. Y. Jia, E. Shelhamer, J.Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional Architecture for Fast Feature Embedding, arXiv preprint arXiv:1408.5093, 2014.
14. H. Liu, D. Liu, X. Sun, F. Wu, W. Zeng, On-line fall detection via a boosted cascade of hybrid features, In proceedings of IEEE International Conference on Multimedia Expo Workshops (ICMEW), pp. 249254, 2017.

AUTHORS PROFILE

Ms. Komal Singh is working as an Assistant Professor in Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun, India. She has completed her graduation from Graphic Era Deemed to be University, Dehradun and Post-graduation from National institute of technology Kurukshetra. Her area of interests include machine learning and network security.

Mr. Akshay Rajput is working as an Assistant Professor in Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun, India. He has completed his graduation from Graphic Era Deemed to be University, Dehradun and post-graduation from the IIT-Delhi. He has previously worked in Works Application Singapore PTE LTD. His research interests include machine learning, computer networks and IoT.

Dr. Sachin Sharma, Associate Dean, International Affairs and Associate Professor, Department of Computer Science and Engineering at Graphic Era Deemed to be University, Dehradun, UK, India. He is also Co-founder and Chief Technology officer (CTO) of IntelliNexus LLC, Arkansas, USA based company. He also worked as a Senior Systems Engineer at Belkin International, Inc., Irvine, California, USA for two years. He received his Philosophy of Doctorate (Ph.D.) degree in Engineering Science and Systems specialization in Systems Engineering from University of Arkansas at Little Rock, USA with 4.0 out 4.0 GPA and M.S. degree in Systems engineering from University of Arkansas at Little Rock with 4.0 out 4.0 GPA and He received his B.Tech. degree from SRM University, Chennai including two years at University of Arkansas at Little Rock, USA as an International Exchange Student. His research interests include wireless communication networks, IoT, Vehicular ad hoc networking and network security.