Activity Recognition for Indoor Fall Detection Using Convolutional Neural Network

Kripesh Adhikari
Bournemouth University
Media School, Poole, Dorset, UK
kadhikari@bouremouth.ac.uk

Hamid Bouchachia
National Centre for Computer Animation
Bournemouth University, Dorset, UK
abouchachia@bournemouth.ac.uk

Hammadi Nait-Charif
National Centre for Computer Animation
Bournemouth University, Dorset, UK
hncharif@bournemouth.ac.uk

Abstract

Falls are a major health problem in the elderly population. Therefore, a dedicated monitoring system is highly desirable to improve independent living. This paper presents a video based fall detection system in an indoor environment using convolution neural network. Identifying human poses is important in detecting fall events as specific "change of pose" defines a fall. Knowledge of series of poses is a key to detecting fall or non-fall events. A lying pose which may be considered as an after-fall pose is different from other normal activities such as lying/sleeping on the sofa or crawling. This paper uses Convolutional Neural Networks (CNN) to recognise different poses. Using Kinect, the following image combinations are explored: RGB, Depth, RGB-D and background subtracted RGB-D. We have constructed our own dataset by recording different activities performed by different people in different indoor set-ups. Our results suggest that combining RGB background subtracted and Depth with CNN gives the best possible solution for monitoring indoor video based falls.

1 Introduction

Fall can be described as an unintentional or sudden change of position of the body from an upright, sitting or lying position to a lower inclining position as defined in the paper [1]. A fall is an event that results in a person coming to rest on the ground or any lower level unintentionally [2]. Globally, falls are the biggest threats to the elderly with substantial physical, emotional and financial implications. Falls are the biggest economic burden to the society and falls related costs range between 0.85% and 1.5% of the total healthcare expenditures in the USA, Australia and the United Kingdom [3]. Moreover, the rate of fall is increasing among elderly especially above the age of 65 [4]. Although fall accidents cannot be completely prevented, a fall detection system can save lives if it can accurately recognise a fall incident and generate an immediate alert. A monitoring system should be able to distinguish between fall events and normal activities. This is a difficult task as certain daily activities such as abruptly sitting down or lying on the sofa or going from standing position to lying down, have strong similarities to falls. Usually, a fall ends in lying pose with an inactivity period on the floor. Therefore, in this paper, we attempt to recognise 5 different activities which are standing, sitting, lying, bending and crawling with of focus on accurately recognising a lying pose using computer vision and machine learning. A similar type of research work was done by Alhimale et al. [6] where a neural network was implemented to recognise pose for fall detection using human silhouette features after background subtraction. Similarly, Zhang et al. [7] also recognised 5 activities implementing RGBD images for fall detection for elderly.

2 Methodology

To develop a fall detection system, we need to address two major problems: (1) Classification of activities and (2) Analysing the characteristics of a fall represented by sequential change in pose. For example, a sequence of poses that ends in lying can be considered as a fall event whereas the sequence of poses where the final pose is not lying can be a non-fall event.

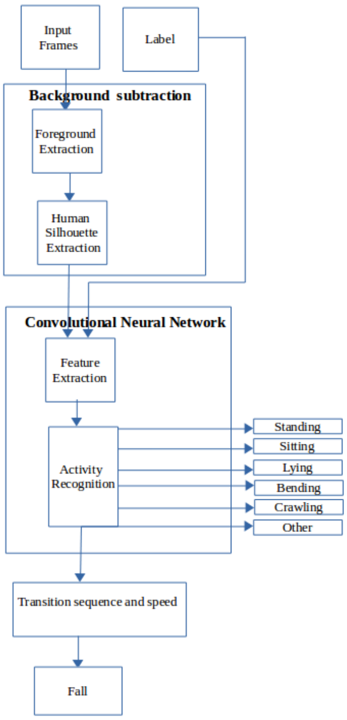


Figure 1. Conceptual Block Diagram.

On the basis of these two important clues, we can identify a fall event. Apart from that, the speed with which these changes happen can also add extra information about fall events.

This paper focuses only on human silhouette extraction and activities recognition for fall detection purpose at this stage. Although CNN is capable of learning from the raw data, the idea of feeding CNN with subtracted silhouette is due to the fact that we are concerned with human silhouette only and the pose. Our model was tested with a different combination of inputs and it was found that feeding human silhouette features from background subtraction is better in terms of accuracy than raw images.
In figure 1, the input frames block represents input images and the label block contains information about the pose which represents a different type of activities in each image. The background subtraction block extracts human silhouette. We shall discuss further the human silhouette extraction using background subtraction in section 2.1.

The convolutional neural network block takes the resulting image after background subtraction and their corresponding labels as input. The network is first trained with the labelled images under supervised learning. Actual fall events have not been applied or tested at this stage. However, a fall event can be considered as the series of change of poses and therefore recognition of all the poses in consecutive frames can indicate that the fall event might have occurred within those frames. However, we need to further verify the falls and fall like events by considering other characteristics such as speed of change of pose, aspect ratio and inclination angle in our future work.

2.1 Human Silhouette Extraction

Detection of human silhouette is possible with the help of background subtraction. Our fall detection is based in an indoor environment where background is relatively static. The foreground obtained after background subtraction is used for human silhouette feature extraction. Let C denote the current image and B the background image. We consider our background to be static and therefore choose it as a reference image where no human or any moving object is present. A pixel $C(i,j)$ in the image C is classified as foreground if $|C(i,j) - B((i,j)| > \tau$ as follows

$$F = \begin{cases} 1, & |C(i,j) - B((i,j)| > \tau \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where τ is a predefined threshold. The threshold plays a significant role in separating foreground from background. We calculate the mean values of the pixels of both RGB and Depth images and used them as the threshold. The human silhouette had some noticeable noise due to the certain variation of light. However, depth images are less affected by lighting changes. Furthermore, depth images also come with its limitation of distance. A fall detection model can suffer due to the curse of distance limitation and noise if only depth type input is used [9] as Kinect only provide depth estimates for up to a limited distance (typically less than 5m) [7]-[8]. Consequently, implementing RGB together with Depth image could be the solution as they can complement each other as the model can use RGB information when depth information is affected due to distance limitation and similarly use depth information when RGB is affected by light variation.

2.2 Proposed Convolutional Neural Network Model

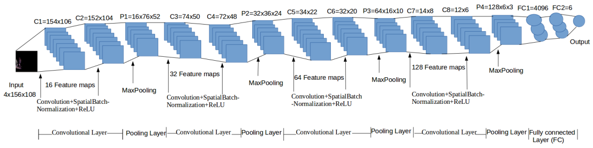


Figure 2. Proposed CNN Architecture

.

In figure 2 illustrated above, an image patch of size 156 x 108 is randomly cropped from the entire image which is of size 160 x 120 scaled from the original size of 640 x 480. The input volume seen by the input layer is equal to a 4 (RGBD) x 156 x 108. C1 and C2 represent the first and second stages of convolution which generate 16 features map that is extracted from 16 filters of size 3 x 3 at a stride of 1 x 1 with zero padding. For batch normalisation, mean and standard deviation are calculated from a mini batch of inputs as a pre-process and then all the features from each feature map are then subtracted by the calculated mean and divided by calculated standard deviation(σ). Then, non-linearity (ReLU) is applied to the inputs to separate high-level and low-level features from the input. Tanh and Sigmoid are the other rectifiers that can be used in place of ReLU. The advantage of using ReLU is that it does not saturate for large inputs. The gradient is always high (equal to 1) if the neuron activates. In contrast Tanh and sigmoid tend to saturate and the gradient is very small when the input is very high or very low. As a result, weights update during backpropagation is also very slow [5].
Further to that, only high-level features are selected with the help of max-pooling from a region. We have two fully connected layers FC1 and FC2. In FC1 layer, we consider 4096 features. A number of features are carefully selected on the basis of observation assuring a significant number of features are considered for recognition and at the same time, these number of features does not lead to over-fitting. Similar, in FC2, we consider 6 non-linear combinations of features out of 4096 features which also represent the class score (6 different classes). Finally, a softmax classifier computes the probability of the class score. The architecture was designed mainly with the reference of VGGnet [12]. However, the parameters were tuned after several trials.

3 Dataset

The images are recorded with Kinect Sensor which has a frame rate of 30 fps. Therefore, the frames captured is fast enough to capture each activity as a pose in a frame and thus each frame has a separate crawling, lying or any other pose and it is labelled accordingly. However, there is a possibility of a presence of another person in the frame as well as the moving of furniture. These scenarios are to be considered in the future work. For training purpose, we have five labelled five poses. Apart from these 5 different poses, the system should also recognise the cases where the occupant is not present or totally occluded, these are labelled as 'other'. The dataset used in this paper is made of raw RGB and Depth images of size 640x480 recorded from a single uncalibrated Kinect sensor. The Kinect sensor is fixed at roof height of approx.2.4m. The dataset contains a total of 21499 images, out of which 15800 images are used for training, 3199 images for cross-validation and 2500 images for testing. The images in the dataset are recorded in 5 different rooms from 8 different view angles. There are altogether 5 different participants. There are two male participants of age 32 and 50 and three female participants of age 19, 28 and 40. All the activities of the participants are limited to 5 different categories of poses that are standing, sitting, lying, bending and crawling. Some images in the dataset are empty which are categorised as 'other'. We have used images of 2 participants: the male of age 32 and the female of age 28 combining total of 15800 images for training, and 3199 images for cross-validation which contains a male participant of age 32 from training set but is in the different room to that of training and testing set. Similarly, the test set contains images of 3 participants, 2 females aged 19 and 40 and a 50 years old male participant. These images are recorded in a different room and have not been used in training or validation.

4 Results and Discussion

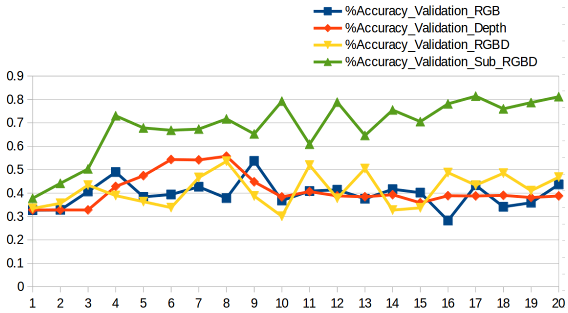


Figure 3. Accuracy plot during test on validation set with different inputs.

Figure 3 illustrates the classification accuracy of the CNN model on a validation set during training with four different types of input images.The plot shows an increase in the accuracy of identifying the correct pose as the number of iteration or epoch increases. This is a positive sign that the model is learning and improving with every iteration. After performing training on each mini-batch, at the end of the epoch, the model is tested on validation set and is saved. Next, at the end of the second epoch, if the new model after testing is better than the previous model, the new model is saved. Otherwise, the old model is preserved and the new is discarded. In this way, we save the best model. Our proposed model was able to achieve the best performance using the subtracted RGBD input with 81% accuracy at the 17th epoch. This can be clearly observed by the green line clearly outperforming all other types.

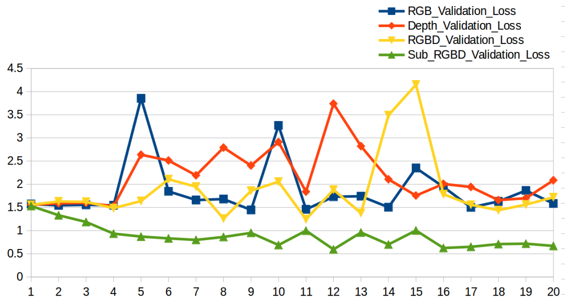
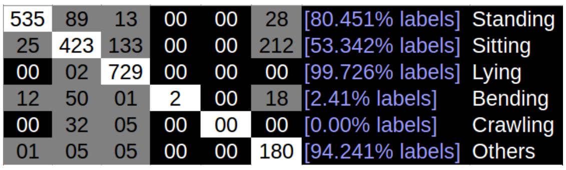


Figure 4. Error plot during test on validation set with different inputs.

Figure 4 illustrates the error observed on the individual inputs. Clearly, subtracted RGBD has outperformed other inputs once again. The minimum error observed by the model was 0.59 at the 12th epoch for sub-RGBD input. A steady error was observed plot for sub-RGBD from the 16th epoch.



535	89	13	00	00	28	[80.451% labels]	Standing
25	423	133	00	00	212	[53.342% labels]	Sitting
00	02	729	00	00	00	[99.726% labels]	Lying
12	50	01	2	00	18	[2.41% labels]	Bending
00	32	05	00	00	00	[0.00% labels]	Crawling
01	05	05	00	00	180	[94.241% labels]	Others

Figure 5. Output of confusion matrix on test.

Finally, the best model was tested on the test set. The overall accuracy on the test set was 74% as illustrated by the output of the test confusion matrix in the figure 5. The elements highlighted in white along the diagonal represents the number of correctly classified poses for the corresponding class in the same order as illustrated in the above matrix. The diagonal elements demonstrate the correctness of the model and the remaining represent the confusion due to miss-classification by the model. Global accuracy is the ratio of the sum of elements in diagonal by the sum of all the elements in the matrix (i.e number of test set).Therefore, an overall accuracy is 74% was achieved.

Similarly, the percentage on the right to the matrix elements associated with the class are the sensitivity of each class. From the above matrix in figure 5, for lying case, the diagonal value that represents the correctly classified class (True Positive) is 729. Only 2 times the model got confused (False Negative) considering actual lying pose (seen from the 3rd column) to be sitting (seen from 2nd row). Therefore, the sensitivity of class is 99%. Furthermore, the sensitivity of 'bending' class is 2% and crawling is 0%. This is very poor and indicates that the class are not well balanced. Similarly, the sensitivity of sitting with 53% is also weak.

However, the model was able to achieve 99% sensitivity in lying pose which is very supportive to the aim of identifying an after-fall pose.

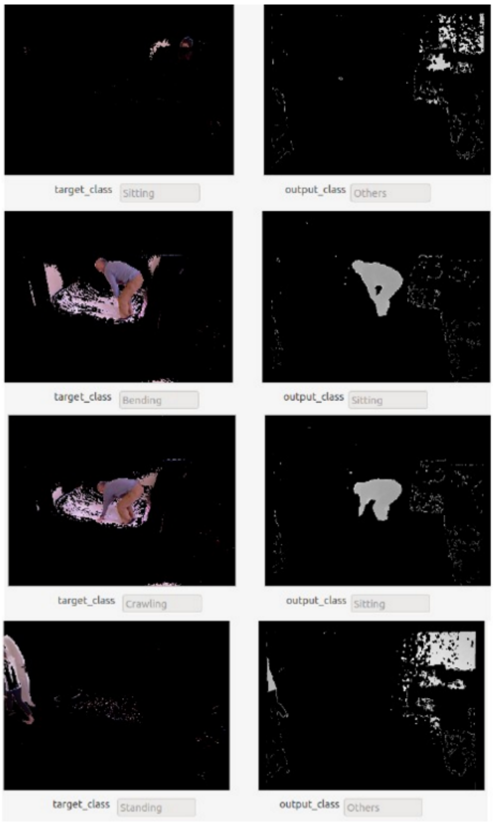


Figure 6. Incorrectly predicted pose on unseen image.

The model is trained to classify an empty image as 'other' in training examples. Therefore, the model is predicting as 'other' in the output class considering the person is absent in the image. This indicates a noisy image is also vulnerable and can be wrongly classified by this model. In the second and third output images of figure 6, the person seems to be in transition from bending to standing or vice-versa. The prediction goes wrong due to the confusion of the pose. In this case, it is actually labelled as 'bending' as the model gets confused. A lesser sensitivity in standing pose is also due to the confusion between standing and bending pose. In the fourth case, only a part of the body is visible while going out or or coming into the room. This is hard to recognise for the model. Here too the model does not get enough information about the orientation of the body and therefore does not recognise a pose. Even to label such partially seen body is difficult, and we assume that including several images of such types in the training data can improve the classification.

5 Conclusion

In this paper, we have proposed an efficient method of activity recognition for fall detection using RGB and Depth images from an inexpensive Kinect sensor. We have used indoor scenarios illustrating activities in real life for a single person. We have tested our proposed CNN model with the different combination of input images to evaluate its performance on activity recognition. The experiment result showed that the CNN trained with background subtracted RGBD is the most appropriate approach for pose recognition. Using the background subtraction, the model has benefited from the greater exposure of the region of interest only, which ultimately helped the model to pick up significant features during convolution from that region.
We were able to achieve 74% accuracy on test data where the person and the scene were never seen in the training set. This is better in comparison to the similar work done in [11]; they were able to achieve only 64.2%. Furthermore, we achieved 99% on analysing the sensitivity of lying pose which is extremely desirable in fall detection where an after-fall pose is considered to be lying. Similarly, our model is not affected by the position of the person and can face any side and angle and walk around. Therefore our sub-RGBD based CNN approach can also perform better allowing free movement within the room in comparison to [6] where they have mentioned that their approach suffers in a situation when a person walks towards the camera.

References

[1] Noury, Norbert, et al. "Fall detection-principles and methods." 2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE, 2007.

[2] Hyndman, Dorit, Ann Ashburn, and Emma Stack. "Fall events among people with stroke living in the community: circumstances of falls and characteristics of fallers." Archives of physical medicine and rehabilitation 83.2 (2002): 165-170.

[3] Heinrich, S., et al. "Cost of falls in old age: a systematic review." Osteoporosis international 21.6 (2010): 891-902.

[4] Anonymous,a Fall, What Can Happen After. "Important Facts about Falls."

[5] Karpathy, Andrej. "Convolutional neural networks for visual recognition." (2015).

[6] Alhimale, Laila, Hussein Zedan, and Ali Al-Bayatti. "The implementation of an intelligent and video-based fall detection system using a neural network." Applied Soft Computing 18 (2014): 59-69.

[7] Zhang, Chenyang, Yingli Tian, and Elizabeth Capezuti. "Privacy preserving automatic fall detection for elderly using RGBD cameras." International Conference on Computers for Handicapped Persons. Springer Berlin Heidelberg, 2012.

[8] Henry, Peter, et al. "RGB-D mapping: Using Kinect-style depth cameras for dense 3D modeling of indoor environments." The International Journal of Robotics Research 31.5 (2012): 647-663.

[9] Stone, Erik E., and Marjorie Skubic. "Fall detection in homes of older adults using the Microsoft Kinect." IEEE journal of biomedical and health informatics 19.1 (2015): 290-301.

[10] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556 (2014).

[11] Sung, Jaeyong, et al. "Human Activity Detection from RGBD Images." plan, activity, and intent recognition 64 (2011).