Deep Learning for Fall Detection: 3D-CNN Combined with LSTM on Video Kinematic Data

Na Lu*, Yidan Wu, Li Feng and Jinbo Song

***Abstract*— Fall detection is an important public healthcare problem. Timely detection could enable instant delivery of medical service to the injured. A popular non-intrusive solution for fall detection is based on videos obtained through ambient camera, and the corresponding methods usually require a large dataset to train a classifier and are inclined to be influenced by the image quality. However, it is hard to collect fall data and instead simulated falls are recorded to construct the training dataset, which is restricted to limited quantity. To address these problems, a three-dimensional convolutional neural network (3D CNN) based method for fall detection is developed which only uses video kinematic data to train an automatic feature extractor and could circumvent the requirement for large fall dataset of deep learning solution. 2D CNN could only encode spatial information, and the employed 3D convolution could extract motion feature from temporal sequence, which is important for fall detection. To further locate the region of interest in each frame, a LSTM (Long Short-Term Memory) based spatial visual attention scheme is incorporated. Sports dataset Sports-1M with no fall examples is employed to train the 3D CNN, which is then combined with LSTM to train a classifier with fall dataset. Experiments have verified the proposed scheme on fall detection benchmark with high accuracy as 100%. Superior performance has also been obtained on other activity databases.**

***Index Terms*—activity recognition, convolutional neural network, deep learning, fall detection, visual attention**

I. INTRODUCTION

IN the field of public healthcare, fall detection is the crucial technique to improve the survival rate after falls. According to the report from World Health Organization, 28% to 35% of the people at age 65 or over fall each year, and the numbers increase to 32% to 42% for the people of 70 years old or over [1]. These figures get even worse with the increase of the age. A review on fall detection methods has indicated that falls are the major cause of injury related death for seniors aged 79 or more [2]. According to the data from National Institutes of Health of the U.S., about 1.6 million elderly people are suffering from fall related injuries in the United States each year [3]. Meanwhile, the elderly population is increasing rapidly over the world. China is facing the fastest aging population in human history, and about 17% of the population would be elderly people by 2020. The figure will raise to about 35% by 2050 [3]. In Canada, the population aged 65 or over will be about 20% in 2026 [4]. A fact is that about 93% of the seniors live in private homes, among which 29% live alone [4]. It was reported that 50% of the elderly who lay on the floor for more than one hour after falls died within six months after the accident even without direct injuries [5]. There is an urgent need to develop intelligent surveillance system for the older people, which can automatically and immediately detect fall event and inform the caregivers or the family. In this paper, a deep learning solution for fall detection based the combination of 3D CNN and LSTM has been developed, which could well locate the region of interest in the video and encode the motion information.

Some thorough reviews on fall detection methods can be found in [2, 6-8]. According to the devices involved, the existing methods could be roughly categorized into two classes, which are wearable sensor based methods and ambient sensor based methods [7]. Among the ambient sensors, camera has been widely used and the fast development of computer vision has boosted the application of video-based methods for fall detection [2]. From the view of methodology, fall detection methods could be classified as rule-based methods and machine learning based methods [7].

Wearable sensors mainly include tilt switch, accelerometer, gyroscope and barometric pressure sensor [7]. Piezoelectric sensor, acoustic sensor, infrared sensor, camera, Kinect RGBD camera etc. are the often used ambient sensors [9]. Most methods based on the data collected from wearable sensors employ thresholding technique or a set of rules to detect falls [10, 11]. Sannino et al. [1] developed a supervised approach which could extract IF-THEN rules to determine whether an event is fall or not based on the data from an accelerometer. Fall activities are discriminated from daily activities through imposing thresholding on the magnitude of inertial frame vertical velocity in [12]. Tamura et al. [11] built a fall detection system with an accelerometer and a gyroscope which triggers a wearable airbag when a fall event is detected. To develop the system, 16 subjects have been recruited to generate mimicked falls and thresholding technique is used to classify the

This work is supported by National Natural Science Foundation of China grant 61673312, Beijing Advanced Innovation Center for Intelligent Robots and Systems grant 2016IRS19, Fundamental Research Funds for the Central Universities, Research Fund for the Doctoral Program of Higher Education of China grant 20100201120040, China Postdoctoral Science Foundation grant 20110491662, 2012T50805.

*Na Lu is with the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, Shaanxi, China, and Beijing Advanced Innovation Center for Intelligent Robots and Systems, Beijing, China (Correspondence e-mail: lvna2009@xjtu.edu.cn). Yidan Wu is with the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, Shaanxi, China. Li Feng is with the Second Medical Imaging Department, Affiliated Hospital of Xi'an Jiaotong Unviersity (Ninth Hospital of Xi'an). Jinbo Song is with the Systems Engineering Institute, Xi'an Jiaotong University, Xi'an, Shaanxi, China.

activities.

Wearable sensors require relative strict positioning and thus bring along inconvenience especially for the seniors who may even forget to wear them [4]. This fact has made the ambient sensor a necessary choice. Video, audio and vibration sensors are the major equipment of this category. A fall detection method based on floor vibration measured by piezoelectric sensor is developed in [13]. In [14], audio segments recorded by a single far-field microphone were modeled by Gaussian mixture model, based on which a support vector machine is trained to classify the audio segments. As a special image and depth sensor, Microsoft Kinect sensor has been applied in multiple studies [9, 15]. The silhouette extracted from the depth image has been employed to estimate the subject posture, through evaluating the height of the head [3], body centroid [15] or both [9]. Video obtained by monocular camera has been widely employed for fall detection in this decade [4, 16-18]. The solutions include one-camera based detection [4, 17] and multi-camera based detection [19, 20]. The fall detection methods employed in the existing monocular systems mainly depend on the extraction of the subject silhouette [21] or trajectory of the moving subject [22]. In multi-camera system, multi-cue analysis from multiple views [19] or 3D reconstruction [18] have been conducted.

All the existing video-based methods need to extract the subject first which is inclined to be influenced by image noise, illumination variation and occlusion [19, 20]. The training data are usually generated by simulated falls under a controlled environment, which makes it difficult to obtain large quantity of training samples and thus restricts the possible performance of the trained classifier. To develop a robust fall detection method which does not depend heavily on the simulated fall data, a deep learning solution for fall detection is proposed in this paper which employed kinematic data to train an efficient motion feature extractor instead of the small scale fall data. In this decade, deep learning [23, 24] especially convolutional neural network (CNN) has achieved great success in fields like object recognition [25], image segmentation [26], image understanding [27], natural language processing [28] and machine translation [29], which requires large training dataset. The small size of the fall dataset could easily lead to overfitting for deep learning solution of fall detection. However, deep learning could automatically learn efficient representation (feature) of objects [30], which inspired us to train a feature extractor using CNN on other activity related video database considering the lack of fall data. The accelerometer and gyroscope data for fall event is even scarce, so it is currently very difficult to develop a deep learning solution based on this type of data. In addition, motion information is very important in case of fall detection. To encode both the spatial information over each frame and the temporal information inbetween frames, a 3D CNN is developed. In the meanwhile, fall event only occurs in part of the image which could actually provide discriminant feature. Therefore, LSTM-based attention mechanism is employed in combination with the 3D CNN to locate the activity and further improve the detection accuracy. The 3D CNN is trained on a kinetic database Sports-1M, which is then used for feature extraction from the fall activity videos. Contiguous frames of the videos are fed to the trained 3D CNN, the output of which is used as the input to the following LSTM, where a classifier is trained. Extensive experiments have been conducted, and the proposed scheme has obtained superior performance on both fall detection benchmark and large activity recognition database.

The paper mainly has three contributions. Firstly, it is the first deep learning solution for fall detection without handcrafted feature extraction to the best knowledge of the authors; secondly, a combinatorial scheme of 3D CNN and LSTM-based visual attention mechanism is developed to extract the spatio-temporal features of the video sequence which has well encoded the motion information within the region of interest; thirdly, kinematic data are used to train the feature extractor without real or simulated fall data, which demonstrates that the spatio-temporal features extracted by 3D CNN can be effectively used for never seen motion recognition.

The paper is organized as follows: Section I introduces the background and related work; Section II gives the details of the proposed method; Section III reports the experiments and discusses the results; Section IV forms the conclusion.

II. Visual Attention Guided 3D Convolutional Neural Network

Convolutional neural network has turned out to be especially efficient for pattern recognition. A brief introduction to CNN could be found in [31]. For single image recognition, 2D CNN could extract effective features to perform tasks like classification and detection. While for videos, the motion information is buried inbetween frames which would get lost when 2D CNN is employed. However, posture change is a key feature to detect fall event which could not be well revealed by single static image. In a previous study [32], a 2D CNN solution with optical flow image as the input has been developed for fall detection, which tried to incorporate motion information via optical flow. Considering that 3D CNN could extract both the spatial and temporal features, it could be a comprehensive and natural choice for motion recognition. In addition, the fall event only occurs at a sub-region of the surveillance image in most cases. Therefore, salient feature may get diminished if the whole frame is treated equally. In addition, LSTM has been successfully combined with 2D CNN to incorporate visual attention and locate the region of interest. Therefore, to address the above discussed two issues, a framework based on 3D CNN and LSTM has been developed. In our study, LSTM is combined with 3D CNN to implement visual attention guided encoding of the motion information embedded in the video sequence.

A. Architecture of the 3D convolutional neural network

Different from 2D CNN, the input to 3D CNN is a clip of video sequence and the convolution is conducted both spatially and temporally. Fig. 1 gives an illustration of 3D convolution where the convolution kernel is of three dimensions, i.e. two spatial dimensions and one temporal depth. The 3D kernel is convolved with the video frames with stride 1 (which means the

step between each convolution operation is 1 voxel) at all dimensions and the convolution result of each feature map is a feature cube as shown in Fig. 1.

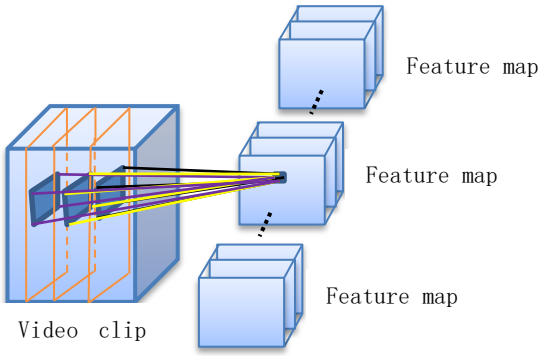

Fig. 1. Illustration of 3D convolution.

Several studies of 3D CNN have been performed in previous research and two representative works are reported in [33, 34]. There are no rigorous theoretical analyses about the optimal architecture of a deep neural network [35]. The architecture and parameters of all the popular deep networks were optimized through experiments [36]. A unanimous recognition is that a deeper neural network has more powerful representation capability [37]. The latest research suggests that the increasing depth of the network has both positive and negative effect, therefore the optimal selection of the depth is a balance between the contradicted influences [38]. It has been indicated in [34] that CNN with small 3D convolutional kernel and deep layers could obtain the best performance according to experiments. Du et al [34] have tested different architectures of 3D CNN on a medium scale database UCF101. It is suggested that homogeneous 3D convolutional kernel of size $3\times3\times3$ could achieve the best performance. Similar finding has been reported in [39] that 2×2 kernel is the best choice for 2D CNN. Liu et al [40] proposed a two-stream 3D CNN for action recognition, which used the spatial and temporal volume of the skeleton joint coordinate as input. A long-term temporal convolutions architecture was proposed in [41] with time dimension incorporated and the importance of the motion information demonstrated. A two-stream flow-guided convolution network was developed by Tran and Cheong [42] which applies a cross-link layer to combine the temporal and spatial network.

Based on the performance comparison of previous studies, we have adopted a 3D CNN with similar structure and similar parameter setting as that of [34]. The major difference is that no full connection layer is employed. The features obtained at the fifth pooling layer are used as the input to the following LSTM, where the attention mechanism will be included. The reason to use the fifth pooling layer rather than the full connection layer is that the pooling layer could provide the spatial correlation between the feature and the input image, while the full connection layer has obscured the relation. The following LSTM aims at discovering the spatial importance over the input image where spatial correlation is required.

The architecture of the 3D CNN is illustrated in Fig. 2. The input to the CNN is an image cube composed of 16 frames segmented from the video sequence. There are eight convolutional layers, where the third and fourth convolutional layers are connected directly, and so are the fifth and sixth layers, seventh and eighth layers. The number of feature maps in the eight convolutional layers are respectively 64, 128, 256, 256, 512, 512, 512 and 512. The dimension of each layer has been indicated in Fig.2. There are five pooling layers where max pooling has been employed, where max filter is employed on non-overlapping regions to down-sample the input representation. To further retain the motion information embedded between frames, the convolution kernel size of the first pooling layer is set as $1\times2\times2$, and for the rest pooling layers, the kernel size is respectively set as $2\times2\times2$, $2\times2\times2$, $2\times2\times2$ and $1\times1\times2$.

With the goal of fall detection where kinetic feature is salient, the dataset of sports videos Sports-1M has been selected to train

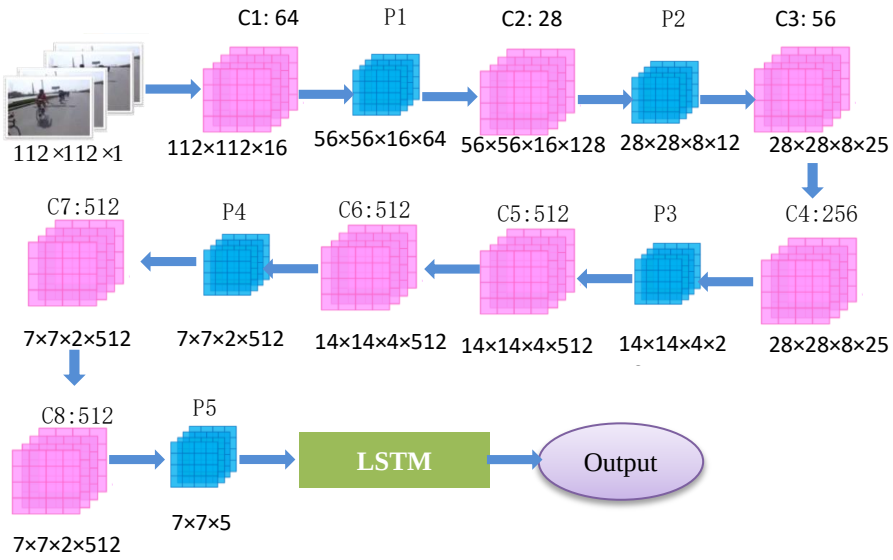

Fig. 2. Structure of the 3D convolutional neural network.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2018.2808281, IEEE Journal of Biomedical and Health Informatics

the 3D CNN, which is also the largest video classification benchmark currently. Similar to the operations in [34], each frame has been resized into 128×171 and then the input image cube is randomly clipped from the video with length of 16 frames and size of 112×112 for each frame to incorporate temporal and spatial jittering. Thus, the input to the CNN is 112×112×16 as shown in Fig. 2. With one input propagated through the network, a feature cube of 7×7×512 will be obtained at the fifth pooling layer, which will then be fed to train the following LSTM.

B. LSTM-based visual attention model

Through the above 3D CNN, a feature cube of $k\times k\times L$ ($7\times7\times512$ in this study) is obtained for each 16 frames of segment, where each dimension of the feature is treated equally. However, the activity related feature may only present itself at specific position or time period. Therefore, introduction of an attention mechanism is expected to improve the classification performance. LSTM model could introduce temporal memory naturally [43]. The conventional LSTM incorporates all the input as vector, which could not preserve the spatial correlation within an image. Xu et al [44] and Sharma et al [45] have developed spatial attention model based on LSTM, which combines the temporal and spatial attention mechanism into a single scheme. The proposed model has been respectively used for image captioning and action recognition in their work. For each frame, a feature map tensor is obtained from the last convolutional layer of a GoogleNet [46] and fed to the following LSTM to predict the action label and the region of attention. However, the motion within a video has been ignored in this method. Li et al [47] developed VideoLSTM which used both motion (optical flow) and appearance feature as the input. A hybrid deep learning framework was developed in [48] which has combined two CNNs with the static image and optical flow as the input respectively. The output of the two CNNs was separately fed to two LSTM networks and then integrated by a regularized fusion network. In the work of Dohahue et al [49], LSTM has been combined with 2D CNN and CRF for activity recognition.

In this study, we have employed similar but modified LSTM soft attention implementation in Xu et al [44] and Sharma et al [45] for 3D feature. It was reported in [45] that no significant improvement has been observed changing the number of LSTM layers. Considering the balance between the computation cost and the classification performance, a simpler yet efficient LSTM network has been adopted. Specifically, one layer of LSTM with 24 cells is used following the 3D CNN. The LSTM can be described by Eqns. (1) to (3), where T is an affine transformation matrix with the parameters to be learned, i_t, f_t, o_t, h_{t-1} and c_t are, respectively, the input gate, forget gate, output gate, hidden state and memory (cell state), g_t select the input to update the memory, L is the dimension of the input to the LSTM, which is 512 in this study, and m is the dimension of i_t. σ and $\odot$ respectively denote the sigmoid activation function and element-wise product.

$$\begin{pmatrix} i_t \\ f_t \\ o_t \\ g_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{pmatrix} T_{m+D,4m} \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} \tag{1}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{2}$$

$$h_t = o_t \odot tanh(c_t) \tag{3}$$

The architecture of the developed CNN is shown in Fig. 3. As mentioned above, the output of the 3D CNN was a feature cube of size $k\times k\times L$, which could be viewed as a k^2L dimensional vector as

$$X_t = \left[X_{t,1}, \cdots, X_{t,k^2} \right], \qquad X_{t,i} \in \mathbb{R}^L, \tag{4}$$

where each column vector $X_{t,i}$ is a L-dimensional vector representing the feature of region i. Then, the feature cube was weighted by a spatial attention weight matrix α of size $k\times k$. For convenience, denote the weight matrix as a $1\times k^2$ vector, and each entry of vector α is determined by

$$\alpha_{t,i} = \frac{\exp(W_i e_{t,i})}{\sum_{j=1}^{k\times k} \exp(W_j e_{t,i})}, \tag{5}$$

where W_i is the weight mapping to element i of the spatial weight matrix, $e_{t,i}$ is obtained as

$$e_{t,i} = tanh(W_a X_{t,i} + W_{ah} h_{t-1} + b_a), \tag{6}$$

which is a multilayer perceptron conditioned on the current input $X_{t,i}$ and the previous hidden state h_{t-1}. The $\alpha_{t,i}$ evaluates the importance of the ith region in the frame of time point t. Given the spatial weight obtained through Eq. (5) and based on the soft attention mechanism in [50], the attention weighted input to the LSTM could be obtained as

$$x_t = \sum_{i=1}^{k\times k} \alpha_{t,i} X_{t,i} . \tag{7}$$

With two separate multilayer perceptrons, the memory state and the hidden state could be initialized as the output of the perceptrons given the input as the average of all the feature cubes from one video, which is similar to what was done in [44]. Based on this initialization, the first spatial weight could be calculated. At each time step, the LSTM will predict the weight matrix for the next time step. The input to the LSTM is a weighted version of the feature cube and the weight matrix, as shown in Fig. 3. The output of the LSTM includes two parts, which are a softmax of the activity labels and a softmax of the weight matrix. The activity class was further obtained through a hyperbolic tangent activation function. The activity class of each video is obtained through averaging the predictions of all the employed LSTMs.

C. Training procedures

As shown in Fig. 4, the video input to the CNN is split into clips of 16 frames with different amount of overlapping between two consecutive clips. The overlapping in Fig. 4 is 8 frames. There are mainly two training stages, which are training of the 3D CNN and training of the LSTM-based attention

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2018.2808281, IEEE Journal of Biomedical and Health Informatics

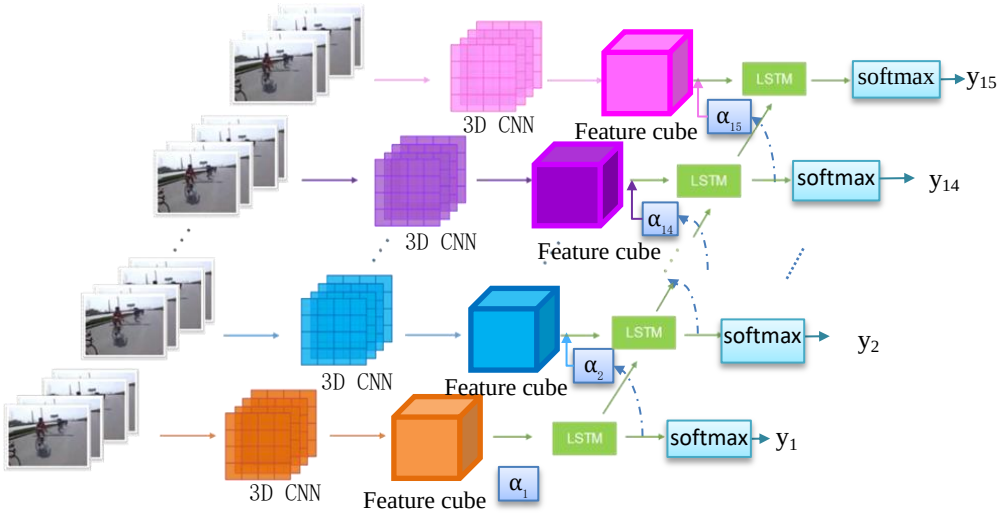

Fig. 3. Illustration of visual attention guided 3D CNN.

network. Sports-1M dataset has been employed to train the 3D CNN which is the largest video benchmark. The LSTM attention model with much less parameters could be trained on a relative small dataset to improve the training speed. Therefore, the 3D CNN and LSTM have been separately trained. The 3D CNN is pre-trained on dataset Sport-1M and LSTM is trained on different dataset for specific application. To train the LSTM based attention model, the 3D feature cube obtained through the pre-trained 3D CNN will be fed into the LSTM as input. Adam optimization algorithm [51] has been employed to train the model with batch size of 30 examples. The initial learning rate is set as 0.003 which is divided by 2 after each 150K iterations. For all the non-recurrent connections, dropout of 0.5 is adopted [52]. The weights of the 3D CNN are not fine tuned while training the attention model.

III. Experiments and Discussions

A. *Datasets and experiment setting*

Sports-1M dataset [53] consists of 1.1 million sports videos belonging to 487 classes, including wrestling, bowling, figure skating, gymnastics and so on. These sport videos have exhibited various human motions and thus could well provide kinematic information for extracting motion related feature. The dataset was split by the ratio of 7:1:2, respectively as the training set, validation set and test set.

Fall detection experiments have been performed on Multiple Cameras Fall Dataset [54]. This dataset covers 24 scenarios and in each scenario, nine activities could happen including walking, falling, lying on the ground, crouching and so on. For each scenario, eight cameras separately captured the scene which thus resulted in 192 videos including 736 actions in total. The size of each frame is 720×480 and the frame rate is 30 fps. The dataset is split by the ratio of 4:1 as training set and test set. Comparisons among different methods are conducted on the same split which is also the case for other dataset. The fall detection dataset (FDD) reported in [55] has also been employed for experiments, which contains 191 videos of daily activities and falls. The frame rate is 25 fps and the resolution is 320×240. The videos are recorded from seven different locations. The dataset is also split by the ratio of 4:1 as training and test set. UR Fall Dataset (URFD) [56] with 70 videos (30 falls and 40 activities of daily living) has also been tested with the same experiment setting.

Experiments on dataset UCF11 [45] have also been conducted to demonstrate the performance of the proposed scheme on activity classification. UCF11 is the YouTube Action dataset which is composed of 1,600 videos of 11 actions, including basketball shooting, biking/cycling, diving and so on. Each video only include one action. The frame rate of the video is 29.97 fps. UCF11 is divided into training set, validation set and test set by the ratio of 3:1:1. In addition, experiments on UCF101 [57] have been performed, which includes 13K clips of YouTube videos from 101 categories of activities with resolution of 320×240. Same experiment setting has been adopted.

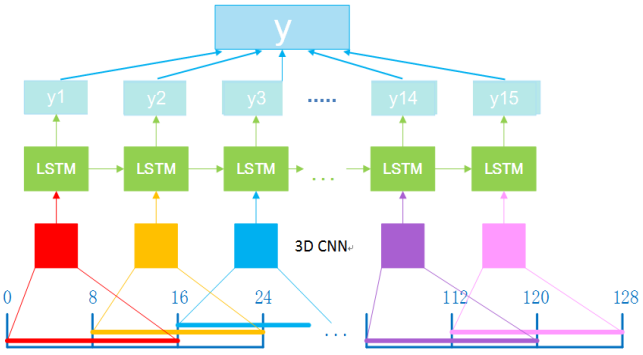

Fig. 4. Illustration of the splitting of the input video.

For further activity classification comparison, experiments on dataset HMDB-51 [58] have also been performed. HMDB-51 is a large human motion database which provides three train-test splits of 51 action categories. Each split includes 5,100 videos of human actions like jump, hug and clapping. In each split, the training and test set respectively have 3,570 videos and 1,530 videos.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2018.2808281, IEEE Journal of Biomedical and Health Informatics

In the experiments, all the videos have been resized to 128×171 and fed to the 3D CNN. The feature obtained at the fifth pooling layer is employed as the input to the following LSTM. A single layer LSTM with 24 cells has been used. Experiments with more cells like 512 have also been conducted which may obtain better performance on one dataset but suggest overfitting when the trained network is used on other datasets. Details could be found in Section III-C-3).

TABLE I
PERFORMANCE OF 3D CNN ON MULTIPLE CAMERAS FALL DATASET WITH ONE FRAME INTERVAL (%)

	Fold1	Fold2	Fold3	Fold4	Fold5	Avg±SD
Positive example	218	194	223	214	206	211±11
Negative example	5456	5480	5451	5461	5468	5463±11
FPR	0.15	0.22	0.17	0.09	0.15	0.15±0.05
FNR	3.67	4.64	2.69	1.87	3.88	3.35±1.08
TPR	96.33	95.36	97.31	98.13	96.12	96.65±1.08
TNR	99.85	99.78	99.83	99.91	99.85	99.85±0.05
Accuracy	99.72	99.63	99.74	99.84	99.72	99.73±0.07

TABLE II
PERFORMANCE COMPARISON VERSUS STATE-OF-THE-ART METHODS

Method	Accuracy (%)
Silhouette Area Variation [19]	94.01
Full Procrustes distance [4]	96.20
Mean matching cost [4]	95.40
Bounding box ratio [4]	56.6
2D vertical velocity [4]	89.70
Normalized 2D vertical velocity[4]	87.30
3D CNN	99.73

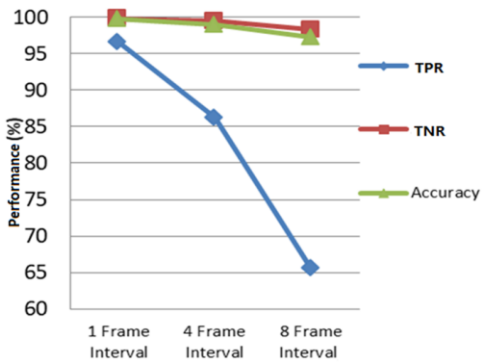

Fig. 5. Performance of 3D CNN on fall dataset with different sampling interval.

To compare with 2D CNN, VGGNet [39] and GoogleNet [46] have been included for comparison, which are representative 2D CNNs. The 2D CNNs are trained on dataset ImageNet [59], and LSTM has also been employed as in our method for fair comparison. The size of each input image to the 2D CNN has been resized to 112x112.

Experiments with 3D CNN as the feature extractor and SVM as the classifier have been conducted, where no attention mechanism has been incorporated. The feature vector obtained from the full connection layer is fed to the SVM as input. The performance has verified the efficiency of the LSTM-based attention mechanism.

TABLE III
PERFORMANCE OF 3D CNN ON MULTIPLE CAMERAS FALL DATASET WITH FOUR FRAMES INTERVAL (%)

	Fold1	Fold2	Fold3	Fold4	Fold5	Avg±SD
Positive example	57	59	50	51	45	52±6
Negative example	1364	1361	1370	1369	1365	1366±4
FPR	0.44	0.44	0.58	0.66	0.07	0.44±0.22
FNR	19.30	11.86	8.00	9.80	20.00	13.79±5.52
TPR	80.70	88.14	92.00	90.20	80.00	86.21±5.52
TNR	99.56	99.56	99.42	99.34	99.93	99.56±0.23
Accuracy	98.80	99.08	99.15	99.01	99.29	99.07±0.18

TABLE IV
PERFORMANCE OF 3D CNN ON MULTIPLE CAMERAS FALL DATASET WITH EIGHT FRAMES INTERVAL (%)

	Fold1	Fold2	Fold3	Fold4	Fold5	Avg±SD
Positive example	18	24	30	22	26	24±4
Negative example	693	687	682	689	685	687±4
FPR	3.17	1.31	1.03	1.31	1.31	1.63±0.87
FNR	27.78	37.50	16.67	36.36	53.85	34.43±13.69
TPR	72.22	62.50	83.33	63.64	46.15	65.57±13.69
TNR	96.83	98.69	98.97	98.69	98.69	98.37±0.87
Accuracy	96.20	97.47	98.31	97.61	96.77	97.27±0.81

B. *Experiments on 3D CNN for fall detection*

This set of experiment aims at evaluating the performance of the 3D CNN. In the experiment, one full-connection layer has been added after the fifth pooling layer. The dimension of the full-connection layer is 4096. A linear SVM was trained using the Multiple Cameras Fall Dataset, where the input to the SVM is obtained by feeding the video to the 3D CNN. Softmax instead of SVM has also been tested and the performance difference between them is quite trivial. Each video is split into consecutive clips of 16 frames with a sampling interval of one frame. Thus, for each clip, a 4096-dimensional feature vector will be obtained. To train the linear SVM, a five-fold cross-validation scheme is employed. Specifically, the Multiple Cameras Fall Dataset is divided into five subsets of equal size, four of which are used for training and the remaining one for testing. For different dataset split, the classifier was trained separately based on the features extracted by 3D CNN.

The experimental results are reported in Table I. The algorithm performance is evaluated in terms of true positive rate (TPR), true negative rate (TNR), false positive rate (FPR), false negative rate (FNR) and accuracy.

It can be seen from Table I that the performance of the 3D CNN on fall detection is quite good. An average classification accuracy of 99.73% has been obtained, and the average TPR and TNR are, respectively, 96.65% and 99.85%. To compare with the performance of other methods, Table II gives some comparison results against several state-of-the-art methods as reported in [16]. The compared algorithms have been evaluated on Multiple Cameras Fall Dataset. Detection accuracy is employed to evaluate the performance. It can be seen from the

results that the 3D CNN has obtained the best classification accuracy among the compared methods. An increment of 3.53% against the second best result can be observed, which has verified the efficiency of the 3D CNN for fall detection.

In addition, to test the performance of 3D CNN under different sampling scheme, experiments on variant length of sampling interval between consecutive clips have been conducted. The results are summarized in Tables III and IV, respectively, with four and eight frames of interval. To give a better illustration, the performance variation with different sampling interval has been plotted in Fig. 5. When the interval is four frames, the processing frame rate is 41 fps. When the interval is eight frames, 81 fps of processing frame rate could be reached. It could be seen from Tables III and IV that the decrease of the accuracy and TNR is not drastic, while the TPR has, respectively, decreased by 10.44% and 31.08% with the increase of the sampling interval as compared with the results of Table I. When the interval between two consecutive clips increases, some informative frames of fall event may get missed, which thus would lead to decrease of positive samples and TPR. It can be concluded that it is not an appropriate option to raise the sampling interval for computation speedup in case of activity recognition.

C. Experiments on visual attention guided 3D CNN

1) Illustration of visual attention results

LSTM-based visual attention model has been combined with the 3D output of CNN to locate the key region of attention. To provide an illustration of the attentional focus obtained through LSTM, Fig. 6 visualizes some experiment results of dataset UCF11, where the top rows are the original frames and the bottom rows are the visualization of the attentional regions. The attention model based on LSTM employed in the paper is a soft weight model and the corresponding weight over the image has been visualized in a bright blob. The larger the weight is, the brighter the related pixel. It can be seen from the results in Fig. 6 that the key regions which are most informative for each image have been correctly discovered. In most cases of the experiments on dataset UCF11, appropriate attentional focus has been obtained, which could efficiently improve the classification performance as reported in the rest of this Section.

2) Performance on fall detection

With the attention mechanism incorporated, the performance of the 3D CNN on Multiple Cameras Fall Dataset has been further improved. In this set of experiments, the kernel size of the pool5 layer is also set as $1\times1\times2$. Accordingly, the dimension of the feature cube fed to the following LSTM is $7\times7\times512$. The overlapping between two consecutive clips is eight frames. Five-fold cross validation is employed. The obtained classification accuracy is 100%, which is the best performance on the Multiple Cameras Fall Dataset ever reported. The 3D CNN could well encode the motion related temporal feature, and the LSTM based attention mechanism could locate the most informative part for fall detection in the image, which have ensured the good performance of the proposed method.

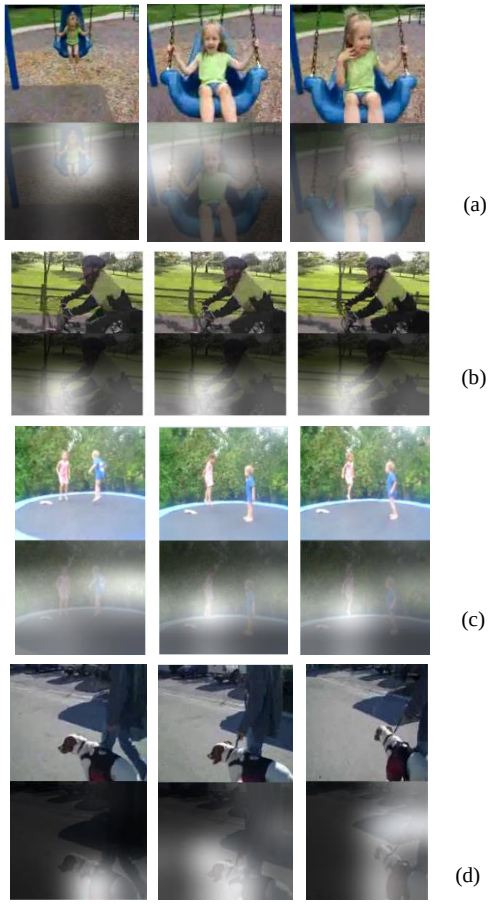

Fig. 6. Attention region over different videos located by visual attention guided 3D CNN. In each sub-figure, the top row is the original frame, and the bottom row is the attention localization result.

To further confirm the performance of the proposed method, experiments on FDD and URFD have also been conducted. An average classification accuracy of 99.36% and 99.27% has been obtained respectively. These results have further verified the efficiency of the proposed scheme on fall detection.

3) Performance on activity recognition

To verify the performance of the visual attention guided 3D CNN on activity recognition benchmark, experiments on dataset UCF11 and UCF101 have been conducted. Multiple experiments have been carried out and the results of five experiments are given in Table V. It can be seen from Table V that the average classification accuracy on dataset UCF11 and UCF101 is respectively 92.01% and 90.46% which is superior or comparable to the state-of-the-art performance. For better illustration, Table VI presents comparison of the proposed method versus some state-of-the-art methods [45]. Same attention mechanism with LSTM of the same dimension has been combined with the renowned GoogleNet [46] and VGGNet [39], which are 2D CNN. Same training set and testing set have been employed for all the compared methods to ensure a fair comparison. Classification accuracy is adopted as the evaluation criterion. Among these compared methods, visual attention guided 3D CNN has obtained the best classification accuracy, which has improved the performance by 7.05% (UCF11) and 6.19% (UCF101) against the second best method, which is GoogleNet with LSTM soft attention.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JBHI.2018.2808281, IEEE Journal of Biomedical and Health Informatics

Further experiments on different dimensional LSTM have been conducted and the results are reported in Table VII, where the best results are highlighted in bold. The numbers in the parenthesis indicate the dimension of the employed LSTM, that is the number of cells. The results of 3D CNN combined with LSTM and average pooling instead of the soft attention model discussed in Section II-B have also been reported in Table VII. The deep model used for UCF11 and HMDB-51 was trained separately. Classification accuracy is employed as the performance evaluation criterion. These results have shown the superior performance of the proposed method, which has obtained the best results among the three compared methods. The efficiency of the attention model has also been verified by the performance improvement of the LSTM attention model against the LSTM average pooling. Paired *t*-test has been conducted between the proposed visual attention guided 3D CNN and the other two methods (VGGNet with LSTM soft attention and 3D CNN with LSTM average pooling), and the obtained *p*-value are repectively 6.5×10^{-5} and 0.06. This suggests that the performance improvement of the proposed method against VGGNet with LSTM soft attention is significant. However, the difference between the 3D CNNs with and without attention mechanism is not significant.

In addition, with the increase of the LSTM cells, the performance of all the compared methods can be improved, as shown in Table VII. Meanwhile, the computation complexity will increase accordingly. Specifically, when 512 cells were employed instead of 24 cells, the computation speed of the LSTM is about 10 times slower. In addition, the deep model trained on UCF11 can be directly used on HMDB-51, where the classification accuracy is 43.23% (24 cells) and 40.02% (512

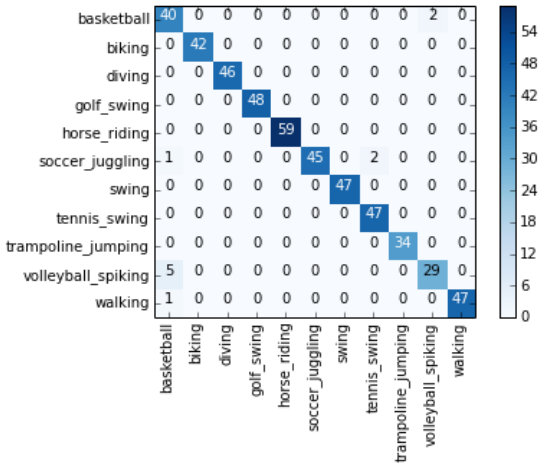

Fig. 7. Confusion matrix on UCF11 dataset of the proposed method.

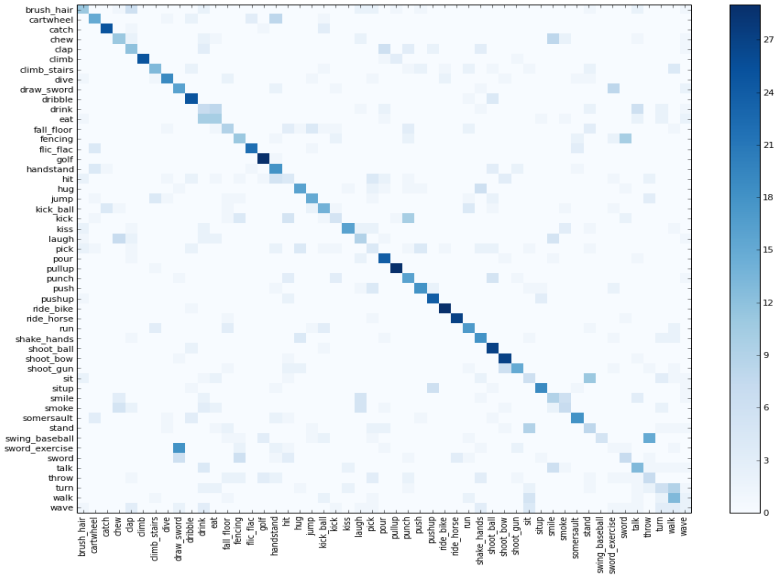

Fig. 8. Confusion matrix on HMDB-51 dataset of the proposed method.

TABLE V
EXPERIMENT RESULTS OF THE AVERAGE CLASSIFICATION ACCURACY (%) OF THE VISUAL ATTENTION GUIDED 3D CNN ON DATASET UCF11

	Exp1	Exp2	Exp3	Exp4	Exp5	Avg
Training	91.81	93.01	92.11	91.71	91.73	92.07±0.55
Validation	93.03	90.61	90.00	90.00	93.03	91.33±1.57
Testing	92.16	92.42	92.12	91.52	91.82	92.01±0.35

TABLE VI
COMPARISON RESULTS OF THE AVERAGE CLASSIFICATION ACCURACY (%) OF THE VISUAL ATTENTION GUIDED 3D CNN VERSUS OTHER STATE-OF-THE-ART METHODS ON DATASET UCF11 AND UCF101

Dataset	Visual attention guided 3D CNN	GoogleNet [46] with average pooling	GoogleNet with max pooling	GoogleNet with LSTM soft attention	VGGNet [39] with LSTM soft attention
UCF11	**92.01±0.35**	82.56±0.98	81.60±1.75	84.96±0.32	84.87±0.53
UCF101	**90.46±0.38**	81.13±0.55	81.00±0.65	83.38±0.39	84.27±0.49

TABLE VII
AVERAGE ACCURACY (%) OF DIFFERENT DIMENSIONAL LSTM ON DATASET UCF11 AND HMDB-51

	UCF11(24)	UCF11(64)	UCF11(128)	HMDB-51(24)	HMDB-51(512)	HMDB-51(1024)
VGGNet with LSTM soft attention	84.87±0.53	88.38±0.43	91.10±0.62	43.01±0.37	44.02±0.59	44.89±0.72
Visual attention guided 3D CNN	**92.01±0.35**	**97.78±0.41**	**97.88±0.58**	**48.62±0.58**	**50.65±0.39**	**50.58±0.73**
3D CNN with LSTM average pooling	84.27±0.36	96.97±0.43	97.17±0.56	46.84±0.50	48.04±0.49	48.68±0.70

cells) respectively. This suggests overfitting with 512 cells employed. Therefore, the cell number is set as 24 in other experiments of the paper.

To give a better illustration of the algorithm performance, the confusion matrices on dataset UCF11 and HMDB-51 obtained by the proposed method have been given in Fig. 7 and Fig. 8. It could be seen from Fig. 7 that most of the activities (7 out of 11) have been correctly classified. While for HMDB-51 dataset, the performance is relatively poor due to the large number of activity classes which is 51.

D. Time cost analysis

The training time and runtime of the proposed method have been evaluated. A single K40 Tesla GPU has been employed for all the experiments. The training duration on dataset Sports-1M, UCF11 and HMDB-51 are respectively about 162, 21 and 72 hours. The runtime (test time) was evaluated by extracting the feature of the whole UCF11 dataset, which takes about 1 hour. In terms of FPS, it is about 62.8 frames per second.

E. Discussions

The above experiment results have shown that the performance of the proposed scheme on both fall detection and activity recognition is superior or comparable to the state-of-the-art methods. The proposed scheme has incorporated temporal motion information through 3D CNN. The LSTM based attention mechanism has well located the region of interest. The combination of 3D CNN and LSTM has well combined temporal and spatial features with informative region localized. In addition, the proposed scheme has provided an efficient deep learning solution for fall detection with small fall dataset for training. However, the key motion sequence usually includes more than three frames, while inclusion of more frames in 3D CNN will further increase the computational cost. Thus it remains a challenge to cover long term temporal correlation over long sequence. In addition, the current solution could only detect fall event after it has happened. How to predict possible fall event based on motion history remains a problem.

IV. CONCLUSIONS

A visual attention guided 3D CNN is developed in the paper, which has incorporated a soft attention mechanism through LSTM into 3D CNN for video analysis. Motion information is crucial for fall detection and activity recognition. To encode the motion feature efficiently, 3D CNN is employed for feature representation learning. Meanwhile, based on the application of LSTM attention model, temporal and spatial attention has been incorporated into the proposed scheme. In case of fall detection, the 3D CNN could be trained on motion related dataset instead of simulated fall dataset, which is difficult to construct. Sports-1M dataset has been employed to train the 3D CNN which is then used for motion feature extraction. The feature cube obtained through the 3D CNN is then fed to the following one layer of LSTM, which could locate the informative region within each frame of the video. Experiments have shown the effectivity of the proposed scheme on fall detection and activity recognition. Specifically, on the Multiple Cameras Fall Dataset, 100% accuracy could be obtained which is the best result ever reported. Also, compared to other state-of-the-art methods, the proposed scheme has obtained superior performance on activity classification benchmark including UCF11 and HMDB-51.

REFERENCES

[1] G. Sannino, I. De Falco, and G. De Pietro, "A supervised approach to automatically extract a set of rules to support fall detection in an mHealth system," *Applied Soft Computing*, vol. 34, pp. 205-216, 2015.

[2] M. Mubashir, L. Shao, and L. Seed, "A survey on fall detection: Principles and approaches," *Neurocomputing*, vol. 100, pp. 144-152, 2013.

[3] L. Yang, Y. Ren, H. Hu, and B. Tian, "New fast fall detection method based on spatio-temporal context tracking of head by using depth images," *Sensors*, vol. 15, pp. 23004-23019, 2015.

[4] C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Robust video surveillance for fall detection based on human shape deformation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, pp. 611-622, 2011.

[5] S. Lord, S. Smith, and J. Menant, "Vision and falls in older people: risk factors and intervention strategies," *Clinics in Geriatric Medicine*, vol. 26, pp. 569-581, 2010.

[6] R. Igual, C. Medrano, and I. Plaza, "Challenges, issues and trends in fall detection systems," *BioMedical Engineering OnLine*, vol. 12, pp. 1-24, 2013.

[7] N. Pannurat, S. Thiemjarus, and E. Nantajeewarawat, "Automatic fall monitoring: A review," *Sensors*, vol. 14, pp. 12900-12936, 2014.

[8] Z. Zhang, C. Conly, and V. Athitsos, "A survey on vision-based fall detection," in *Proceedings of the 8th ACM International Conference on Pervasive Technologies Related to Assistive Environments*, Corfu, Greece, July 1-3, 2015, pp. 1-7.

[9] E. E. Stone and M. Skubic, "Fall detection in homes of older adults using the Microsoft Kinect," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, pp. 290-301, 2015.

[10] L. J. Kau and C. S. Chen, "A smart phone-based pocket fall accident detection, positioning, and rescue system," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, pp. 44-56, 2015.

[11] T. Tamura, T. Yoshimura, M. Sekine, M. Uchida, and O. Tanaka, "A wearable airbag to prevent fall injuries," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, pp. 910-914, 2009.

[12] G. Wu and S. Xue, "Portable preimpact fall detector with inertial sensors," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 16, pp. 178-183, 2008.

[13] M. Alwan, P. J. Rajendran, S. Kell, D. Mack, S. S. Dalal, M. Wolfe, *et al.*, "A smart and passive floor-vibration based fall detector for elderly," in *Proceedings of the Second Information and Communication Technologies*, Damascus, Syria, April 24-28, 2006, pp. 1003-1007.

[14] M. Popescu and A. Mahnot, "Acoustic fall detection using one-class classifiers," in *Proceedings of Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Minneapolis, MN, USA, September 3-6, 2009, pp. 3505-3508.

[15] L. Yang, Y. Ren, and W. Zhang, "3D depth image analysis for indoor fall detection of elderly people," *Digital Communications and Networks*, vol. 2, pp. 24-34, 2016.

[16] B. Mirmahboub, S. Samavi, N. Karimi, and S. Shirani, "Automatic monocular system for human fall detection based on variations in silhouette area," *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 427-436, 2013.

[17] M. Bosch-Jorge, A.-J. Sánchez-Salmerón, Á. Valera, and C. Ricolfe-Viala, "Fall detection based on the gravity vector using a wide-angle camera," *Expert Systems with Applications*, vol. 41, pp. 7980-7986, 2014.

[18] D. Anderson, R. H. Luke, J. M. Keller, M. Skubic, M. Rantz, and M. Aud, "Linguistic summarization of video for fall detection using voxel person and fuzzy logic," *Computer Vision and Image Understanding*, vol. 113, 2009.

[19] N. Thome, S. Miguet, and S. Ambellouis, "A real-time, multiview fall detection system: A LHMM-based approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, pp. 1522-1532, 2008.

[20] D. Anderson, R. H. Luke, J. M. Keller, M. Skubic, M. Rantz, and M. Aud, "Linguistic summarization of video for fall detection using voxel person and fuzzy logic," *Computer Vision and Image Understanding*, vol. 113, pp. 80-89, 2009.

[21] D. Anderson, J. M. Keller, M. Skubic, X. Chen, and Z. He, "Recognizing falls from silhouettes," in *Proceedings of IEEE Annual International Conference of Engineering in Medicine and Biology Society*, New York, NY, USA, August 30-September 3, 2006, pp. 6388-6391.

[22] H. Nait-Charif and S. J. McKenna, "Activity summarisation and fall detection in a supportive home environment," in *Proceedings of International Conference on Pattern Recognition*, Cambridge, UK, August 26, 2004, pp. 323-326.

[23] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436-444, 2015.

[24] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, pp. 504-507, 2006.

[25] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, June 26-July 1, 2016.

[26] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, pp. 142-158, 2016.

[27] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 1915-1929, 2013.

[28] R. Sarikaya, G. E. Hinton, and A. Deoras, "Application of deep belief networks for natural language understanding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, pp. 778-784, 2014.

[29] H. Geoffrey, D. Li, Y. Dong, D. George E, M. Abdel-rahman, J. Navdeep, *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Processing Magazine*, vol. 29, pp. 82-97, 2012.

[30] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence* vol. 35, pp. 1798-1828, 2013.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proceedings of Advances in Neural Information Processing Systems*, Lake Tahoe, December 3-6, 2012.

[32] A. Nunez-Marcos, C. Azkune, and I. Arganda-Carreras, "Vision-based fall detection with convolutional neural networks," *Wireless Communications and Mobile Computing*, vol. 2017, p. 16, 2017.

[33] S. Ji, W. Xu, M. Yang, and K. Yu, "3D convolutional neural networks for human action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 221-231, 2013.

[34] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proceedings of IEEE International Conference on Computer Vision*, Chile, December 13-16, 2015, pp. 4489-4497.

[35] J. Feng and T. Darrell, "Learning the structure of deep convolutional networks," in *Proceedings of IEEE International Conference on Computer Vision*, Chile, December 13-16, 2015, pp. 2749-2757.

[36] G. Huang, Z. Liu, K. Q. Weinberger, and L. v. Maaten, "Densely connected convolutional networks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, Hawaii, July 21-26, 2017.

[37] M. Bianchini and F. Scarselli, "On the complexity of neural network classifiers: A comparison between shallow and deep architectures," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, pp. 1553-1565, 2014.

[38] S. Sun, W. Chen, L. Wang, X. Liu, and T.-Y. Liu, "On the depth of deep neural networks: A theoretical view," in *Proceedings of AAAI Conference on Artificial Intelligence*, Phoenix, Arizona, USA, February 12-17, 2016, pp. 920-927.

[39] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of International Conference on Learning Representations*, San Diego, CA, USA, May 7-9, 2015.

[40] L. Hong, T. Juanhui, and L. Mengyuan, "Two-stream 3D convolutional neural network for skeleton-based action recognition," *arXiv:1705.08106v2*, 2017.

[41] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, pp. 1-1, 2017.

[42] T. An and C. Loong-Fah, "Two-stream flow-guided convolutional attention networks for action recognition," *arXiv:1708.09268* 2017.

[43] H. Sepp and S. Jurgen, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735-1780, 1997.

[44] K. Xu, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. S. Zemel, *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of International Conference of Machine Learning*, Lille, France, July 6-11, 2015.

[45] S. Sharma, R. Kiros, and R. Salakhutdinov, "Action recognition using visual attention," in *Proceedings of International Conference on Learning Representations*, San Juan, Puerto Rico, May 2-4, 2016.

[46] C. Szegedy, L. Wei, J. Yangqing, P. Sermanet, S. Reed, D. Anguelov, *et al.*, "Going deeper with convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 1-9.

[47] Z. Li, K. Gavrilyuk, E. Gavves, M. Jain, and C. G. M. Snoek, "VideoLSTM convolves, attends and flows for action recognition," *Computer Vision and Image Understanding*, vol. 166, pp. 41-50, 2018.

[48] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, "Modeling spatial-temporal clues in a hybrid deep learning framework for video classification," in *ACM international conference on Multimedia*, Brisbane, Australia, 2015, pp. 461-470.

[49] J. Donahue, L. A. Hendricks, M. Rohrbach, S. Venugopalan, S. Guadarrama, K. Saenko, *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 677-691, 2017.

[50] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of International Conference on Learning Representations*, San Diego, CA, USA, May 7-9, 2015.

[51] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proceedings of International Conference on Learning Representations*, San Diego, CA, USA, May 7-9, 2015.

[52] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, 2014.

[53] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH, USA, June 23-28, 2014, pp. 1725-1732.

[54] E. Auvinet, C. Rougier, J. Meunier, A. St-Arnaud, and J. Rousseau, "Multiple camera fall dataset," *Technical Report*, 2010.

[55] I. ICharfi, J. Miteran, J. Dubois, M. Atri, and R. Tourki, "Definition and performance evaluation of a robust svm based fall detection solution," in *Proceedings of International Conference on Signal Image Technology and Internet Based Systems*, Naples, Italy, November 25-29, 2012, pp. 218-224.

[56] B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Computer Methods and Programs in Biomedicine*, vol. 117, pp. 489-501, 2014.

[57] S. Khurram, Z. Amir Roshan, and S. Mubarak, "UCF101: A dataset of 101 human actions classes from videos in the wild," *arXiv:1212.0402* 2012.

[58] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A large video database for human motion recognition," in *Proceedings of International Conference on Computer Vision*, Barcelona, November 6-13, 2011, pp. 2556-2563.

[59] J. Deng, W. Dong, R. Socher, L. J. Li, L. Kai, and F.-F. Li, "ImageNet: A large-scale hierarchical image database," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, June 20-25, 2009, pp. 248-255.