A Fusion of Appearance based CNNs and Temporal evolution of Skeleton with LSTM for Daily Living Action Recognition

Srijan Das, Michal Koperski, Francois Bremond
INRIA, Sophia Antipolis
2004 Rte des Lucioles, 06902, Valbonne, France
name.surname@inria.fr

Gianpiero Francesca
Toyota Motor Europe
Hoge Wei 33, B - 1930 Zaventem
gianpiero.francesca@toyota-europe.com

Abstract

In this paper, we propose efficient method which combines skeleton information and appearance features for daily-living action recognition. Many RGB methods focus only on short term temporal information obtained from optical flow. Skeleton based methods on the other hand show that modeling long term skeleton evolution improves action recognition accuracy. In this paper we propose to fuse skeleton based LSTM classifier which models temporal evolution of skeleton with deep CNN which models static appearance. We show that such fusion improves recognition of actions with similar motion and pose footprint, which is especially crucial in daily-living action recognition scenario. We validate our approach on public available CAD60 and MSRDailyActivity3D, achieving state-of-the art results.

1. Introduction

In this work we focus on solving problem of daily living action recognition. This problem facilitates many applications such as: video surveillance, patient monitoring and robotics. The problem is challenging due to complicated nature of human actions such as: pose, motion, appearance variation or occlusions.

Holistic RGB approaches focus on computing handcrafted or deep features (eg. CNN). Such methods usually model short term temporal information using optical flow. Long term temporal information is either ignored or modeled using sequence classifiers such as HMM, CRF or most recently LSTM.

Introduction of low-cost depth sensors and advancements in skeleton detection algorithms lead to increased research focus on skeleton based action recognition. 3D skeleton information allows to build action recognition methods based on high level features, which are robust to view-point and appearance changes [44]. Similarly to holistic methods – different ways of modeling temporal in-

formation were proposed: HMM, CRF, trees, LSTM.

In this paper we propose to fuse RGB information and skeleton information. We claim that it is especially important in daily-living action recognition, where many actions have similar motion and pose footprint (eg. drinking and taking pills), thus it is very important to model correctly appearance of objects involved in the action.

Secondly we claim that temporal evolution of skeleton is better captured by LSTM, than evolution of appearance features. Cognitive studies show that observation of skeleton joint points is a very strong clue in action recognition [13]. Evolution of appearance features is less evident. For instance let's consider drinking action and appearance evolution of patch around hand. The patch most of the time will contain hand and a cup, while angle changes of cup will be captured by appearance features, the changes are more subtle than temporal evolution of human skeleton. In addition occlusions may affect appearance features computation and introduce noise to temporal evolution of appearance features, thus we claim that appearance features modeling is much more difficult with LSTM.

This paper shows (1) that LSTM does not handle well high dimensional data, (2) in action recognition it is more effective to use stacked LSTMs over single LSTM. It can be explained in the way that first LSTM is modeling local low level temporal features, while top LSTM models high level more long term relationships. Our paper also shows, that proper LSTM architecture can be trained from scratch, even though there is little amount of data available. In our work we propose to use late fusion of skeleton based LSTM classifier with appearance based CNN classifier. Both classifier work independently and we fuse their classification scores to obtain final classification label. In this way we take advantage of LSTM classifier which is able to capture long term temporal evolution of pose and CNN classifier which focuses on static appearance features.

We validate our work on 2 public daily-activity datasets: CAD60, MSRDailyActivity3D. Our experiments show that we obtain new state-of-the-art results on both datasets.

2. Related Work

Earlier authors would use holistic representation and local representation of human body to recognize actions. Authors in [28, 38] recognized actions based on the extraction of local features. In [22], Laptev *et al.* have proposed a local representation for action recognition by extracting descriptor from interest point detection followed by aggregation of local descriptors. Then authors like [16, 5, 23] started using Histogram of Gradients (HoG), Histogram of Optical Flow (HoF) and Motion Boundary Histogram (MBH) as motion descriptors respectively.

Dense Trajectories [38] combined with Fisher Vector (FV) aggregation have shown good results for action recognition. Combining depth and RGB information have been proposed in [19, 18, 2, 6] for action recognition. The introduction of cheap kinect sensors have made it possible to detect the skeleton poses of human body easily which can be exploited to recognize actions as in [40]. Wu *et al.* [39] proposed a hierarchical dynamic framework that first extracts high level skeletal features and then uses the learned representation for estimating probability to infer action sequences.

Over the last few years, deep learning based approaches have become popular on video based human action recognition. The emergence of Deep Learning in computer vision has also affected the results in the accuracy of action recognition as they show some promising results [21]. One of the key point to use deep learning for action recognition is that, it not only focuses on extracting the deep features from CNNs but also consider the temporal evolution of these features using the Recurrent Neural Networks (RNNs). Since temporal information modeling the spatial features is an important dimension in action recognition, so we focus on using the fusion of deep spatial features along with temporal information to recognize actions. Authors in [14] have used pre-trained models for action recognition. In [8], the authors have used two stream networks for action recognition. One for appearance features from the RGB frames and one for flow based features from optical flow. They propose to fuse these features in the last convolutional layer rather than fusing them in the softmax layer which further improves their accuracy of the framework. Mahasseni *et al.* [26] have proposed that action recognition in video can be improved by providing an additional modality in training data namely, 3D human skeleton sequences. For recognition, they used Long Short Term Memory (LSTM) grounded via a deep CNN onto the video. They regularized the training of LSTM using the output of another encoder LSTM grounded on 3D human skeleton training data. Authors in [45] have represented 3-D skeletons using some geometric features and feed them into 3 layer LSTM. This framework not only works well on a single camera but also on cross view. They have also proposed the joint line distance to be the most dis-

criminative feature. But, the distance based features are not discriminative enough in case of daily living actions where the inter class variation is low. They also use joint coordinates on LSTM to classify actions and claim to overfit less as compared to other distance based features like pairwise, joint-line, plane-plane, joint-plane distance and so on, due to their less discriminative nature.

Recent action recognition tasks focus on using global aggregation of local features or temporal evolution of the human poses. In [4], the authors have used different parts of the skeleton so as to extract the CNN features from each of them. These features are aggregated with min-max pooling to classify the actions. The authors claim to use the temporal information by taking the difference of these CNN features followed by the max-min aggregation. But this aggregation ignores the temporal modeling of the spatial features. In [6], the authors have proposed a fusion of depth based and RGB based skeleton extracted from pose machines using the action recognition framework used in [4] to improve the daily living actions. This inspires us to use the parts based CNN features for spatial layout along with some temporal information to recognize daily living actions.

From [6], it is clear that parts based CNN features from depth based skeleton gives better action recognition rate. So, for action recognition network we use [4] where we take the RGB images along with the depth skeleton information to produce CNN features considering both the flow and appearance features. Moreover, we use the body translated joint coordinates from the depth information so as to find the temporal modeling of the actions using 2-layer LSTM followed by a SVM. Then, a similar kind of fusion is performed between the parts based CNN features and the temporal features as proposed in [6].

3. Proposed Method

The proposed method consists of extracting the human poses from depth information using kinect sensors which is discussed in section 3.1, extracting the parts based CNN features for spatial layout using the skeleton information along with the RGB frames (discussed in section 3.2), the LSTM architecture we use, for extracting the temporal information, are discussed in section 3.3 and section 3.4 refers to the fusion of parts based CNN features obtained from the action recognition network with the temporal features from LSTM.

3.1. Skeleton from RGB-D

Using depth sensors, like Microsoft Kinect or other similar devices, it is possible to design activity recognition systems exploiting depth maps, which are a good source of information because they are not affected by environment light variations, can provide body shape, and simplify the problem of human detection and segmentation. Further-

more, the availability of skeleton joints extracted from the depth frames allows having a compact representation of the human body that can be used in many applications. The depth based method use [33] to determine the body parts of the skeleton. They compute the body parts in a two stage process: first computing the depth map and then use it to classify the body parts. They have highlighted the importance of breaking the whole skeleton into parts. First the depth map is computed and then a million of depth invariant features from the depth images are trained by a randomized decision forest without overfitting, learning invariance to predict both pose and shape.

3.2. Pose based CNN features

We selected [4] as an action recognition network which uses two distinct networks based on appearance flow and optical flow respectively to produce concatenated CNN features. The input to this framework includes the skeleton information from kinect and the RGB frames. The skeleton information allows to crop the images around the joints to get the different body part patches. These patches include right hand, left hand, upper body, full body and full image. These patches are taken from both the RGB frames as well as the optical flow images. Then these images are resized to 224×224 and fed into two different CNN architectures respectively. Each networks consists of 5 convolutional and 3 fully connected layers. The output of the last layer consists of 4096 values which is considered as the frame descriptor. For the RGB patches, we use VGG-f network that has been pre-trained on the ImageNet ILSVRC-2012 challenge dataset. For the flow patches, we use the motion network provided by [9] that has been pre-trained for action recognition task on the UCF-101 dataset. In order to construct a fixed length video descriptor, they are aggregated over time using max and min pooling over all the frames to obtain a fixed-length video descriptor. In order to consider the dynamic features, the same aggregation strategy is applied on the difference of CNN features at time stamp $t + \delta t$ and t, where δt is 4 frames. Finally, video descriptors for motion and appearance from all the parts are normalized by dividing the video descriptors by the average l_2-norm of the descriptor for each part p over time frame t, f_t^p from the training set and concatenated to get the final CNN features.

Thus χ^2-kernel is computed from the CNN features constructed using the depth based skeleton and the RGB frames. This kernel is used to classify the actions. The kernel is computed using:

$$\chi^2(x, y) = 1 - \sum_{i=1}^{n} \frac{(x_i - y_i)^2}{\frac{1}{2}(x_i + y_i)} \qquad (1)$$

The χ^2-kernel is input to the SVM classifier for classification. The results in [6] also shows better action recognition
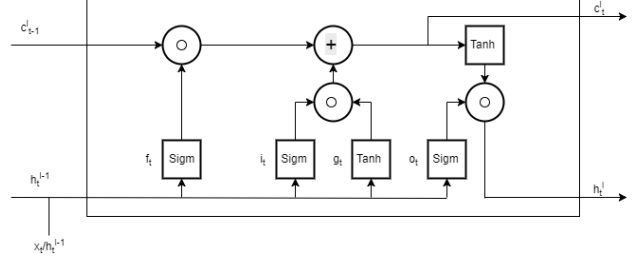

Figure 1. Structure of the neuron of LSTM.

rate for parts based CNN features taken from depth based skeleton. Thus, in this work, depth based skeleton is taken as input in the P-CNN framework.

3.3. LSTM architecture

In order to model the temporal relationship among the spatial features for action recognition, we use the advanced RNN architecture Long Short-Term Memory (LSTM) [10].

LSTM mitigates the vanishing gradient problem faced by RNNs by learning when to remember or forget information stored inside its internal memory cell (c^t). An LSTM cell is characterized by input (i), forget (f), self-recurrent (g) and output gate (o) as shown in fig. 1.

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} Sigm \\ Sigm \\ Sigm \\ Tanh \end{pmatrix} \left(W \begin{pmatrix} x_t \\ h_{t-1} \end{pmatrix} \right) \qquad (2)$$

$$c_t = f \circ c_{t-1} + i \circ g \qquad (3)$$

$$h_t = o \circ \tanh(c_t) \qquad (4)$$

Here, $\circ$ is the element wise multiplication operator and W is the weight matrix. The hidden state h_t is dependent on the cell state c_t which is characterized by the input, forget and self-recurrent gates. These gates enable the RNN to determine what new information are going to be stored in the next cell state and what old information should be discarded.

From the LRCN framework proposed in [7], we use the concept of using the CNN features on LSTM in order to model the temporal relationship between the evolution of activity. The LRCN framework tested on UCF101 [34] has actions with different dynamics and different environment, which results in an improvement in the action recognition rate. But in Daily living activities where, the actions have very similar motions and are acted in the same environment, these parts based CNN features cannot model the temporal evolution of the frames and hence cannot distinguish the actions effectively. So, we opted for exploiting the 3D skeletons on LSTM based on the analysis and framework proposed in [44]. Here, the authors have proposed that joint-line distance between the skeleton joints proved to be
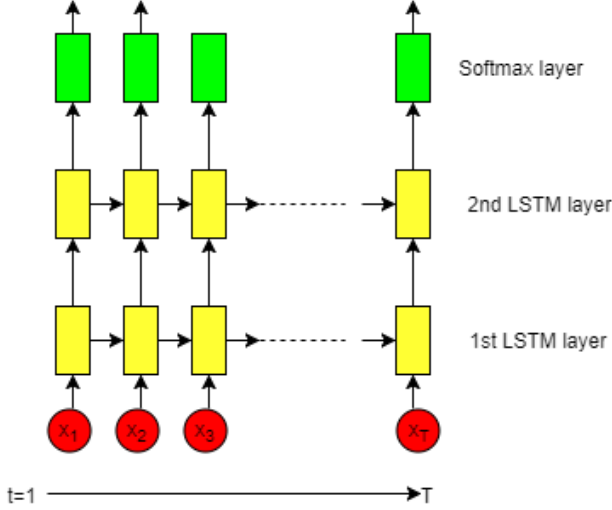
Figure 2. Proposed LSTM architecture where each row of yellow rectangles represent an LSTM layer.

the most discriminative feature. The authors in [44] tested their experiments on NTURGB+D [30] , SBU-kinect [43] datasets where the actions are much alike and distinguishable. But, based on our experimentations, we concluded that distance based features like pairwise distance, joint-line distance cannot distinguish actions where the motion are so similar. The results of these approaches are shown in the experimental section. So, in our approach, we use skeleton joint coordinates as input to the LSTM for considering the temporal evolution of the skeleton sequences. We use 2-layer LSTM and the main reason for stacking LSTM is to allow for greater model complexity. It allows to perform hierarchal processing on difficult temporal tasks and more naturally capture the structure of sequences. Moreover, we take the output features from the second LSTM and use them as the discriminative feature with a linear SVM. This improves the performance in capturing the temporal information since, we consider the output from all the time steps and does not only concentrate on the last time step which is a resultant from the previous time steps. In order, to ensure fixed size features input to the SVM, the video sequences are padded with maximum number of frames in the training video sequences. The LSTM learns to ignore the rows of zeros while training. The output from the 2-layer LSTM is given by :

$$y_i = \sigma(W_{hy}h_t^l) \qquad (5)$$

where W_{hy} is the output weight matrix, l=2 (in this case) and i=1...T. The output of each time steps from the 2nd layer LSTM is taken as the input to a linear SVM in order to classify the actions.
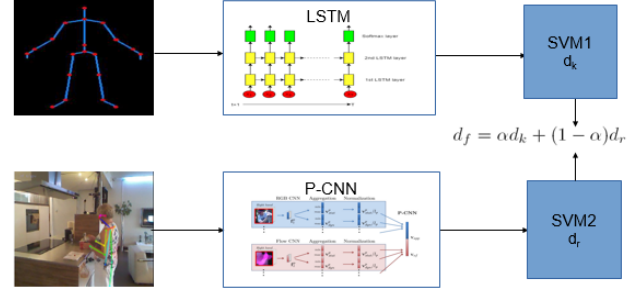

Figure 3. Proposed framework for action recognition.The two branches show two different inputs (skeleton and RGB frames). Skeleton features processed with LSTM, while RGB frames are modeled using P-CNN features. Both pipelines make independent classification.

3.4. Fusing Parts based CNN features with temporal evolution of Skeleton joints

The body based CNN features model the salient features in the global video. But these features are not discriminative enough to model the difference between actions with less intra-class variance. On the other hand, the features from LSTM models the temporal evolution of the salient features over the entire video. It captures the motion of the activity performed by a subject. So, the key idea in this paper is to fuse the appearance based CNN features with the temporal evolution of the body translated skeleton coordinates. Let's define d as the distance of test example to SVM decision plane, then d_k is the distance of test example to decision plane of SVM trained on input from parts based CNN features and d_r is the distance of test example to decision plane of SVM trained on input from LSTM. To fuse both the classifiers, we propose to use the following weighted sum

$$d_f = \alpha d_k + (1 - \alpha)d_r \qquad (6)$$

The value for α is found using cross-validation on the validation set.
Fig. 3 represents the whole pipeline of the proposed framework for action recognition includes CNN features using their part patches around the joints. Then, the framework includes fusing these CNN features with the temporal evolution of skeleton coordinates by fusing their classifier scores.

4. Experiments

4.1. Dataset description

CAD-60 [36] - contains 60 RGB-D videos with 4 subjects performing 14 actions each. This dataset is challenging because of less number of training videos available which leads to the problem of overfitting.

MSRDailyActivity3D [40] - contains 320 RGB-D videos with 10 subjects performing 16 actions each. This dataset has been captured in a living room and consists of all daily living activities. This dataset has different environment sequence for each action which makes it more challenging for recognizing actions using cropped part patches.

For both these public datasets, we have extracted skeleton information captured by kinect. We evaluate these datasets by setting up a cross actor training and testing validation setup. We left out one actor from the training set which is used for testing. This is repeated for all the subjects.

4.2. Implementation Details

We build our LSTM framework on the platform of keras toolbox [3] with TensorFlow [1]. The concept of Dropout [35] is used with a probability of 0.5 to eliminate the problem of overfitting. The concept of Gradient clipping [37] is used by restricting the norm of the gradient to not to exceed 1 in order to avoid the gradient explosion problem. Adam optimizer [15] initialized with learning rate 0.005 is used to train both the networks.

Since all the datasets, we chose consists of a single subject performing the actions, so we transform the camera coordinates to the body coordinates by translating the original point of body coordinate to the center of hips. For CAD60, we use actions performed by two subjects in the training set, action performed by a subject in validation set. For MSR-DailyActivity3D, we use actions performed by 8 subjects in the training set and actions performed by one subject in the validation set. We use actions performed by a subject in the test set. For CAD60, where the training set is very small and tends to overfit, we use 128 neurons each in the 2-layer LSTM. For MSRDailyActivity3D, we use 256 neurons each in the 2-layer LSTM. We set the batch sizes for CAD60 and MSRDailyActivity3D dataset to 17 and 32 respectively. For body coordinates with LSTM, the features from the last layer of the second LSTM are extracted and used with a SVM classifier to classify the actions.

4.3. Performance Comparison

In this section, we compare different techniques on the body based CNN features and body translated skeleton coordinates on LSTM and their fusion. We summarize a comparison between all these approaches in table 1. We perform our experiments on the standard daily living activity datasets and our proposed fusion approach clearly shows an improvement in recognizing the actions. Inspired from [7], we deployed the parts based CNN features with LSTM. But the LSTM fails to model the salient features over time. At the same time, when body translated skeleton joints are used with LSTM, it models the temporal relationship of the evolution of the joints sequences over a course of action.
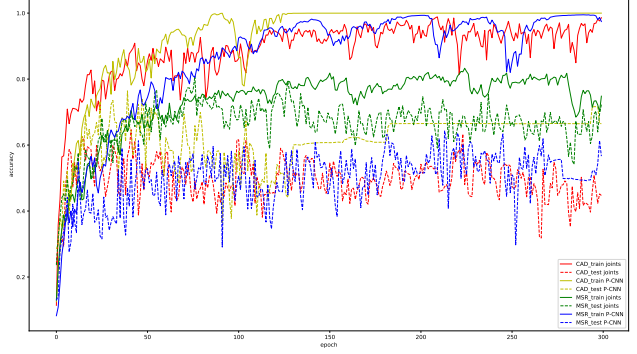

Figure 4. Performance of P-CNN features and Skeleton Coordinates features on 2-Layer LSTM on CAD60 (CAD) and MSRDailyActivity3D (MSR).

The LSTM with body translated joint coordinates perform better on MSRDailyActivity3D as compared to CAD60 because of the presence of more static actions with temporal evolution of some body parts of the subject in MSRDaily-Activity3D. Moreover, the features from the last layer of the 2-layered LSTM when input to the SVM classifier improves the classification accuracy by 3% and 8% approximately for CAD-60 and MSRDailyActivity3D respectively as evident from table 1.

Fig. 4 shows that training LSTM with CNN based features tends to overfit. For instance on MSR LSTM with CNN features obtains almost 100% of accuracy (continuous blue line), while on test set the accuracy is bellow 60% (blue dashed line). This indicates a problem with generalization. LSTM based on skeleton features suffers from this problem much less. Continuous and dashed green lines show, that gap between training and testing accuracy is much smaller. This indicates much better generalization of LSTM with skeleton features. Moreover, it is observed that measures like gradient clipping [37] and dropouts [35] hampers the LSTM to learn during training. Taking the features from LSTM to discriminate the actions with a SVM proved to be efficient in case of using the joint coordinates but not in case of the part based CNN features. So, we use body coordinates on LSTM to model the temporal relationship between the skeleton sequences. Thus, we propose a fusion of the parts based CNN features along with the temporal features from the LSTM in order to improve the static actions. Table 1 clearly shows that our proposed fusion improves in recognizing the daily living actions.

4.4. Discussion

In this subsection, we present the confusion matrix of CAD60 and MSRDailyActivity3D in order to analyze the actions whose recognition is improved due to our proposed fusion. Fig. 5 shows the confusion matrix of CAD60 on

Method	CAD60	MSRDailyActivity3D
P-CNN + Kinect [6]	94.11	83.75
P-CNN + Fusion [6]	95.58	84.37
P-CNN + LSTM	74.17	61.30
Body Coordinates + LSTM	64.65	80.90
Body Coordinates + LSTM + SVM	67.64	89.37
P-CNN + (Body Coordinates + LSTM) + SVM (Fusion)	**97.06**	**96.25**

Table 1. Comparison of different approaches with CNN features and body translated skeleton coordinates on CAD60 and MSRDailyActivity3D.
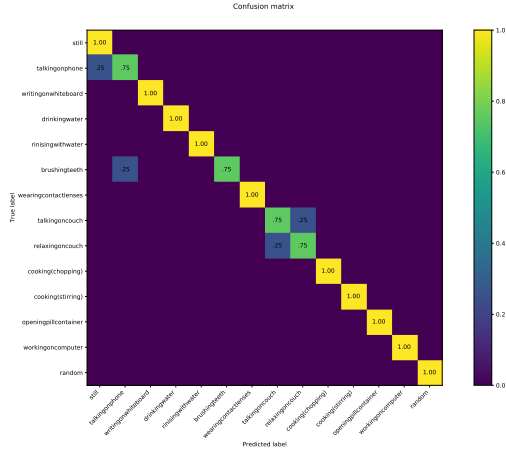

Figure 5. Confusion Matrix of P-CNN for CAD60 with depth based skeleton detection.
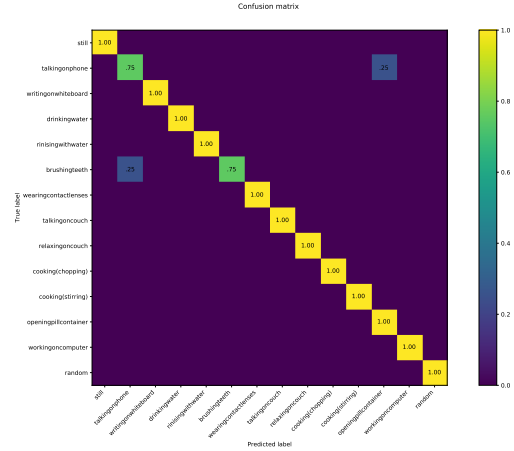

Figure 6. Confusion Matrix for CAD60 with fusion of P-CNN and Geometric Coordinates on LSTM.

P-CNN features with depth skeleton. The confusion matrix depicts that the actions like talking on couch and relaxing on couch are confused with each other because of the same posture of the subject and environment. But with taking the evolution of the skeleton sequences into account, they are recognized with full accuracy as evident from fig. 6. Though actions like talking on phone and brushing teeth are not improved with this fusion strategy because of lack of much skeleton sequence movements involved in these actions.

Fig. 7 and 8 represents the confusion matrix of MSR-DailyActivity3D on P-CNN and Fusion of P-CNN and geometric coordinates with LSTM followed by SVM respectively. They clearly show the improvement in recognition accuracy for static actions like reading, calling, writing, gaming and playing guitar along with actions like drinking water and sitting down which involves actions with a temporal change in skeleton sequences. For MSRDailyActivity3D, where most of the actions have temporal evolution of the skeleton sequences, geometric coordinates on LSTM followed by SVM classifier gives decent recognition accuracy.
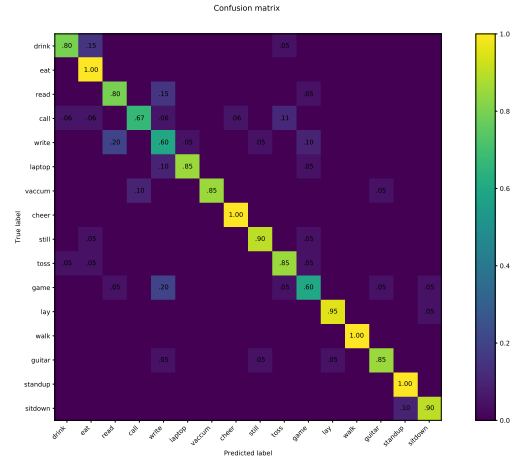

Figure 7. Confusion Matrix of P-CNN for MSRDailyActivity3D with depth based skeleton detection.

We also present a comparison of our proposed approach with the state-of-the-art methods on these datasets. Table 2 and 3 shows that our proposed approach outperforms the state-of-the-art results on CAD60 and MSRDailyAc-
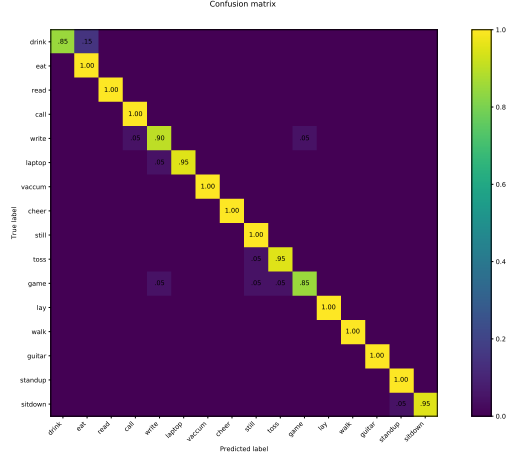
Figure 8. Confusion Matrix for MSRDailyActivity3D with fusion of P-CNN and Geometric Coordinates on LSTM.

Method	Accuracy [%]
STIP [46]	62.50
Order Sparse Coding [14]	65.30
Object Affordance [20]	71.40
HON4D [27]	72.70
Actionlet Ensemble [40]	74.70
MSLF [19]	80.36
JOULE-SVM [11]	84.10
P-CNN + kinect + Pose machines [6]	95.58
(P-CNN + kinect) +	
(Body coordinates + LSTM)	**97.06**

Table 2. Recognition Accuracy comparison for CAD-60 dataset. The performances of baseline methods are obtained from [18].

Method	Accuracy [%]
NBNN [29]	70.00
HON4D [27]	80.00
STIP + skeleton [46]	80.00
SSFF [32]	81.90
DSCF [41]	83.60
P-CNN + kinect + Pose machine [6]	84.37
Actionlet Ensemble [40]	85.80
RGGP + fusion [24]	85.60
MSLF [19]	85.95
Super Normal [42]	86.26
BHIM [17]	86.88
DCSF + joint [41]	88.20
JOULE-SVM [12]	95.00
Range Sample [25]	95.60
(P-CNN + kinect) +	
DSSCA-SSLM [31]	97.50
(Body coordinates + LSTM)	**96.25**

Table 3. Recognition Accuracy comparison for MSRDailyActivity3D dataset. The performances of baseline methods are obtained from [18].

A future direction of this work can be to understand the features responsible for accurately recognize the action. This can reduce the redundancy of features which are acting as noise to the classifier in the recognition process.

tivity3D. The fusion enhances the action recognition rate maximum for MSRDailyActivity3D where the actions have temporal sequences of the skeleton coordinates.

5. Conclusion

In this work, we propose a fusion of parts based CNN features and temporal evolution of skeleton joints to improve the daily living action recognition rate. In this work, we show that CNN based spatial features and distance based geometric features cannot discriminate daily actions where the actions are very similar and static. Thus, we propose to use body translated skeleton coordinates with LSTM followed by a linear SVM to compute the temporal information of the action sequences. Finally, we propose to use both the parts based CNN features along with the temporal evolution of the body sequences by fusing their classifier scores in order to improve the daily action recognition rate. Our approach achieves the state-of-the art results on publicly available CAD60 and MSRDailyActivity3D.

References

[1] M. Abadi et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] P. Bilinski, M. Koperski, S. Bak, and F. Bremond. Representing Visual Appearance by Video Brownian Covariance Descriptor for Human Action Recognition. In *AVSS*, 2014.

[3] F. Chollet et al. Keras, 2015.

[4] G. Chron, I. Laptev, and C. Schmid. P-cnn: Pose-based cnn features for action recognition. In *ICCV*, 2015.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893 vol. 1, June 2005.

[6] S. Das, M. Koperski, F. Bremond, and G. Francesca. Action recognition based on a mixture of rgb and depth based skeleton. In *AVSS*, 2017.

[7] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[8] C. Feichtenhofer, A. Pinz, and A. Zisserman. Convolutional two-stream network fusion for video action recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[9] G. Gkloxari and J. Malik. Finding action tubes. In *CVPR*, 2015.

[10] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, Nov. 1997.

[11] J.-F. Hu, W.-S. Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for RGB-D activity recognition. In *CVPR*, 2015.

[12] J. F. Hu, W. S. Zheng, J. Lai, and J. Zhang. Jointly learning heterogeneous features for rgb-d activity recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(11):2186–2200, Nov 2017.

[13] G. Johansson. Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, 14(2):201–211, Jun 1973.

[14] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-Scale Video Classification with Convolutional Neural Networks. In *CVPR*, 2014.

[15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[16] A. Klaser, M. Marszaek, and S. Cordelia. A spatio-temporal descriptor based on 3d-gradients. In *BMVC*, 2008.

[17] Y. Kong and Y. Fu. Bilinear heterogeneous information machine for RGB-D action recognition. In *CVPR*, 2015.

[18] M. Koperski, P. Bilinski, and F. Bremond. 3D Trajectories for Action Recognition. In *ICIP*, 2014.

[19] M. Koperski and F. Bremond. Modeling spatial layout of features for real world scenario rgb-d action recognition. In *AVSS*, 2016.

[20] H. S. Koppula, R. Gupta, and A. Saxena. Learning human activities and object affordances from rgb-d videos. *Int. J. Rob. Res.*, 32(8):951–970, July 2013.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[22] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, 2003.

[23] I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[24] L. Liu and L. Shao. Learning discriminative representations from rgb-d video data. In *IJCAI*, 2013.

[25] C. Lu, J. Jia, and C. K. Tang. Range-sample depth feature for action recognition. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 772–779, June 2014.

[26] B. Mahasseni and S. Todorovic. Regularizing lstm with 3d human-skeleton sequences for action recognition. In *CVPR*, 2016.

[27] O. Oreifej and Z. Liu. Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences. In *CVPR*, 2013.

[28] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *ICPR*, 2004.

[29] L. Seidenari, V. Varano, S. Berretti, A. Del Bimbo, and P. Pala. Recognizing actions from depth cameras as weakly aligned multi-part bag-of-poses. In *CVPRW*, 2013.

[30] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang. Ntu rgb+d: A large scale dataset for 3d human activity analysis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[31] A. Shahroudy, T. T. Ng, Y. Gong, and G. Wang. Deep multimodal feature analysis for action recognition in rgb+d videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2017.

[32] A. Shahroudy, G. Wang, and T.-T. Ng. Multi-modal feature fusion for action recognition in rgb-d sequences. In *ISCCSP*, 2014.

[33] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *CVPR*, 2011.

[34] K. Soomro, A. Roshan Zamir, and M. Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. 12 2012.

[35] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, Jan. 2014.

[36] J. Sung, C. Ponce, B. Selman, and A. Saxena. Unstructured human activity detection from rgbd images. In *ICRA*, 2012.

[37] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pages 3104–3112, Cambridge, MA, USA, 2014. MIT Press.

[38] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013.

[39] D. Wu and L. Shao. Leveraging hierarchial parametric networks for skeletal joints based action segmentation and recognition. In *CVPR*, 2014.

[40] Y. Wu. Mining actionlet ensemble for action recognition with depth cameras. In *CVPR*, 2012.

[41] L. Xia and J. Aggarwal. Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In *CVPR*, 2013.

[42] X. Yang and Y. Tian. Super normal vector for activity recognition using depth sequences. In *CVPR*, 2014.

[43] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras. Two-person interaction detection using body-pose features and multiple instance learning. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*. IEEE, 2012.

[44] S. Zhang, X. Liu, and J. Xiao. On geometric features for skeleton-based action recognition using multilayer lstm networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 148–157, March 2017.

[45] S. Zhang, X. Liu, and J. Xiao. On geometric features for skeleton-based action recognition using multilayer lstm networks. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 148–157, March 2017.

[46] Y. Zhu, W. Chen, and G. Guo. Evaluating spatiotemporal interest point features for depth-based action recognition. *Image and Vision Computing*, 32(8):453 – 464, 2014.