



Combining 2D and 3D deep models for action recognition with depth information

Ali Seydi Keçeli¹ · Aydın Kaya¹ · Ahmet Burak Can¹

Received: 16 June 2017 / Revised: 27 December 2017 / Accepted: 8 February 2018 / Published online: 16 March 2018
© Springer-Verlag London Ltd., part of Springer Nature 2018

Abstract

In activity recognition, usage of depth data is a rapidly growing research area. This paper presents a method for recognizing single-person activities and dyadic interactions by using deep features extracted from both 3D and 2D representations, which are constructed from depth sequences. First, a 3D volume representation is generated by considering spatiotemporal information in depth frames of an action sequence. Then, a 3D-CNN is trained to learn features from these 3D volume representations. In addition to this, a 2D representation is constructed from the weighted sum of the depth sequences. This 2D representation is used with a pre-trained CNN model. Features learned from this model and the 3D-CNN model are used in training of the final approach after a feature selection step. Among the various classifiers, an SVM-based model produced the best results. The proposed method was tested on the MSR-Action3D dataset for single-person activities, the SBU dataset for dyadic interactions, and the NTU RGB+D dataset for both types of actions. Experimental results show that proposed 3D and 2D representations and deep features extracted from them are robust and efficient. The proposed method achieves comparable results with the state of the art methods in the literature.

Keywords Action recognition · Dyadic actions · Deep learning · Feature selection · RGB-D data

1 Introduction

Human activity recognition is an old problem, which is generally studied with RGB videos in previous works [1–3]. With the progress of technology, depth sensors have become popular and are easily accessible in the open market. Since then, depth-based methods have been used to take advantage of depth data with the objective of improving recognition accuracy. By using real-time depth data from an RGB-D sensor, segmentation and detection of body parts could be achieved more accurately and more efficiently than pure RGB-based approaches.

Early studies in human activity recognition [1,2] used RGB videos and generally focused on recognizing single-

person actions. Grayscale intensity, texture, color, and motion-based features were used in these studies to recognize actions. Although some studies [4–6] use only depth data to recognize human activities, some other studies in the literature use both features extracted from depth maps and RGB video data [7]. Li et al. [8] define the actions as pose sequences and a transposition matrix that stores the probability of transpositions between different actions were proposed. After clustering recognized poses, actions are considered as combinations of prominent poses. Iosifidis et al. [9] constructed self-organizing maps from a multi-camera setup to recognize actions. Iosifidis et al. [10] also proposed another method based on binary action volumes. They obtained features from circular shift invariance property of the magnitudes of the Discrete Fourier Transform. Tsai et al. [11] extended conventional MHI approach with optical flow. Mahbub et al. [12] combined RANSAC and optical flow method for action recognition.

With the advent of RGBD sensors, depth data-based action recognition methods gained popularity. Xia et al. [13] proposed a method to map 3D joint locations to a spherical coordinate system. The histograms of spherical joint locations are computed and used as a view-invariant pos-

✉ Ali Seydi Keçeli
aliseydikeceli@gmail.com; aliseydi@cs.hacettepe.edu.tr

Aydın Kaya
aydinkaya@cs.hacettepe.edu.tr

Ahmet Burak Can
abc@cs.hacettepe.edu.tr

¹ Department of Computer Engineering, Faculty of Engineering, Hacettepe University, 06800 Ankara, Turkey

ture representation. Raptis et al. [14] proposed a method to recognize dance gestures. They used the joint skeleton model of Shotton et al. [15] to extract features. Sung et al. [16] proposed a double layer Hidden Markov Model (HMM) to recognize complex activities. Ryoo and Aggarwal [17] proposed a method that recognizes actions, composite actions, and interactions. This method is established on multi-level HMMs. Another hierarchical approach is proposed by Waltisberg et al. [7].

Comparing to single-person actions, interactions between persons are less studied in the literature. Most studies in the field of interaction detection used RGB surveillance videos or videos obtained from the Internet or streaming sites [4,18]. The hierarchical systems [16,17,19] are the most popular versions of the methods that use videos. As in the single-action recognition methods, these systems first detect simple actions on the first level of classification and then detect composite actions and interactions on the upper levels. Ji et al. [20] proposed depth data-based interaction recognition method. Another study that uses depth maps and skeletal features is proposed by Ji et al. [21]. This method also uses joint features like the distance and the motion of joints. They proposed a machine learning method called Contrastive Feature Distribution Model (CFDM) for prediction. The method proposed by Yun and et al. [22], utilized the Multiple Instance Learning (MIL) approach for interaction recognition. The proposed method used features extracted from a skeleton model, which is constructed from depth maps.

Recently, there have been many studies on action recognition based on deep learning. In deep learning, a large observation set is needed for estimation of space parameters. In traditional machine learning approaches, this operation is done with a limited set of observations. Ji et al. [23] applied a 3D-convolutional neural network (CNN) model into RGB videos for action recognition. Wu and Shao [24] used skeletal features with deep neural networks. Le et al. [25] utilized independent space analysis (ISA) with CNN for action recognition. Baccouche et al. [26] used deep learning in the recognition of action sequences. In this approach, action videos were taken as 3D input data and a 3D-CNN was constructed for action recognition. Wang et al. [27] used three-channel CNN's working with depth map sequences. Valle and Starostenko [28] used a 2D-CNN for recognition of walking and running actions. Tran et al. [29] employed a 3D-CNN on RGB videos to learn the spatiotemporal features of actions. Simonyan et al. [30] proposed a multi stream CNN architecture to recognize actions in RGB videos.

Deep learning based studies on action recognition generally work with 2D image sequences [23]. In this paper, a 3D-CNN and 2D-CNN-based method is proposed to learn spatiotemporal features from depth data. Unlike the other methods, the proposed method uses both 3D depth volumes and 2D templates and can classify both single person

actions and dyadic interactions. For each action, 3D and 2D-representations are obtained from depth sequences. 3D representations are constructed by considering spatiotemporal information of depth sequences and used for training of a 3D-CNN model. After training the 3D-CNN model, activation values from the fully connected layers are extracted. As the second stage of the method, we used a pre-trained 2D-CNN (AlexNet [31]) to extract deep features from 2D templates. The 2D templates are generated from the weighted sum of the depth maps. After combining features from 2D and 3D representations, Relieff algorithm is applied to select strong features. Several classification models are trained with the selected features. Trained models are tested on the MSR-Action3D [8] dataset for single-person actions and tested on the SBU [22] dataset for dyadic interactions. Additionally, the models are tested on NTU RGB+D dataset which is a large-scale action dataset [32].

2 Method

The flow of the proposed system can be divided into three parts. The first part is the preparation of the 3D and 2D representations. The second part is the feature extraction and the final part is model training. A schematic outline of the proposed method is shown in Fig. 1. 3D representation is generated by combining the depth maps in the spatiotemporal domain and then resizing the volume to a fixed size. A 3D-CNN is constructed from 3D volumes in the training set. For 2D representation, the weighted sum of the depth sequences is computed and given as input to a pre-trained CNN. Features are then extracted from the fully connected layers of the CNN's. Finally, the Relieff feature selection algorithm is applied to select strong features.

2.1 3D convolutional neural networks

A common CNN comprises one or more convolutional layers that are followed by one or more fully connected layers. In general, the architecture of a CNN is designed to take advantage of the 2D structured inputs. Thus CNN's are widely used on 2D image data to extract spatial features. Although such information is useful in many problems, temporal information from image sequences or videos is also needed in certain applications. In action recognition specifically, it is necessary to capture motion information from multiple frames. However, it is not easy to generate temporal information from 2D inputs. A third dimension would be useful to extract temporal information. 2D-CNNs can be extended for 3D structures such as 3D matrices and voxel grids. There are some 3D-CNN definitions for colored images [31]. These CNNs, which are also known as multichannel CNNs, use the third dimension to store a different type of spatial information, which is color

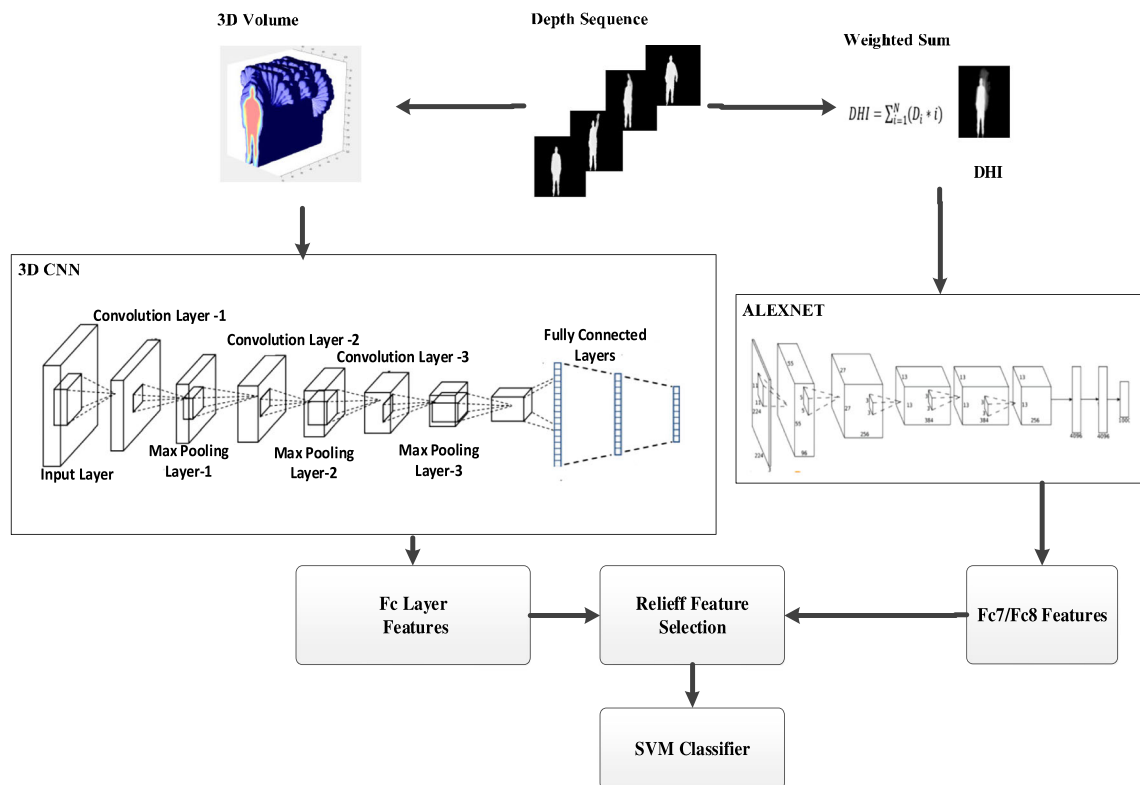


Fig. 1 Outline of the proposed method

information. However, the third dimension can be utilized to store temporal information. Therefore, we build a 3D-CNN to combine both spatial and temporal information of action data.

3D-CNN’s are based on 3D convolution operation, which is performed with 3D kernels. The feature maps are constructed from multiple concatenated frames. The feature map of the i^{th} layer is given in Eq. (1).

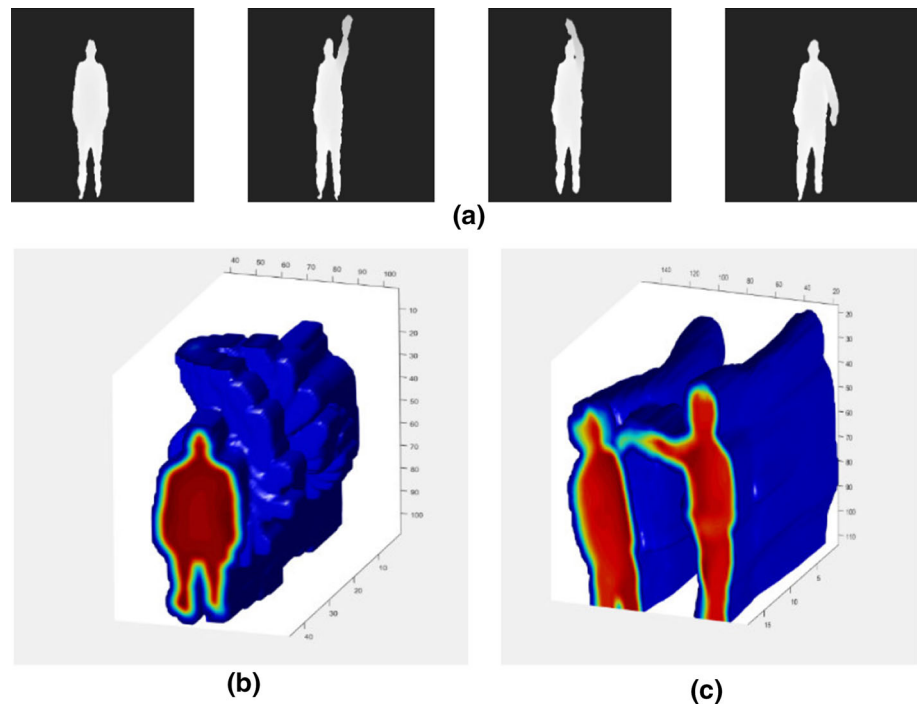
$$v_i^{xyz} = \tanh \left(b_i + \sum_m \sum_{p=0}^{P-1} \sum_{q=0}^{Q-1} \sum_{r=0}^{R-1} w_{im}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(z+r)} \right) \tag{1}$$

where P , Q , and R are the sizes of the kernel in spatial and temporal dimensions, v is the feature map, $v_{(i-1)m}^{(x+p)(y+q)(z+r)}$ is the kernel connected to the previous feature map, \tanh is the hyperbolic tangent function, b_i is the bias for current feature map, m is an index for the feature maps, and w_{im}^{pqr} is the value of the kernel function. The kernel function refers to the sets of weights that are convoluted with the input. The kernel function is a cuboid for a 3D CNN.

A CNN is formed with local connections and tied weights followed by pooling operations which results in subsampled

translation invariant features. The input of a convolutional layer is an $m \times n \times r$ sized 3D image, where m , n are the height and width of the image and r is the number of frames. In convolutional layers, there are k filters (or kernels) of size $h \times h \times q$. The first two dimensions (h) of the filter should be smaller than the dimensions of the input and the third dimension q can either be the same or smaller and may vary for each kernel. The kernel sizes of our three convolution layers are $9 \times 9 \times 9$, $7 \times 7 \times 7$, and $5 \times 5 \times 5$. We choose the kernel sizes experimentally. Very small filter sizes will capture very fine details of the image. On the other hand, having a bigger filter size will leave out details in the image. However conventional kernel sizes are 3×3 , 5×5 and 7×7 in various architectures [33]. We applied h^3 kernels as some 3D CNN architectures used in medical image segmentation [34]. Well-known classification architectures (AlexNet, CifarNet) contain subsequent convolution pooling layers, and some fully connected layers before softmax layers [31,35]. The filters are convolved with the input image and k feature maps are produced. The numbers of feature maps of the convolutional layers are 32, 64, and 128. Each map produced by the locally connected structures is then subsampled typically with max-pooling over $p \times p \times q$ contiguous regions. The additive bias and sigmoidal functions are performed to each feature map before or after the subsampling layer. The CNN is responsible for the high-level feature-learning task. After

Fig. 2 **a** Depth data of hand-waving action from the MSRAction3D dataset, **b** Volume data constructed from depth data of hand-waving action, and **c** Volume data for punching action from SBU dataset



3D convolution operations, 3D max-pooling is applied. Pooling layers are located between sequential convolution layers. Their function is to reduce the spatial size of the representation. The reduction of the number of parameters will reduce the computational costs. We apply $3 \times 3 \times 3$ subsampling on both max-pooling layers. Finally, three fully connected layer transformations are applied. The vector dimensions of the fully connected layers are 1048, 512, and 256.

2.2 3D depth volume representations

As explained above, 3D-CNNs can be used to represent both spatial and temporal information of action data. Since the proposed model uses depth data to recognize actions, depth frame sequences must be converted as an input for the proposed 3D-CNN model. Each depth frame is considered as a 2D image and is represented with the first two dimensions of the 3D input matrix. The third dimension is considered as a time dimension and consecutive frames are added to this dimension to represent the temporal information. Normalization is applied to the 3D input data before giving it to the convolutional model. Values of all voxels are fitted into the interval 0–1. Min–Max normalization is applied at this stage to reduce the disparity value variations between the templates. Eq. 2 is applied to all points in the depth maps. Min and Max values are computed from the whole dataset.

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)} \quad (2)$$

All input volumes are resized to a fixed size. First, depth data is resized to meet the input dimensions of the 3D matrix. The size of all depth sequences are fixed to 64×64 in x and y dimensions and 30 in the t -dimension (frames). This volume data structure enables us to represent both spatial and temporal information in a single input format. Since some actions have less or more than 30 frames, trilinear interpolation is applied over 3D volumes for resizing t -dimension. Trilinear interpolation is a multivariate interpolation over the 3D grid, and it is an extension of linear interpolation. While linear interpolation operates in 1D spaces, bilinear and trilinear interpolation operates in 2D and 3D, respectively. For short action sequences, some new pixels are constructed by means of interpolation so t -dimension is enlarged. For long sequences, t -dimension is shrunk by applying interpolation. Thus all resized volumes have 30-frames length with this method. During the experiments, although different t -dimension sizes are tested between 20 and 40, best results are obtained with numbers greater than 25 frames. No significant change is observed after 30, so this parameter is selected as 30. Sample depth images of a hand-waving depth sequence and volumes that are obtained from these sequences are shown in Fig. 2.

Finally, we feed the 3D-CNN model with multiple 3D volumes. The activation values computed by the fully connected layers of the 3D-CNN are extracted and used as features. The features obtained from the connected layers are used in the training of the SVM classifier. In some studies in the literature, SVM classifiers work better with deep features [36], which is also verified by our experimental results.

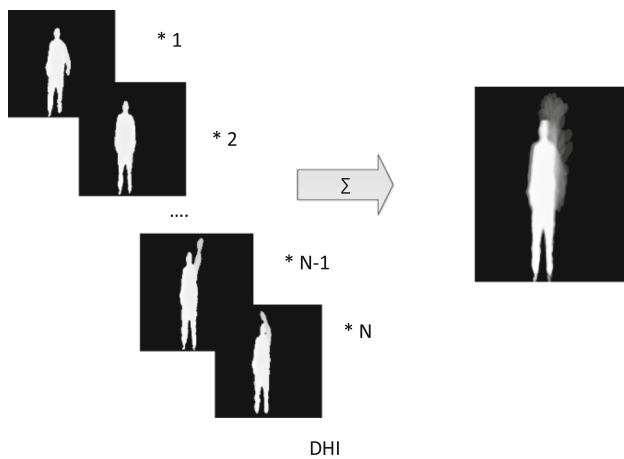


Fig. 3 Depth maps of hand wave action and generated DHI

2.3 2D depth representation

In addition to features from 3D-CNN, we generate a 2D representation from the depth image sequences and apply transfer learning to extract robust features. Deep convolutional networks trained on large-scale image databases usually outperform handcrafted features [31]. In most of the cases, training a convolutional neural network from scratch is not effective when there is no huge amount of training data. For such cases, the common practice in deep learning is to use a network trained previously on a large dataset. We extracted deep features from a 2D representation generated from depth data. For each depth sequence, the weighted sums of the depth sequences are computed and given to a pre-trained CNN, AlexNet, as input.

The weighted sum is a static image template to define action sequence. In this template, pixel intensity values represent the frequency and depth information. The template is generated by computing weighted sum of depth maps. The template of action sequences is computed with Eq. (3). In Eq. (3) D_i is the i th depth image in the sequence, N is the frame count in the sequence, and i is the frame index number. This generated template is called as Depth History Image (DHI) [37]. The sample depth maps of hand wave action and template calculated from this action are shown in Fig. 3.

$$\text{DHI} = \sum_{i=1}^N (D_i * i) \quad (3)$$

In the template image, the more recent frames will be represented with higher intensity values as in the Motion History Image (MHI) [38]. Different from MHI, we have used depth maps to define pixel intensity as a function of the motion. So, a DHI captures spatial information of action location and temporal information by indicating more recent action phases. AlexNet is fed with DHI images in order to learn

features. The input layer of the CNN is fixed, so all the input data are resized to 227×227 pixels, and all grayscale images are converted to three channels through replication. When learning features using AlexNet, we utilize representations from fully connected layer 7 (Fc7) and 8 (Fc8) and use the activations of Fc7 and Fc8 as features. Representations obtained from different layers correspond to varying levels of abstraction [39]. CNN features are more generic in early layers and more original dataset-specific in later layers [40]. In our experiments, we have observed that Fc7 and Fc8 activations generally produce the best results.

2.4 Feature selection and classification

After feature learning, the Relieff [41] feature selection algorithm is applied to the learned features. Feature selection is a process of selecting a discriminative subset of features by computing the weights of features. The weight is the strength of a feature in representing a specified class or cluster. Relieff is a supervised feature selection algorithm based on error minimization [41]. This algorithm aims to select a subset of features that maximize the classification accuracy. Selection is made by finding the subset that minimizes the Bayesian error rate. The original Relief algorithm is for two-class data and could only work with nominal and numerical data. The Relieff approach solves the multi-class and incomplete, noisy data problems.

All features that are obtained from 3D-CNN and AlexNet are concatenated, and then feature selection is applied to gather a more robust and stronger feature subset. Approximately the strongest first 10% of the features are selected after Relieff application. When selecting this ratio of 10%, several ratios are studied with experiments. Ratios below 10% have a negative effect on classification accuracy. There is no significant improvement on classification accuracy between 10% and 50%. Thus, we select the first 10% for both improving the accuracy and reducing the high dimensionality. Finally, a linear kernel SVM predictor is trained with these selected features and cross-validation is applied to optimize SVM parameters. The use of deep features with a linear SVM classifier usually produce better results compared to Softmax classifier [29]. In our experiments, we observed that linear SVMs produce remarkable results with deep features.

3 Experiments

The proposed method is evaluated on the SBU, MSRAction-3D, and NTU RGB+D datasets. The datasets used in the experiments are recorded with RGBD sensors. 2D and 3D representations for an action are generated by reading frames of the action in these datasets. All experiments and model trainings were made on a work station with an Intel i7 3.4

Table 1 Detection performance of various classifiers on SBU, MSRAction3D, and NTU RGB+D dataset

Methods	SBU	MSRAction3D	NTU RGB+D
3D-CNN	89.7	81.3	52.8
DHI+Relieff+SVM	92.8	90.1	37.1
3D-CNN+SVM	91.9	82.7	54.2
3D-CNN+Relieff+SVM	92.8	86.5	55.8
3D-CNN+DHI+Relieff+SVM	94.7	92.8	61.9
3D-CNN+DHI+Relieff+KNN	86.4	81.5	34.6
3D-CNN+DHI+Relieff+RF	87.5	91.7	42.3

GHz CPU and 32 GB of memory and Nvidia GTX760 GPU. Training time for the 3D-CNN model is approximately 15–16 hours for MSRAction-3D and SBU datasets. For NTU RGB+D dataset training time takes multiple days. Although initial training of the model takes too long, feature extraction from a trained model just takes few seconds. 2D template generation and feature extraction from AlexNet take less than a second for a single action.

3.1 Experiments on SBU dataset

We first evaluate the efficiency of the proposed model on the SBU dataset and compare the results with state of the art methods. Actions in the SBU dataset are as follows: approaching, kicking, handshaking, pushing, departing, punching, hugging, and exchanging objects. All actions are performed by seven different individuals, and there are $C(7,2) = 21$ pairs of subjects. All subject pairs perform all of the eight interactions in the dataset. The dataset consists of synchronized RGB frames and depth maps for all interactions. Resolution of the depth maps is 640×480 pixels. There are 15 joints for each subject, and the normalized joint coordinates are also provided in the dataset.

Table 1 lists the results on the SBU dataset with different combinations of the proposed model. In Table 1, the results are in the following order: the first row (3D-CNN) contains the classification results obtained from the Softmax layer of our 3D-CNN architecture; the second row (DHI+Relieff+SVM) includes the classification results obtained from the SVM predictor trained with selected 2D deep features; the third row (3D-CNN+SVM) contains the results obtained from the linear SVM trained with deep features acquired from fully connected layers of the 3D-CNN. The last 3 rows show the results obtained from various classifiers trained with selected 2D and 3D deep features.

In Table 1, the results on the SBU dataset are obtained with a fivefold cross-validation, since this validation method is also used in the related previous studies. Different types of base classifiers (e.g., Random Forest, KNN) are tested, and the best results were acquired with the linear SVM classifier.

3.2 Experiments on MSRAction-3D dataset

The MSR Action 3D dataset contains 20 different types of actions from 10 subjects. The actions in this dataset are high arm wave, horizontal arm wave, hammer, hand catch, forward punch, high throw, draw x, draw tick, draw circle, hand clap, two hand wave, side boxing, bend, forward kick, sidekick, jogging, tennis swing, tennis serve, golf swing, and pick up and throw. The results are obtained with cross-subject testing on the MSRAction3D dataset as in the most of the action recognition studies on the MSRAction3D dataset. In cross-subject testing, actions performed by the half of the subjects are used in training, and actions of the remaining subjects are used in testing. The training of 3D-CNN is a time-consuming process, therefore this test is applied with only one combination. As in the SBU dataset, various combinations of features from 3D-CNN and AlexNet are tested with different classifiers. The experimental results obtained on the MSRAction3D dataset are given in Table 1. The RF classifier produces close results to the linear kernel SVM for this dataset. For this dataset, the most confused actions are jogging and the sidekick. Generally, the jogging action is confused with the sidekick. The reason for this confusion is that the movement path of legs in both actions. The other most confused actions are the tennis swing and tennis serve due to the similarity of these actions.

3.3 Experiments on NTU RGB+D dataset

NTU RGB+D dataset is large-scale action recognition dataset gathered from 40 different subjects. This dataset contains 60 different actions including daily, mutual and health-related actions. The results obtained from this dataset are given in Table 1. Due to high number of action classes and action complexities, classification accuracy is low compared with the results obtained from the other datasets. The large intra-class and viewpoint variations make this dataset very challenging. Cross-Subject testing is applied to this dataset like MSRAction 3D. Front view data are used in our experiments.

3.4 Comparison with other studies

As observed from above experimental results, the best results were obtained from combined features. As shown in Table 1, 3D and 2D representation based end-to-end models produced lower results than the models that combine features from different models. Additionally, employing a classifier with learned deep features can give better results compared with a softmax layer. Especially, the SVM classifier trained with deep features produce better results in small datasets. We can conclude that 3D-CNN features and features obtained by transfer learning are complementary with each other.

Table 2 Comparison with the other methods on SBU dataset

Methods	CA
CFDM [21]	89.4
Linear SVM [22]	86.2
Contrast mining [20]	86.9
Our method	94.7

Table 3 Comparison with the other methods on MSRAction3D dataset

Methods	CA
Bag of 3D points [8]	74.7
HOJ3D [13]	79.0
EigenJoints [42]	83.3
HON4D [43]	88.8
Yang et al. [44]	85.5
Actionlet ensemble [45]	88.2
Joint angles similarities [46]	94.84
Our method	92.8

In Table 2, the results of the proposed method on the SBU dataset are compared with other methods from the literature. Our method has the best results on the SBU dataset. The results show that the proposed model shows a significant improvement over the models [20–22]. CFDM [21], which use contrastive features of interactions to create a dictionary, has the closest performance to our model.

In Table 3, the results of the proposed method on the MSRAction3D dataset are compared with the other methods from the literature. For the MSRAction3D dataset, the proposed method gives reasonable and comparable results. Although our method does not have the best result, it is still better than the most of the studies in the literature. Because of high view and subject variation of action sequences in this dataset, classification accuracy is slightly lower. The high classification ratio on both datasets states that our method can learn the dataset invariant features. The superiority of deep learning plays an important role in these results by extracting descriptive features to define structural information of action sequences.

Comparison of the results on NTU RGB+D dataset with other methods are shown in Table 4. The proposed method did not outperform the other methods in the literature but produced promising results. Due to complexity and size of this dataset, classification accuracies on this dataset is not as good as the results obtained from other datasets. The first reason for low accuracy is depth information of the unrelated foreground objects. Although we have used masked depth maps, there have been objects that occlude silhouettes of the actors partly. These objects also caused noise in the 2D and 3D templates. Using skeletal features for this dataset could be more beneficial.

Table 4 Comparison with the other methods on NTU RGB+D dataset

Methods	CA
Shahroudy et al. [32]	62.93
Liu et al. [47]	69.2
Du et al. [48]	59.07
Liu et al. [49]	66.85
Song et al. [50]	73.4
Our method	61.9

4 Conclusion

In this paper, we proposed an action recognition method that can be used in single-person actions or dyadic interactions. Firstly, high-level deep features extracted from depth sequences were extracted with deep convolutional neural networks. We defined 3D representations to capture temporal information of an action. 3D volume representations were given as input to the 3D-CNN model to gather deep features defining the temporal structure of the action. Then 2D representations were generated to learn features from a pre-trained CNN by using transfer learning approach. The features from 2D and 3D CNN models were combined and ranked with the Relieff algorithm to select strong features. Finally, an SVM classifier was trained with these features.

The developed framework was evaluated on SBU, MSRAction3D, and NTU RGB+D datasets. By combining features from 3D and 2D representations, we achieved the best classification results on SBU dataset and comparable results on MSRAction3D and NTU RGB+D datasets. Although the results obtained directly from deep features were good in general, applying Relieff algorithm on the features helped to increase the classification accuracy and reduce computational cost. Selecting features with the Relieff algorithm increased the performance by 2–3% for all classifiers.

In the future studies, we will explore combining multiple types of data for interaction detection. In other words, we plan to combine features from raw RGB and depth data features with skeletal features to obtain better results.

References

1. Shechtman, E., Irani, M.: Space–time behavior based correlation. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 405–412. IEEE (2005)
2. Ke, Y., Sukthankar, R., Hebert, M.: Spatio-temporal shape and flow correlation for action recognition. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. IEEE (2007)
3. Pei, L.S., Ye, M., Zhao, X.Z., Xiang, T., Li, T.: Learning spatio-temporal features for action recognition from the side of the video. *Signal Image Video Process.* **10**, 199–206 (2016)
4. Ryoo, M., Chen, C.-C., Aggarwal, J., Roy-Chowdhury, A.: An overview of contest on semantic description of human activities

- (SDHA) 2010. In: *Recognizing Patterns in Signals, Speech, Images and Videos*, pp. 270–285. Springer (2010)
5. Zhang, C., Platt, J.C., Viola, P.A.: Multiple instance boosting for object detection. In: *Advances in Neural Information Processing Systems*, pp. 1417–1424 (2005)
 6. Al Ghamdi, M., Zhang, L., Gotoh, Y.: Spatio-temporal SIFT and its application to human action classification. In: *European Conference on Computer Vision*, pp. 301–310. Springer (2012)
 7. Waltisberg, D., Yao, A., Gall, J., Van Gool, L.: Variations of a hough-voting action recognition system. In: *Recognizing Patterns in Signals, Speech, Images and Videos*, pp. 306–312. Springer (2010)
 8. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3D points. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pp. 9–14. IEEE (2010)
 9. Iosifidis, A., Tefas, A., Pitas, I.: View-invariant action recognition based on artificial neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **23**, 412–24 (2012)
 10. Iosifidis, A., Tefas, A., Pitas, I.: Multi-view action recognition based on action volumes, fuzzy distances and cluster discriminant analysis. *Signal Process.* **93**, 1445–57 (2013)
 11. Tsai, D.M., Chiu, W.Y., Lee, M.H.: Optical flow-motion history image (OF-MHI) for action recognition. *Signal Image Video Process.* **9**, 1897–906 (2015)
 12. Mahbub, U., Imtiaz, H., Ahad, M.A.R.: Action recognition based on statistical analysis from clustered flow vectors. *Signal Image Video Process.* **8**, 243–53 (2014)
 13. Xia, L., Chen, C.-C., Aggarwal, J.: View invariant human action recognition using histograms of 3D joints. In: *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 20–27. IEEE (2012)
 14. Raptis, M., Kirovski, D., Hoppe, H.: Real-time classification of dance gestures from skeleton animation. In: *Proceedings of the 2011 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 147–156. ACM (2011)
 15. Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., et al.: Real-time human pose recognition in parts from single depth images. *Commun. ACM* **56**, 116–24 (2013)
 16. Sung, J., Ponce, C., Selman, B., Saxena, A.: Human activity detection from RGBD images. In: *Plan, Activity, and Intent Recognition AAAI Workshop* (2011)
 17. Ryoo, M.S., Aggarwal, J.K.: Recognition of composite human activities through context-free grammar based representation. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, pp. 1709–1718. IEEE (2006)
 18. Ryoo, M.S., Aggarwal, J.K.: Spatio-temporal relationship match: video structure comparison for recognition of complex human activities. In: *2009 IEEE 12th International Conference on Computer Vision*, pp. 1593–1600. IEEE (2009)
 19. Park, S., Aggarwal, J.K.: A hierarchical Bayesian network for event recognition of human actions and interactions. *Multimed. Syst.* **10**, 164–79 (2004)
 20. Ji, Y., Ye, G., Cheng, H.: Interactive body part contrast mining for human interaction recognition. In: *2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, pp. 1–6. IEEE (2014)
 21. Ji, Y., Cheng, H., Zheng, Y., Li, H.: Learning contrastive feature distribution model for interaction recognition. *J. Vis. Commun. Image Represent.* **33**, 340–9 (2015)
 22. Yun, K., Honorio, J., Chattopadhyay, D., Berg, T.L., Samaras, D.: Two-person interaction detection using body-pose features and multiple instance learning. In: *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 28–35. IEEE (2012)
 23. Ji, S., Xu, W., Yang, M., Yu, K.: 3D convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal.* **35**, 221–31 (2013)
 24. Wu, D., Shao, L.: Leveraging hierarchical parametric networks for skeletal joints based action segmentation and recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 724–731 (2014)
 25. Le, Q.V., Zou, W.Y., Yeung, S.Y., Ng, A.Y.: Learning hierarchical invariant spatio-temporal features for action recognition with independent subspace analysis. In: *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3361–3368. IEEE (2011)
 26. Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., Baskurt, A.: Sequential deep learning for human action recognition. In: *International Workshop on Human Behavior Understanding*, pp. 29–39. Springer (2011)
 27. Wang, P., Li, W., Gao, Z., Zhang, J., Tang, C., Ogunbona, P.O.: Action recognition from depth maps using deep convolutional neural networks. *IEEE Trans. Hum. Mach. Syst.* **46**, 498–509 (2016)
 28. Valle, E.A., Starostenko, O.: Recognition of human walking/running actions based on neural network. In: *2013 10th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, pp. 239–244. IEEE (2013)
 29. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3D convolutional networks. In: *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 4489–4497. IEEE (2015)
 30. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. In: *Advances in Neural Information Processing Systems*, pp. 568–576 (2014)
 31. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105 (2012)
 32. Shahroudy, A., Liu, J., Ng, T.T., Wang, G.: NTU RGB plus D: a large scale dataset for 3D human activity analysis. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (Cvpr)*, pp. 1010–1019 (2016)
 33. Mishkin, D., Sergievskiy, N., Matas, J.: Systematic evaluation of CNN advances on the ImageNet. arXiv preprint [arXiv:1606.02228](https://arxiv.org/abs/1606.02228) (2016)
 34. Kamnitsas, K., Ledig, C., Newcombe, V.F.J., Simpson, J.P., Kane, A.D., Menon, D.K., et al.: Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Med. Image Anal.* **36**, 61–78 (2017)
 35. Shin, H.C., Roth, H.R., Gao, M.C., Lu, L., Xu, Z.Y., Nogues, I., et al.: Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE Trans. Med. Imag.* **35**, 1285–98 (2016)
 36. Tang, Y.: Deep learning using linear support vector machines. arXiv preprint [arXiv:1306.0239](https://arxiv.org/abs/1306.0239) (2013)
 37. Keceli, A.S., Can, A.B.: A multimodal approach for recognizing human actions using depth information. *Int. Conf. Pattern Recognit.* **22**, 421–426 (2014)
 38. Ahad, M.A.R.: *Motion history images for action recognition and understanding*. Springer Science & Business Media, Berlin (2012)
 39. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., et al.: DeCAF: a deep convolutional activation feature for generic visual recognition. In: *ICML*, pp. 647–655 (2014)
 40. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: “How transferable are features in deep neural networks?”, *Advances in neural information processing systems*, 3320–8 (2014)
 41. Kononenko, I., Šimec, E., Robnik-Šikonja, M.: Overcoming the myopia of inductive learning algorithms with RELIEFF. *Appl Intell* **7**, 39–55 (1997)
 42. Yang, X., Tian, Y.L.: “Eigenjoints-based action recognition using naive-bayes-nearest-neighbor”, 2012 IEEE Computer Society

- Conference on Computer Vision and Pattern Recognition Workshops. IEEE, pp. 14-9 (2012)
43. Oreifej, O., Liu, Z.: “Hon4d: Histogram of oriented 4d normals for activity recognition from depth sequences”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 716-23 (2013)
 44. Yang, X., Zhang, C., Tian, Y.: “Recognizing actions using depth motion maps-based histograms of oriented gradients”, Proceedings of the 20th ACM international conference on Multimedia. ACM, pp. 1057-60 (2012)
 45. Wang, J., Liu, Z.C., Wu, Y., Yuan, J.S.: “Mining Actionlet Ensemble for Action Recognition with Depth Cameras”. 2012 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr), pp. 1290-7 (2012)
 46. Ohn-Bar, E., Trivedi, M.: “Joint angles similarities and HOG2 for action recognition”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 465-70 (2013)
 47. Liu, J., Shahroudy, A., Xu, D., Wang, G.: “Spatio-Temporal LSTM with Trust Gates for 3D Human Action Recognition”. Computer Vision - Eccv 2016, Pt Iii, 9907, pp. 816-33 (2016)
 48. Du, Y., Wang, W., Wang, H.: “Hierarchical Recurrent Neural Network for Skeleton Based Action Recognition”. 2015 Ieee Conference on Computer Vision and Pattern Recognition (Cvpr), pp. 1110-8 (2015)
 49. Liu, H., Tu, J., Liu, M.: “Two-Stream 3D Convolutional Neural Network for Skeleton-Based Action Recognition”. arXiv preprint [arXiv:1705.08106](https://arxiv.org/abs/1705.08106) (2017)
 50. Song, S., Lan, C., Xing, J., Zeng, W., Liu, J.: “An End-to-End Spatio-Temporal Attention Model for Human Action Recognition from Skeleton Data”, AAAI, pp. 4263-70 (2017)